

QGIS Desktop 3.44 User Guide

QGIS Project

14 de marzo de 2026

Índice general

1 Comenzando **3**
- 1.1 Instalando QGIS . . . 3
  - 1.1.1 Instalación desde binarios . . . 3
  - 1.1.2 Instalando desde la fuente . . . 3
  - 1.1.3 Instalación en medios externos . . . 4
  - 1.1.4 Descargando datos de muestra . . . 4
- 1.2 Iniciando y deteniendo QGIS . . . 4
- 1.3 Sesión de muestra: carga de capas ráster y vectoriales . . . 5

2 Trabajando con Archivos de Proyecto **13**
- 2.1 Introducción a proyectos QGIS . . . 13
- 2.2 Manejo de rutas de archivo rotas . . . 15
- 2.3 Generando la salida . . . 16

3 IGU QGIS **17**
- 3.1 Barra de Menú . . . 18
  - 3.1.1 Proyecto . . . 18
  - 3.1.2 Editar . . . 19
  - 3.1.3 Ver . . . 22
  - 3.1.4 Capa . . . 25
  - 3.1.5 Configuración . . . 28
  - 3.1.6 Complementos . . . 28
  - 3.1.7 Vectorial . . . 28
  - 3.1.8 Ráster . . . 30
  - 3.1.9 Base de datos . . . 32
  - 3.1.10 Web . . . 32
  - 3.1.11 Malla . . . 33
  - 3.1.12 Procesado . . . 33
  - 3.1.13 Ayuda . . . 33
  - 3.1.14 QGIS . . . 34
- 3.2 Paneles y Barras de Herramientas . . . 34
  - 3.2.1 Barras de herramientas . . . 34
  - 3.2.2 Paneles . . . 36
- 3.3 Barra de Estado . . . 36
  - 3.3.1 Barra de localización . . . 36
  - 3.3.2 Reporte de acciones . . . 37
  - 3.3.3 Controlar el lienzo del mapa . . . 37
  - 3.3.4 Mensajería . . . 38

4 El panel Navegador **39**
- 4.1 Recursos que se pueden abrir/ejecutar desde el navegador . . . 41

4.2 Entradas de nivel superior del panel del navegador . . . 42
4.2.1 Favoritos . . . 42
4.2.2 Marcadores espaciales . . . 42
4.2.3 Proyecto Inicio . . . 42
4.2.4 Unidades y sistema de archivos . . . 42
4.2.5 Tipo de datos . . . 43
4.2.6 Entradas de la base de datos . . . 43
4.2.7 Mosaicos y servicios web . . . 47
4.2.8 Proyecto y tipos de archivo compatibles . . . 49

5 Configuración QGIS **51**
5.1 Opciones . . . 51
5.1.1 Configuración general . . . 52
5.1.2 Ajustes del sistema . . . 54
5.1.3 Configuración de los perfiles de usuario . . . 56
5.1.4 Configuración de CRS y Transformaciones . . . 56
5.1.5 Ajustes de Fuentes de Datos . . . 59
5.1.6 Ajustes de Renderización . . . 63
5.1.7 Ajustes de Lienzo y Leyenda . . . 66
5.1.8 Ajustes de herramientas de mapa . . . 68
5.1.9 Ajustes de elevación . . . 72
5.1.10 Configuración 3D . . . 72
5.1.11 Configuración de colores . . . 73
5.1.12 Configuración de fuentes . . . 74
5.1.13 Ajustes de Composiciones . . . 75
5.1.14 Configuración de variables . . . 76
5.1.15 Configuración de la autenticación . . . 76
5.1.16 Configuración de la red . . . 77
5.1.17 Ajustes del GPS . . . 79
5.1.18 Configuración del localizador . . . 81
5.1.19 Ajustes de aceleración . . . 83
5.1.20 Configuración IDE . . . 83
5.1.21 Ajustes de Procesos . . . 87
5.1.22 Ajustes avanzados . . . 88
5.2 Trabajando con perfiles de usuario . . . 88
5.2.1 El concepto . . . 88
5.2.2 Configuración del perfil de usuario . . . 89
5.3 Propiedades del proyecto . . . 90
5.3.1 Propiedades Generales . . . 90
5.3.2 Propiedades de metadatos . . . 94
5.3.3 Ver configuraciones . . . 94
5.3.4 Propiedades CRS . . . 94
5.3.5 Propiedades de Transformaciones . . . 95
5.3.6 Propiedades de Estilos . . . 95
5.3.7 Propiedades de los colores . . . 97
5.3.8 Propiedades de Fuentes de Datos . . . 98
5.3.9 Propiedades de relación . . . 100
5.3.10 Propiedades de variables . . . 100
5.3.11 Propiedades de Macros . . . 100
5.3.12 Propiedades de servidor QGIS . . . 101
5.3.13 Propiedades Temporales . . . 103
5.3.14 Propiedades de elevación . . . 103
5.3.15 Propiedades de los sensores . . . 105
5.4 Personalización . . . 107
5.5 Atajos de teclado . . . 109
5.6 Ejecutando QGIS con ajustes avanzados . . . 111
5.6.1 La línea de comandos y variables de entorno . . . 111
5.6.2 Implementar QGIS dentro de una organización . . . 116

6 Trabajar con Proyecciones **119**
6.1 Vista general de la ayuda de proyección . . . 119
6.2 Sistemas de Coordenadas de Capas . . . 119
6.3 Sistema de Coordenadas de Referencia del Proyecto . . . 121
6.4 Selector del Sistema de Coordenadas de Referencia . . . 123
6.5 Sistema de referencia de coordenadas personalizada . . . 124
6.5.1 Integrar una transformación NTv2 en QGIS . . . 125
6.6 Transformaciones de Datum . . . 126

7 Visualización de mapas **129**
7.1 Vista de mapa 2D . . . 129
7.1.1 Explorando la vista de mapa . . . 130
7.1.2 Control de la representación de mapas . . . 131
7.1.3 Control basado en el tiempo en el lienzo del mapa . . . 132
7.1.4 El panel del controlador de elevación . . . 134
7.1.5 Marcar extensiones en el mapa . . . 135
7.1.6 Decorar el mapa . . . 137
7.1.7 Herramientas de anotaciones . . . 144
7.1.8 Mediciones . . . 150
7.1.9 Ajustando vistas de mapa adicionales . . . 153
7.1.10 Exportando la vista de mapa . . . 154
7.2 Vista de Mapa 3D . . . 156
7.2.1 Configuración de la escena . . . 158
7.2.2 Opciones de Navegación . . . 166
7.2.3 Creando una animación . . . 167
7.2.4 Capas vectoriales 3D . . . 168
7.3 Vista del perfil de elevación . . . 168
7.3.1 La interfaz . . . 168
7.3.2 Creando un perfil de elevación . . . 170
7.3.3 Interactuar con el perfil Vistas . . . 172
7.3.4 Exportar desde el perfil de elevación . . . 173

8 Herramientas generales **175**
8.1 Ayuda de contexto . . . 175
8.2 Paneles . . . 175
8.2.1 Panel de capas . . . 175
8.2.2 Panel de Estilizado de Capa . . . 183
8.2.3 Panel de orden de capa . . . 186
8.2.4 Panel de información general . . . 186
8.2.5 Panel de mensajes de registro . . . 187
8.2.6 Panel de deshacer/rehacer . . . 187
8.2.7 Panel de resumen estadístico . . . 187
8.2.8 Panel de herramientas de desarrollo/depuración . . . 188
8.3 Incorporación de capas de proyectos externos . . . 192
8.4 Interactuando con entidades . . . 193
8.4.1 Seleccionando objetos espaciales . . . 193
8.4.2 Identificando entidades . . . 196
8.5 Guardar y compartir Propiedades . . . 200
8.5.1 Administrando Estilos Personalizados . . . 200
8.5.2 Almacenar estilos en un archivo o una base de datos . . . 202
8.5.3 Archivo de definición de capa . . . 204
8.6 Documentando tus datos . . . 204
8.6.1 Metadatos . . . 204
8.6.2 Notas de capa . . . 205
8.7 Almacenando valores en variables . . . 206
8.8 Widgets comunes . . . 208
8.8.1 Selector de color . . . 208
8.8.2 Widget Símbolo . . . 212

8.8.3 Selector de archivos remoto o incrustado . . . 212
8.8.4 Selector de escala de visibilidad . . . 213
8.8.5 Selector de extensión espacial . . . 213
8.8.6 Selector de Fuente . . . 214
8.8.7 Selector de Unidad . . . 215
8.8.8 Formato de Números . . . 216
8.8.9 Modos de Mezcla . . . 217
8.8.10 Configuración de anulación definida por datos . . . 219
8.8.11 Autenticación . . . 222

9 Sube de nivel con Expresiones **225**
9.1 Expresiones . . . 225
9.1.1 El constructor de cadena de Expresión . . . 225
9.1.2 Editor de Funciones . . . 230
9.2 Lista de funciones . . . 233
9.2.1 Agregar Funciones . . . 233
9.2.2 Funciones Arreglo . . . 243
9.2.3 Funciones de Color . . . 255
9.2.4 Funciones Condicionales . . . 264
9.2.5 Funciones de conversión . . . 267
9.2.6 CRS Functions . . . 273
9.2.7 Funciones personalizadas . . . 273
9.2.8 Funciones de Fecha y Hora . . . 273
9.2.9 Campos y Valores . . . 284
9.2.10 Funciones de Archivos y Rutas . . . 284
9.2.11 Funciones de formulario . . . 287
9.2.12 Funciones Concordancia aproximada . . . 288
9.2.13 Funciones Generales . . . 289
9.2.14 Funciones de Geometría . . . 291
9.2.15 Funciones de diseño . . . 367
9.2.16 Capas de mapa . . . 368
9.2.17 Funciones de Mapas . . . 371
9.2.18 Funciones Matemáticas . . . 375
9.2.19 Funciones de mallas . . . 383
9.2.20 Operadores . . . 385
9.2.21 Funciones de Procesamiento . . . 394
9.2.22 Funciones Ráster . . . 394
9.2.23 Funciones de Registro y Atributos . . . 396
9.2.24 Relaciones . . . 405
9.2.25 Funciones de sensores . . . 405
9.2.26 Funciones de cadena . . . 405
9.2.27 Expresiones de Usuario . . . 417
9.2.28 Variables . . . 417
9.2.29 Funciones recientes . . . 422

10 La librería Style **423**
10.1 El administrador de Estilo . . . 423
10.1.1 El diálogo del Administrador de Estilo . . . 423
10.1.2 Estableciendo una Rampa de Color . . . 429
10.1.3 Creación de una forma de parche de leyenda . . . 431
10.2 EL selector de símbolo . . . 433
10.2.1 El árbol de símbolo de capa . . . 434
10.2.2 Configurando un símbolo . . . 435
10.3 Ajustando una etiqueta . . . 454
10.3.1 Formateando la etqueta de texto . . . 456
10.3.2 Configurando la interacción con etiquetas . . . 466
10.4 Creando Símbolos 3D . . . 477
10.4.1 Capas de puntos . . . 478

10.4.2 Capas lineales . . . 479
10.4.3 Capas Poligonales . . . 480
10.4.4 Sombreando la textura . . . 482
10.4.5 Ejemplo de Aplicación . . . 482

11 Administrar el origen de datos **483**
11.1 Abriendo Datos . . . 483
11.1.1 El panel Navegador . . . 485
11.1.2 El gestor de Base de Datos . . . 488
11.1.3 Herramientas de carga basadas en proveedores . . . 489
11.1.4 Panel de búsqueda de metadatos de capas . . . 510
11.1.5 Formato personalizados QGIS . . . 511
11.1.6 QLR - Archivo de Definición de Capa QGIS . . . 511
11.1.7 Conectar con el servicios web . . . 512
11.2 Creando capas . . . 520
11.2.1 Creando nuevas capas vectoriales . . . 521
11.2.2 Creando nuevas capas desde una capa existente . . . 530
11.2.3 Creando nuevos archivos DXF . . . 535
11.2.4 Creando nuevas capas desde el portapapeles . . . 536
11.2.5 Creating SQL Query Layers . . . 537
11.2.6 Creando capas virtuales . . . 540
11.3 Georreferenciador . . . 543
11.3.1 Capa ráster de georreferenciación . . . 543
11.3.2 Capa vectorial de georreferenciación . . . 549
11.3.3 Mostrar y adaptar las propiedades de las capas . . . 550
11.3.4 Configurar el georreferenciador . . . 550
11.4 Explorando campos y formatos de datos . . . 551
11.4.1 Datos Raster . . . 551
11.4.2 Datos Vectoriales . . . 551
11.4.3 Capas que cruzan los 180° de longitud . . . 559

12 Trabajar con Datos Vectoriales **561**
12.1 El Diálogo de las Propiedades del Vector . . . 561
12.1.1 Propiedades de información . . . 562
12.1.2 Propiedades de fuente . . . 563
12.1.3 Propiedades de simbología . . . 566
12.1.4 Propiedades de etiquetas . . . 589
12.1.5 Propiedades de diagrama . . . 601
12.1.6 Propiedades de Máscaras . . . 606
12.1.7 Propiedades de visor 3D . . . 607
12.1.8 Propiedades de campos . . . 609
12.1.9 Formulario de propiedades de campo . . . 610
12.1.10 Propiedades de unión . . . 622
12.1.11 Propiedades de almacenamiento auxiliar . . . 622
12.1.12 Propiedades de acciones . . . 631
12.1.13 Propiedades a mostrar . . . 637
12.1.14 Propiedades de representación . . . 640
12.1.15 Propiedades Temporales . . . 642
12.1.16 Propiedades de variables . . . 643
12.1.17 Propiedades de elevación . . . 643
12.1.18 Propiedades de metadatos . . . 645
12.1.19 Propiedades de dependencias . . . 645
12.1.20 Propiedades de la leyenda . . . 645
12.1.21 Propiedades de servidor QGIS . . . 646
12.1.22 Propiedades de Digitalizado . . . 648
12.2 Trabajar con la tabla de atributos . . . 650
12.2.1 Prefacio: Tablas espaciales y no espaciales . . . 650
12.2.2 Introducción a la interfaz de la tabla de atributos . . . 651

12.2.3 Interactuar con entidades en una tabla de atributos . . . 656
12.2.4 Editar valores de atributo . . . 660
12.2.5 Exploración de los atributos de los objetos espaciales mediante la herramienta de Identificación . . . 664
12.3 Editando . . . 666
12.3.1 Configuración de la tolerancia de ajuste y el radio de búsqueda . . . 667
12.3.2 Opciones de ajuste y digitalización . . . 667
12.3.3 Edición topológica . . . 670
12.3.4 Digitalizando una capa existente . . . 672
12.3.5 Digitalización avanzada . . . 684
12.3.6 Digitalización de formas . . . 694
12.3.7 El panel de Digitalización Avanzada . . . 697
12.3.8 El modificador de capa de procesamiento in situ . . . 706
12.4 Conexión y edición de datos entre capas . . . 707
12.4.1 Unir objetos espaciales entre dos capas . . . 707
12.4.2 Establecer relaciones entre varias capas . . . 709
12.4.3 Storing and fetching an external resource . . . 724

13 Trabajar con Datos Raster **727**
13.1 Dialogo de Propiedades Ráster . . . 727
13.1.1 Propiedades de información . . . 728
13.1.2 Propiedades de fuente . . . 728
13.1.3 Propiedades de simbología . . . 729
13.1.4 Propiedades de Transparencia . . . 739
13.1.5 Labels Properties . . . 741
13.1.6 Propiedades de Histograma . . . 742
13.1.7 Propiedades de representación . . . 743
13.1.8 Propiedades Temporales . . . 744
13.1.9 Propiedades de Pirámides . . . 745
13.1.10 Propiedades de elevación . . . 746
13.1.11 Propiedades de metadatos . . . 747
13.1.12 Propiedades de la leyenda . . . 748
13.1.13 Propiedades a mostrar . . . 749
13.1.14 Propiedades de servidor QGIS . . . 750
13.1.15 Attribute Tables . . . 751
13.1.16 Identificar celdas ráster . . . 755
13.2 Análisis raster . . . 755
13.2.1 Calculadora Ráster . . . 755

14 Trabajando con Malla de Datos **759**
14.1 Qué es una malla? . . . 759
14.2 Formatos soportados . . . 761
14.3 Propiedades del Conjunto de Datos tipo Malla . . . 762
14.3.1 Propiedades de información . . . 763
14.3.2 Propiedades de fuente . . . 764
14.3.3 Propiedades de simbología . . . 765
14.3.4 Propiedades de etiquetas . . . 771
14.3.5 Propiedades de visor 3D . . . 772
14.3.6 Propiedades de representación . . . 774
14.3.7 Propiedades Temporales . . . 775
14.3.8 Propiedades de elevación . . . 776
14.3.9 Propiedades de metadatos . . . 777
14.4 Exploring mesh datasets . . . 777
14.5 Editando la capa de malla . . . 778
14.5.1 Visión general de las herramientas de digitalización de malla . . . 779
14.5.2 Explorando la asignación lógica del valor Z . . . 779
14.5.3 Seleccionar elementos de malla . . . 781
14.5.4 Modificando elementos de la malla . . . 782

14.5.5 Volviendo a indexar mallas . . . 787
14.6 Calculadora de malla . . . 788

15 Trabajando con Teselas Vectoriales **791**
15.1 ¿Qué son las teselas vectoriales? . . . 791
15.2 Formatos Soportados . . . 792
15.3 Propiedades del conjunto de datos de teselas vectoriales . . . 793
15.3.1 Propiedades de información . . . 793
15.3.2 Propiedades de fuente . . . 793
15.3.3 Propiedades de simbología y etiquetas . . . 794
15.3.4 Propiedades de representación . . . 795
15.3.5 Propiedades de metadatos . . . 796
15.4 QGIS Server Properties . . . 796

16 Trabajar con Puntos Nubes **797**
16.1 Introducción a las Nubes de Puntos . . . 797
16.2 Propiedades de las nubes de puntos . . . 798
16.2.1 Propiedades de información . . . 798
16.2.2 Propiedades de fuente . . . 799
16.2.3 Propiedades de simbología . . . 802
16.2.4 Propiedades de visor 3D . . . 810
16.2.5 Propiedades de representación . . . 812
16.2.6 Propiedades de elevación . . . 812
16.2.7 Propiedades de metadatos . . . 814
16.2.8 Propiedades estadísticas . . . 814
16.3 Nube de puntos virtual . . . 815

17 Trabajar con teselas 3D **819**
17.1 ¿Qué son las teselas 3D? . . . 819
17.1.1 State of 3D Tiles Support . . . 820
17.2 Propiedades de las teselas 3D . . . 821
17.2.1 Propiedades de información . . . 821
17.2.2 Propiedades de fuente . . . 821
17.2.3 Propiedades de simbología . . . 822
17.2.4 Propiedades de visor 3D . . . 823
17.2.5 Propiedades de representación . . . 823
17.2.6 Propiedades de elevación . . . 824
17.2.7 Propiedades de metadatos . . . 824

18 Diseñando los mapas **825**
18.1 Resumen de la Composición de Impresión . . . 825
18.1.1 Sesión de muestra para principiantes . . . 825
18.1.2 El Administrador de Composiciones . . . 826
18.1.3 Menús, herramientas y paneles de la composición de impresión . . . 828
18.2 Elementos del Layout . . . 842
18.2.1 Opciones de Elementos comunes de Composición . . . 842
18.2.2 El elemento del mapa . . . 848
18.2.3 El Elemento Mapa 3D . . . 859
18.2.4 El elemento etiqueta . . . 860
18.2.5 El elemento leyenda . . . 865
18.2.6 El elemento de barra de escala . . . 874
18.2.7 Los elementos de la tabla . . . 879
18.2.8 Los elementos de marcador, imagen y flecha de norte . . . 888
18.2.9 Elemento del perfil de elevación . . . 893
18.2.10 El elemento del marco HTML . . . 894
18.2.11 Los elementos de Forma . . . 897
18.3 Crear salida . . . 900
18.3.1 Configuración de exportación . . . 901
18.3.2 Exportar como imagen . . . 901

18.3.3 Exportar como SVG . . . 902
18.3.4 Exportar como PDF . . . 904
18.3.5 Generar un Atlas . . . 905
18.4 Creando un informe . . . 911
18.4.1 ¿Que es? . . . 911
18.4.2 Empezando . . . 911
18.4.3 Espacio de trabajo de Diseñador de Informes . . . 913
18.4.4 Configuración de exportación . . . 927

19 Trabajando con protocolos OGC / ISO **929**
19.1 Cliente WMS/WMTS . . . 930
19.1.1 Información general de la implementación WMS . . . 930
19.1.2 Información general de la implementación WMTS . . . 930
19.1.3 Seleccionar servidor WMS/WMTS . . . 931
19.1.4 Cargando capas WMS/WMTS . . . 933
19.1.5 Conjunto de teselas . . . 935
19.1.6 Utilizar la herramienta de Identificar objetos espaciales . . . 936
19.1.7 Ver propiedades . . . 936
19.1.8 Mostrar el gráfico de la leyenda de WMS en la tabla de contenido y diseño . . . 940
19.2 WCT Cliente . . . 941
19.3 Cliente WFS y WFS-T . . . 943
19.4 SensorThings . . . 946
19.4.1 Configuración de la conexión . . . 946
19.4.2 Cargando datos de SensorThings . . . 948
19.4.3 Trabajar con una capa vectorial de SensorThings . . . 949
19.5 STAC (SpatioTemporal Asset Catalogs) . . . 950
19.5.1 Configuración de la conexión . . . 950
19.5.2 Browsing STAC Catalogs . . . 951
19.5.3 Filtering and Searching STAC Items . . . 952
19.6 ArcGIS REST Server . . . 954

20 Trabajar con datos GPS **955**
20.1 Introducción a los datos GNSS/GPS . . . 955
20.1.1 ¿Qué es GPS? . . . 955
20.1.2 Transferencia o carga de datos GPS . . . 956
20.2 Seguimiento de GPS en Vivo . . . 958
20.2.1 Barra de herramientas GPS . . . 958
20.2.2 Panel de información GPS . . . 960
20.2.3 Conéctese a un GPS Bluetooth para seguimiento en vivo . . . 962
20.2.4 Ejemplos de conexión de dispositivos GPS en QGIS . . . 962

21 Sistema de autenticación **965**
21.1 Vista general del sistema de Autenticación . . . 965
21.1.1 Base de datos de autenticación . . . 966
21.1.2 Bases de datos de autenticación personalizadas . . . 966
21.1.3 Contraseña maestra . . . 967
21.1.4 Configuraciones de autenticación . . . 968
21.1.5 Métodos de autenticación . . . 971
21.1.6 Utilidades de contraseña maestra y configuración de autenticación . . . 978
21.1.7 Usando configuraciones de autenticación . . . 979
21.1.8 Python bindings . . . 979
21.2 Flujos de trabajo de autenticación de usuarios . . . 980
21.2.1 Autenticación HTTP(S) . . . 980
21.2.2 Base de datos de autenticación . . . 981
21.2.3 Autenticación PKI . . . 982
21.2.4 Manejar capas defectuosas . . . 989
21.2.5 Cambiar el ID de configuración de autenticación . . . 990
21.2.6 Soporte de Servidor QGIS . . . 990
21.2.7 excepciones de servidor SSL . . . 991

22 Integracion GRASS SIG **995**
22.1 Conjuntos de datos demostración . . . 995
22.2 Cargar capas ráster y vectorial de GRASS . . . 995
22.3 Importar datos dentro de una UBICACIÓN DE GRASS mediante arrastrar y soltar . . . 996
22.4 Administrar datos de GRASS en el navegador QGIS . . . 996
22.5 Opciones GRASS . . . 996
22.6 Iniciar el complemento GRASS . . . 997
22.7 Abrir directorio de mapas . . . 997
22.8 LOCALIZACIÓN y DIRECTORIO DE MAPA GRASS . . . 997
22.9 Importar datos dentro de una LOCALIZACIÓN DE GRASS . . . 998
22.9.1 Crear una nueva LOCALIZACIÓN GRASS . . . 999
22.9.2 Añadir un nuevo DIRECTORIO DE MAPA . . . 1000
22.10 El modelo de datos vectoriales de GRASS . . . 1001
22.11 Crear una nueva capa vectorial GRASS . . . 1001
22.12 Digitalizar y editar una capa vectorial GRASS . . . 1002
22.13 La herramienta de región GRASS . . . 1004
22.14 La caja de herramientas GRASS . . . 1005
22.14.1 Trabajando con módulos GRASS . . . 1005
22.14.2 Ejemplos del módulo GRASS . . . 1008
22.14.3 Personalizar la caja de herramientas GRASS . . . 1010

23 Entorno de trabajo de procesamiento de QGIS **1011**
23.1 Introducción . . . 1011
23.2 Configurando el marco de procesamiento . . . 1014
23.2.1 General . . . 1014
23.2.2 Menús . . . 1016
23.2.3 Modelos y scripts . . . 1016
23.2.4 Proveedores . . . 1016
23.3 La caja de Herramientas . . . 1017
23.3.1 El cuadro de diálogo de algoritmo . . . 1019
23.3.2 Resultados generados por algoritmos . . . 1027
23.4 El administrador del historial . . . 1028
23.4.1 El historial del procesamiento . . . 1028
23.4.2 El registro de procesamiento . . . 1029
23.5 El diseñador de modelos . . . 1030
23.5.1 La interfaz del diseñador de modelos . . . 1030
23.5.2 Creando un modelo . . . 1033
23.5.3 Guardar y cargar modelos. . . . 1042
23.5.4 Editar un modelo. . . . 1043
23.6 La interfaz de procesamiento por lote . . . 1044
23.6.1 Introducción . . . 1044
23.6.2 La tabla de parámetros . . . 1045
23.6.3 Llenado de la tabla de parámetros . . . 1046
23.6.4 Ejecutar el proceso por lotes . . . 1047
23.7 Utilizar algoritmos de procesamiento desde la consola . . . 1047
23.7.1 Invocando algoritmos desde la consola de Python . . . 1048
23.7.2 Crear scripts y ejecurarlos desde le Caja de Herramientas . . . 1053
23.7.3 Pre y post-ejecución de la secuencia de comandos hooks . . . 1055
23.8 Usar procesamiento desde la línea de comando . . . 1056
23.9 Escribir nuevos algoritmos de procesamiento como scripts de Python . . . 1058
23.9.1 Extendiendo QgsProcessingAlgorithm . . . 1058
23.9.2 El decorador @alg . . . 1062
23.9.3 Manejo de salida de algoritmo . . . 1064
23.9.4 Comunicarse con el usuario . . . 1064
23.9.5 Documentación de las secuencias de comandos . . . 1064
23.9.6 Banderas . . . 1065
23.9.7 Las mejores practivas al escribir scripts de algoritmos . . . 1065
23.10 Tipos de entrada y salida para algoritmos de procesamiento . . . 1065

23.10.1 Tipos de entrada . . . 1065
23.10.2 Tipos de salida . . . 1068
23.11 Configurar aplicaciones externas . . . 1068
23.11.1 Aclaración para los usuarios de Windows . . . 1069
23.11.2 Aclaración respecto a los formatos de archivos . . . 1069
23.11.3 Nota referente a las seleccion de capas vectoriales . . . 1069
23.11.4 Uso de proveedores externos . . . 1069

24 Proveedor de procesos y algoritmos 1081
24.1 Proveedor de algoritmos QGIS . . . 1081
24.1.1 Teselas 3D . . . 1081
24.1.2 Cartografía . . . 1083
24.1.3 Check Geometry . . . 1105
24.1.4 Base de datos . . . 1150
24.1.5 Herramientas de Archivo . . . 1158
24.1.6 Fix Geometry . . . 1161
24.1.7 GPS . . . 1186
24.1.8 Interpolación . . . 1190
24.1.9 Herramientas de capa . . . 1203
24.1.10 Malla . . . 1206
24.1.11 Metadata tools . . . 1220
24.1.12 Herramientas del Modelador . . . 1225
24.1.13 Análisis de Red . . . 1235
24.1.14 Gráficos . . . 1250
24.1.15 Conversión de nubes de puntos . . . 1264
24.1.16 Gestión de datos de nubes de puntos . . . 1271
24.1.17 Extracción de nubes de puntos . . . 1285
24.1.18 Análisis raster . . . 1291
24.1.19 Creación de ráster . . . 1359
24.1.20 Análisis de terreno ráster . . . 1377
24.1.21 Herramientas de ráster . . . 1390
24.1.22 Análisis vectorial . . . 1403
24.1.23 Cobertura vectorial . . . 1430
24.1.24 Creación vectorial . . . 1433
24.1.25 Vectorial general . . . 1465
24.1.26 Geometría vectorial . . . 1507
24.1.27 Superposición vectorial . . . 1633
24.1.28 Selección vectorial . . . 1653
24.1.29 Tabla vectorial . . . 1671
24.1.30 Teselas Vectoriales . . . 1689
24.2 Proveedor de algoritmos GDAL . . . 1692
24.2.1 Análisis raster . . . 1693
24.2.2 Conversión raster . . . 1723
24.2.3 Extracción ráster . . . 1732
24.2.4 Miscelánea Ráster . . . 1741
24.2.5 Proyecciones ráster . . . 1762
24.2.6 Conversión vectorial . . . 1767
24.2.7 Geoprocesamiento vectorial . . . 1773
24.2.8 Vector misceláneo . . . 1783

25 Complementos 1797
25.1 Complementos de QGIS . . . 1797
25.1.1 Complementos base y externos . . . 1797
25.1.2 El diálogo de complementos . . . 1798
25.2 Usar complementos del núcleo de QGIS . . . 1801
25.2.1 Complemento de Administración de BBDD . . . 1802
25.2.2 Complemento Verificador de Geometría . . . 1806
25.2.3 Cliente de Catálogo de metasearch . . . 1810

25.2.4 Complemento de Edición Fuera de Línea . . . 1817
25.2.5 Complemento Comprobador de Topología . . . 1819
25.3 Consola Python de QGIS . . . 1822
25.3.1 La consola interactiva . . . 1822
25.3.2 El Editor de código . . . 1824

26 Apéndices **1827**
26.1 Apéndice C: Formatos de Archivo de QGIS . . . 1827
26.1.1 QGS/QGZ - El Formato de Archivo de Proyecto de QGIS . . . 1827
26.1.2 QLR - El archivo de Definición de Capa de QGIS . . . 1829
26.1.3 QML: el formato de archivo de estilo QGIS . . . 1830
26.2 Apéndice D: sintaxis del script QGIS R . . . 1831
26.2.1 Entradas . . . 1832
26.2.2 Salidas . . . 1832
26.2.3 Resumen de sintaxis para scripts QGIS R . . . 1832
26.2.4 Ejemplos . . . 1834
26.3 Apéndice E: Conexiones de red de la aplicación QGIS . . . 1836

27 Referencias bibliográficas y web **1839**

CAPÍTULO 1

Comenzando

Este capítulo proporciona un rápido resumen para instalar QGIS, descargar los datos de muestra QGIS, y ejecutar una primera sesión visualizando datos ráster y vectoriales.

1.1 Instalando QGIS

El proyecto QGIS proporciona diferentes modos para instalar QGIS dependiendo de tu plataforma.

1.1.1 Instalación desde binarios

Los instaladores estándar están disponibles para MS Windows y macOS. Se proporcionan paquetes binarios (rpm y deb) o repositorios de software para muchos tipos de GNU/Linux.

Para obtener más información e instrucciones para su sistema operativo, consulte https://qgis.org/download/.

1.1.2 Instalando desde la fuente

Si necesita compilar QGIS desde la fuente, consulte las instrucciones de instalación. Se distribuyen con el código fuente de QGIS en un archivo llamado `INSTALL`. También puede encontrarlos en línea en https://github.com/qgis/QGIS/blob/release-3_44/INSTALL.md.

Si desea crear una versión particular y no la versión en desarrollo, debe reemplazar `master` con la rama de lanzamiento (comúnmente en la forma `release-X_Y`) en el enlace mencionado anteriormente (las instrucciones de instalación pueden diferir).

1.1.3 Instalación en medios externos

Es posible instalar QGIS (con todos los complementos y configuraciones) en una unidad flash. Esto se logra definiendo una opción *–profiles-path* que sobreescriba la ruta predeterminada *user profile* y fuerce **QSettings** para usar este directorio, también. Mira la sección *Ajustes del sistema* para información adicional.

1.1.4 Descargando datos de muestra

Esta guía del usuario contiene ejemplos basados en el conjunto de datos de muestra de QGIS (también llamado `Alaska dataset`). Descargue los datos de muestra de https://github.com/qgis/QGIS-Sample-Data/archive/master.zip y descomprima el archivo en cualquier ubicación conveniente de su sistema.

El conjunto de datos de Alaska incluye todos los datos SIG que se utilizan para los ejemplos y las capturas de pantalla en esta guía del usuario; También incluye una pequeña base de datos GRASS. La proyección para los conjuntos de datos de muestra QGIS es Alaska Albers Equal Area con unidades de pies. El código EPSG es 2964.

```
PROJCS["Albers Equal Area",
GEOGCS["NAD27",
DATUM["North_American_Datum_1927",
SPHEROID["Clarke 1866",6378206.4,294.978698213898,
AUTHORITY["EPSG","7008"]],
TOWGS84[-3,142,183,0,0,0,0],
AUTHORITY["EPSG","6267"]],
PRIMEM["Greenwich",0,
AUTHORITY["EPSG","8901"]],
UNIT["degree",0.0174532925199433,
AUTHORITY["EPSG","9108"]],
AUTHORITY["EPSG","4267"]],
PROJECTION["Albers_Conic_Equal_Area"],
PARAMETER["standard_parallel_1",55],
PARAMETER["standard_parallel_2",65],
PARAMETER["latitude_of_center",50],
PARAMETER["longitude_of_center",-154],
PARAMETER["false_easting",0],
PARAMETER["false_northing",0],
UNIT["us_survey_feet",0.3048006096012192]]
```

Si pretende usar QGIS como interfaz gráfica para GRASS, puede encontrar una selección de ubicaciones de muestra (p. ej. Spearfish o Dakota del Sur) en la página web oficial GRASS GIS, https://grass.osgeo.org/download/data/.

1.2 Iniciando y deteniendo QGIS

QGIS se puede iniciar como cualquier otra aplicación en su computadora. Esto significa que puede iniciar QGIS de la siguiente manera:

- usando menu Aplicaciones, el menú Inicio, o el Dock
- doble clic el ícono en su carpeta de Aplicaciones o atajo de escritorio
- haciendo doble clic en un archivo de proyecto QGIS existente (con extensión `.qgz` o `.qgs`). Tenga en cuenta que esto también abrirá el proyecto.
- escribiebdo `qgis` en un símbolo del sistema (suponiendo que QGIS se agregó en su RUTA o que está en su carpeta de instalación)

Para detener QGIS, use:

- la opción del menú *Proyecto* ► *Salir de QGIS* o use el atajo `Ctrl+Q`
- *QGIS* ► *Salir de QGIS*, o use el atajo `Cmd+Q`

- o use el aspa roja en la esquina superior derecha de la interfaz principal de la aplicación.

1.3 Sesión de muestra: carga de capas ráster y vectoriales

Ahora que ya tiene *QGIS instalado* y un *conjunto de datos de muestra* disponible, veremos una primera sesión de ejemplo. En este ejemplo, visualizaremos una capa ráster y una vectorial. Usaremos:

- la capa ráster `landcover` (`qgis_sample_data/raster/landcover.img`)
- y la capa vectorial `lakes` (`qgis_sample_data/gml/lakes.gml`)

Donde `qgis_sample_data` representa la ruta del conjunto de datos descomprimido.

1. Inicie QGIS como vió en *Iniciando y deteniendo QGIS*.
2. Los datos con los que vamos a trabajar están en `Albers Equal Area`, así que vamos a configurar el SRC del proyecto en consecuencia:
    1. Haga clic en el botón Seleccionar proyección en la parte inferior derecha de la interfaz de QGIS. Se abre el diálogo de propiedades del proyecto con la pestaña *SRC* activa.
    2. Type `2964` in the *Filter* text area.
    3. Seleccione la fila con el nombre SRC `NAD27 / Alaska Albers`.

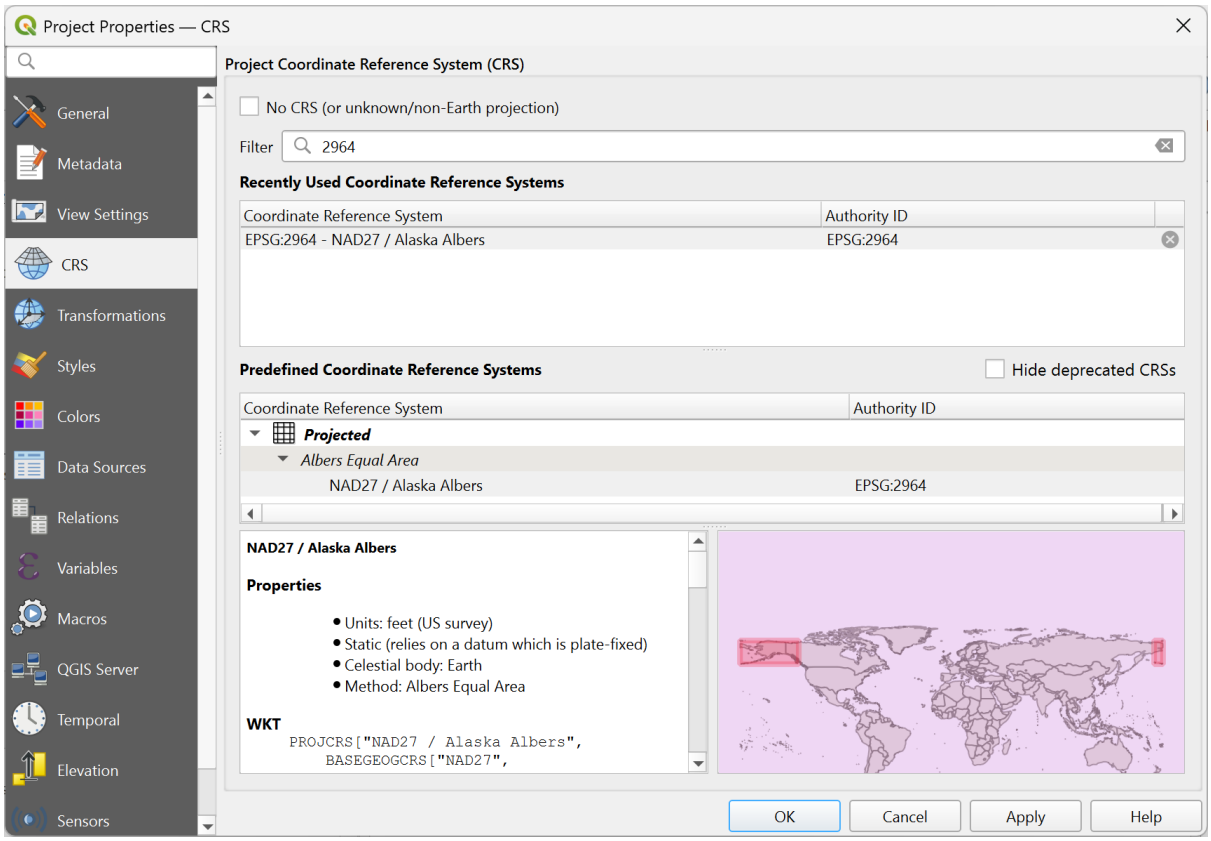


Figura 1.1: Seleccione el Sistema de Coordenadas de Referencia de los datos

4. Presiona *Aceptar*

> **Nota**
>
> Puede ignorar/cerrar por ahora el mensaje «ballpark transform» que podría aparecer.

3. Cargue los archivos en QGIS:

   1. Click en el icono Abrir administrador de fuente de datos. EL Administrador de Fuente de Datos debería abrirse en modo Navegador.
   2. Navegue al directorio `qgis_sample_data/raster/`
   3. Seleccione el archivo ERDAS IMG *landcover.img* y haga doble clic sobre él. La capa de cubierta terrestre se añade en segundo plano mientras la ventana del Administrador de fuentes de datos permanece abierta.

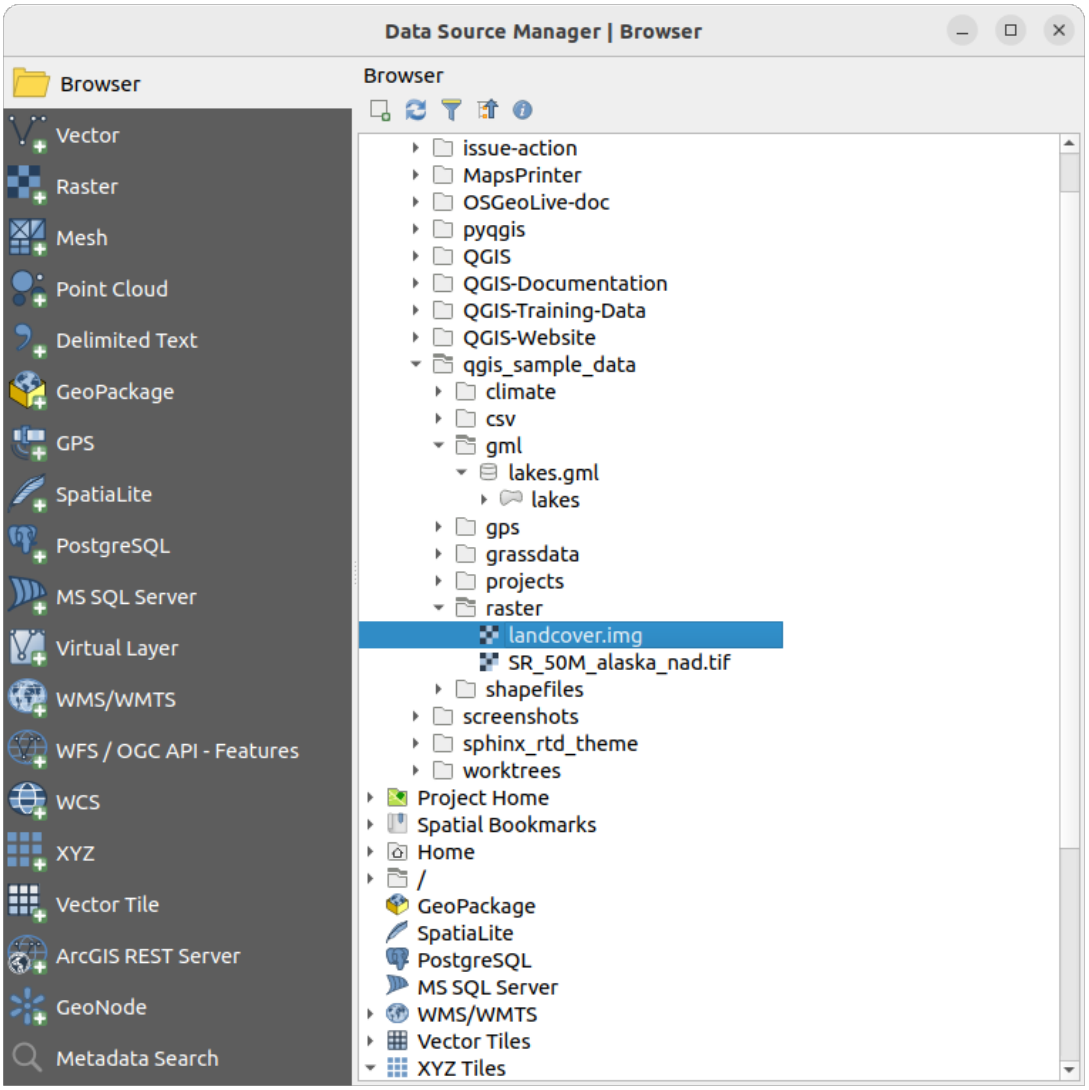


Figura 1.2: Añadiendo datos a un nuevo proyecto en QGIS

4. Para cargar los datos de los lagos, navegue hasta la carpeta `qgis_sample_data/gml/`, y arrastre y suelte el archivo *lakes.gml* sobre el diálogo principal de QGIS. (O simplemente haga doble clic como se mencionó anteriormente).

5. Se abre el diálogo *Seleccionar elementos para añadir*, escaneando el fichero. Esto se debe a que el formato de archivo `.gml` puede almacenar más de una capa a la vez.

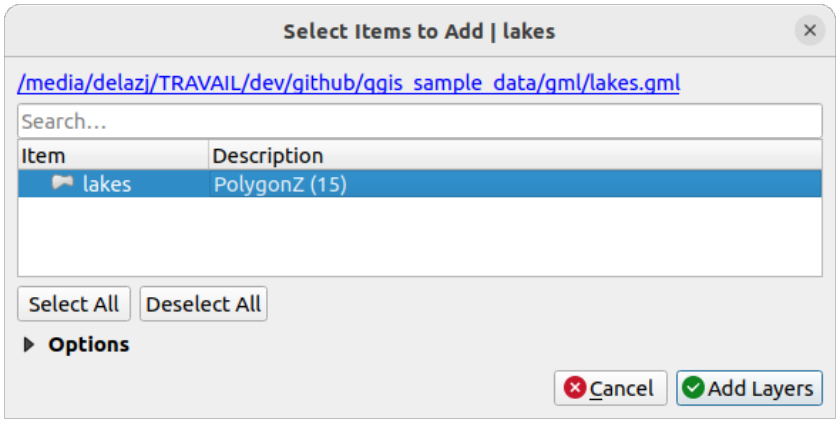


Figura 1.3: Seleccionar capas dentro de un archivo

6. En nuestro caso hay una única capa *lakes*. Selecciónela y pulse *Añadir capas*.
7. La capa se añade al panel *Capas*

4. Cierre la ventana del Administrador de Fuente de Datos

En el panel *Capas*, puede observar que la capa *lakes* muestra ¡La capa no tiene definido un SRC! junto a ella. Vamos a ajustarlo.

1. Haga clic en el icono . Se abre el cuadro de diálogo *Selector de sistema de referencia de coordenadas*.
2. Como se hizo anteriormente, busque y seleccione la entrada SRC *NAD27 / Alaska Albers*.
3. Haz clic en *Aceptar*.

Ahora tiene las dos capas disponibles en su proyecto en algunos colores aleatorios. Haga alguna personalización en la capa lagos.

1. Seleccione la herramienta Acercar zoom en la barra de herramientas *Navegación de mapas*
2. Amplíe un área con algunos lagos
3. Doble-click en la capa `lakes` en la leyenda del mapa para abrir el diálogo *Propiedades de capa*
4. Para cambiar el color de los lagos:
    1. Click en la pestaña *Simbología*
    2. Seleccione azul como color de relleno.

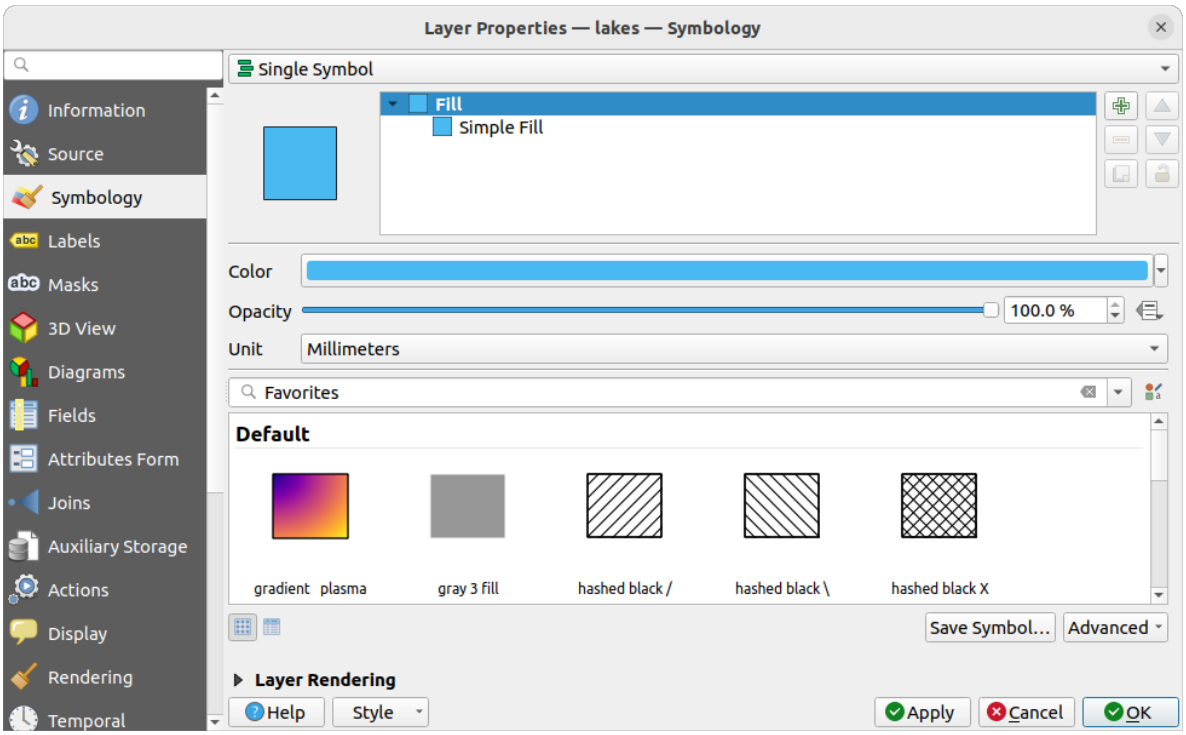


Figura 1.4: Seleccionando el color de los lagos

3. Presione *Aceptar*. Los lagos se muestran ahora en azul en la vista del mapa.

5. Para mostrar el nombre de los lagos:

   1. Reabra el cuadro de diálogo *Propiedades de capa* de la capa `lakes`
   2. Click en la pestaña *Etiquetas*
   3. Seleccione *Etiquetas sencillas* en el menú desplegable para habilitar el etiquetado.
   4. De la lista *Valor*, escoja el campo `NAMES`.

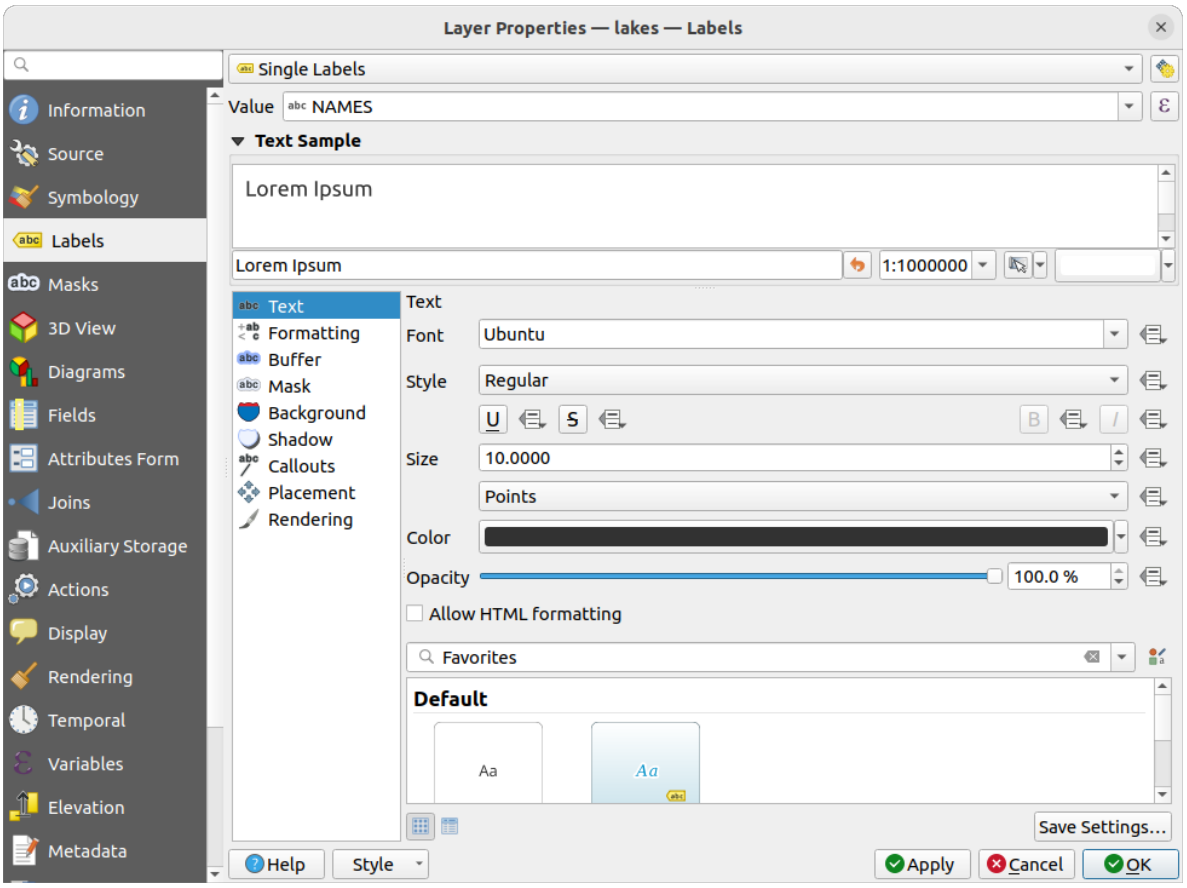


Figura 1.5: Mostrando los nombres de los lagos

5. Presione *Aplicar*. Los nombres se mostrarán ahora sobre los límites.

6. Puede mejorar la legibilidad de la etiquetas añadiendo una zona sombreada blanca a su alrededor

   1. Click en la pestaña *Buffer* en la lista a la izquierda
   2. Marque *Dibujar buffer de texto*
   3. Elija 3 como tamaño de buffer
   4. Click en *Aplicar*
   5. Compruebe si el resultado se ve bien, y actualice el valor si lo necesita.
   6. Finalmente haga click en *Aceptar* para cerrar el Cuadro de diálogo *Propiedades de capa* y aplicar los cambios.

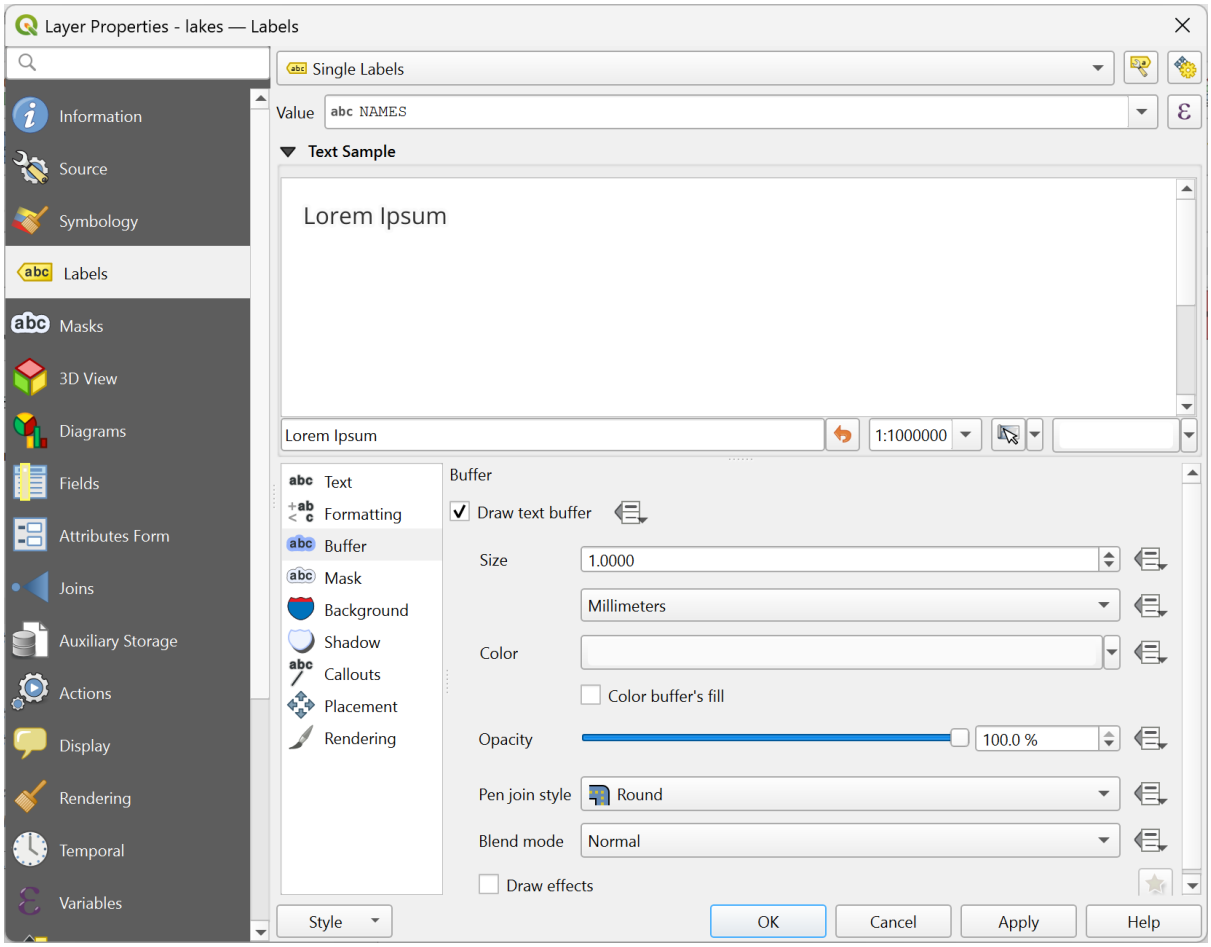


Figura 1.6: Mostrando Buffers alrededor de las etiquetas

Ahora agreguemos algunas decoraciones para darle forma al mapa y exportarlo fuera de QGIS:

1. Seleccione en el menú *Ver* ► *Ilustraciones* ► *Barra de escala*
2. En el diálogo que se abre, marque la opción *Activar barra de escala*
3. Personalice las opciones del diálogo como quiera
4. Presione *Aplicar*
5. Del mismo modo, desde el menú de decoración, agregue más elementos (flecha norte, derechos de autor…) a la vista del mapa con propiedades personalizadas.
6. Click en *Proyecto* ► *Importar/Exportar* ► *Exportar mapa a imagen …*
7. Presione *Guardar* en el cuadro de diálogo abierto
8. Seleccione una localización para el archivo, un formato y confirme presionando de nuevo en *Guardar*
9. Presione *Proyecto* ► *Guardar* para almacenar sus cambios como un archivo de proyecto `.qgz`.

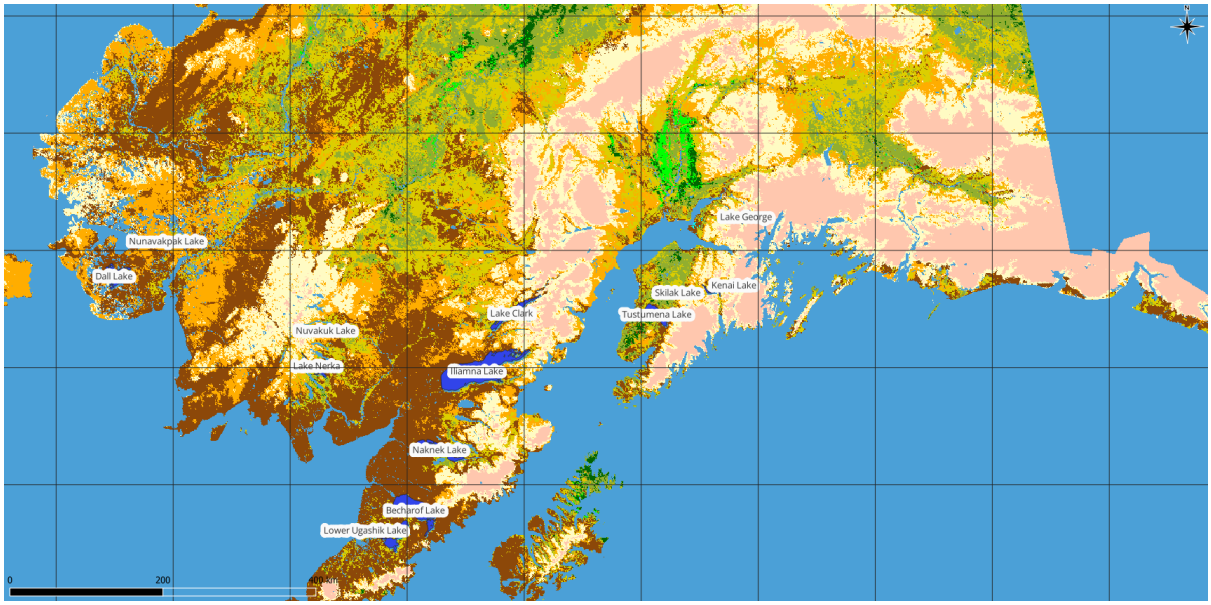


Figura 1.7: Mostrando mapa exportado con Decoraciones

¡Eso es! Puede ver lo fácil que es visualizar capas ráster y vectoriales en QGIS, configurarlas y generar su mapa en un formato de imagen que puede usar en otros software. Pasemos a aprender más sobre las funcionalidades disponibles, características y configuraciones, y cómo usarlas.

Nota

Para continuar aprendiendo QGIS mediante ejercicios paso a paso, siga las instrucciones del Manual de Aprendizaje QGIS.

CAPÍTULO 2

Trabajando con Archivos de Proyecto

2.1 Introducción a proyectos QGIS

El estado de su sesión de QGIS se llama proyecto. QGIS trabaja en un proyecto a la vez. Un ajuste puede ser específico del proyecto o un valor por defecto de la aplicación para los nuevos proyectos (ver sección *Opciones*). QGIS puede guardar el estado de su espacio de trabajo en un *QGIS archivo de proyecto* utilizando las opciones del menú *Proyecto* ► *Guardar* o *Proyecto* ► *Guardar como…*.

Nota

Si el proyecto ha sido modificado, el símbolo * aparecerá en la barra de título y QGIS, por defecto, le preguntará si desea guardar los cambios. Este comportamiento está controlado por la casilla de verificación de configuración *Solicitar guardar proyecto y fuentes de datos modificadas cuando sea necesario* en *Configuración* ► *Opciones* ► *General*.

Puede cargar proyectos existentes a través de *Proyecto* ► *Abrir…*, *Proyecto* ► *Nuevo desde plantilla* o *Proyecto* ► *Abrir reciente* ►.

Desde el *Panel del navegador* puede *Abrir proyecto*, *Extraer símbolos…*: abre el administrador de estilos para exportar símbolos a XML, añadirlos al estilo predeterminado o exportarlos como PNG/SVG. También puede acceder al cuadro de diálogo *Propiedades del archivo…*. El archivo del proyecto se puede expandir para ver sus capas. El menú contextual de una capa ofrece las mismas acciones que en cualquier otra parte del navegador.

Al iniciarse, se muestra una lista de *Plantillas de proyecto* y *Proyectos recientes*, con capturas de pantalla, nombres y rutas de archivos (para un máximo de diez proyectos). La lista *Proyectos Recientes* es útil para acceder a los proyectos utilizados recientemente. Haga doble clic en una entrada para abrir el proyecto o la plantilla de proyecto. Haga clic con el botón derecho en una entrada para *Anclar a la lista*, *Abrir directorio…* o *Eliminar de la lista*. Si ve su proyecto en la lista *Proyectos Recientes* pero no puede encontrarlo en su gestor de archivos utilice la opción *Abrir Directorio…* para ayudarle a localizar los proyectos que puedan faltar, haber sido movidos o renombrados. También puede ir a *Borrar Lista* si desea eliminar todos los proyectos de la lista de *Proyectos Recientes*. Si tiene proyectos anclados, la acción *Borrar lista* irá seguida de un cuadro de mensaje en el que se le preguntará si los proyectos anclados también deben eliminarse. También puede añadir una capa para crear un nuevo proyecto automáticamente. Las listas desaparecerán, dando paso al lienzo del mapa.

Si desea borrar su sesión y comenzar de nuevo, vaya a *Proyecto* ► *Nuevo*. Esto le pedirá que guarde el proyecto

existente si se han realizado cambios desde que se abrió o se guardó por última vez.

Cuando abre un proyecto nuevo, la barra de título mostrará `Proyecto sin título` hasta que lo guarde.

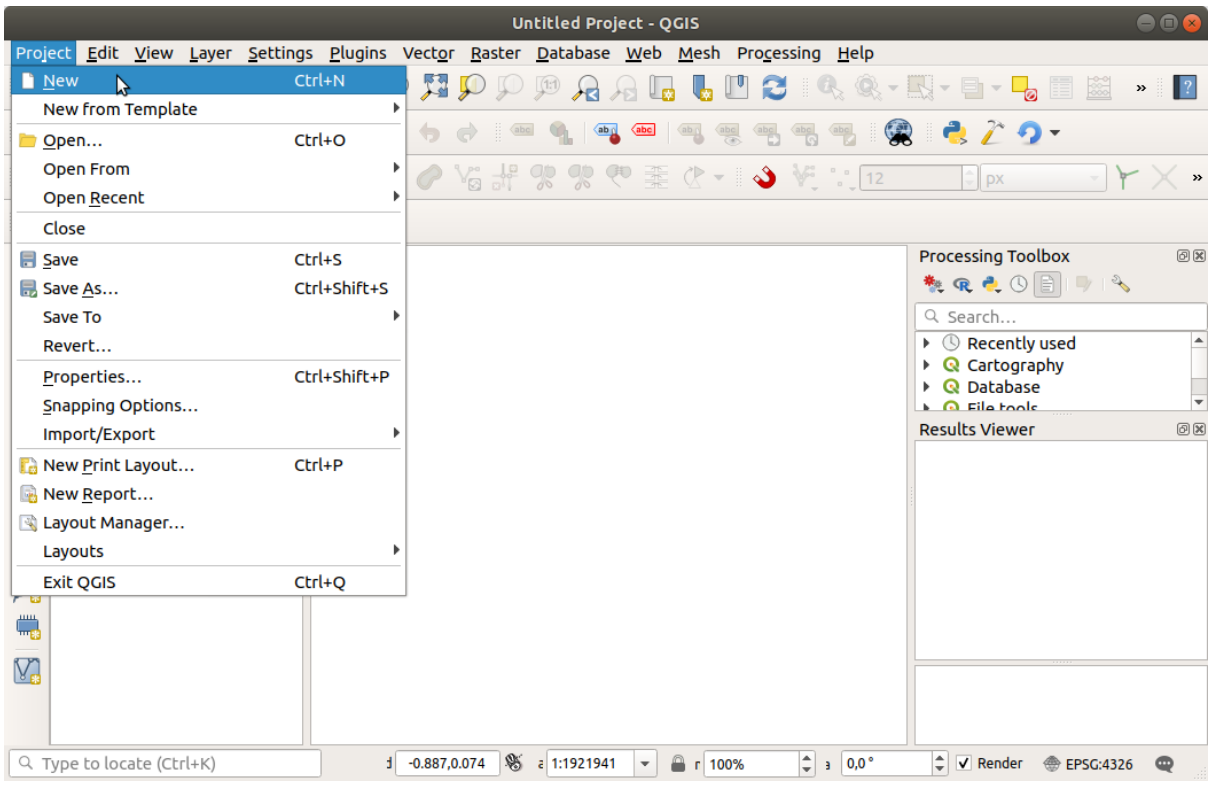


Figura 2.1: Empezando un nuevo Proyecto en QGIS

La información guardada en un archivo de proyecto incluye:

- Capas añadidas
- Que capas pueden ser consultadas
- Propiedades de capa, incluyendo simbología y estilos
- Notas de capa
- Vistas de mapas en 2D y 3D
- Proyección para cada vista del mapa
- Última extensión vista para cada mapa
- Diseños de impresión
- Elementos del diseño de impresión con ajustes
- Ajustes del diseño de impresión del atlas
- Ajustes de digitalización
- Relaciones de tabla
- Macros del proyecto
- Estilos predeterminados del proyecto
- Ajustes de complementos
- Ajustes del servidor QGIS desde la pestaña de ajustes de OWS en las propiedades del proyecto
- Consultas almacenadas en el Administrador de BBDD

El archivo del proyecto se guarda en formato XML (consulte *QGS/QGZ - El Formato de Archivo de Proyecto de QGIS*). Esto significa que es posible editar el archivo fuera de QGIS si sabe lo que está haciendo. El formato del archivo del proyecto se ha actualizado varias veces. Los archivos de proyecto de versiones anteriores de QGIS puede que no funcionen correctamente.

Nota

Por defecto, QGIS le advertirá sobre las diferencias de versión. Este comportamiento se controla en la pestaña *General* tab of *Settings* ► *Options* (☑ *Warn when opening a project file saved with an older version of QGIS*).

Cada vez que guardas un archivo de proyecto `.qgs` en QGIS, se crea una copia de seguridad del archivo en el mismo directorio que el archivo de proyecto, con la extensión `.qgs~`.

La extensión de los proyectos QGIS es `.qgs`, pero al guardar desde QGIS, por defecto se utiliza un formato comprimido con la extensión `.qgz`. El archivo `.qgs` está incrustado en el archivo `.qgz` (un archivo zip), junto con su base de datos SQLite asociada (`.qgd`) para los datos auxiliares .. Puede acceder a estos archivos descomprimiendo el archivo `.qgz`.

Nota

El mecanismo *Propiedades de almacenamiento auxiliar* hace que un proyecto comprimido sea particularmente útil, ya que incorpora datos auxiliares.

Los proyectos también pueden guardarse/cargarse en/desde una base de datos PostgreSQL, GeoPackage u Oracle utilizando los siguientes elementos del menú Proyecto:

- *Project* ► *Open from*
- *Project* ► *Save to*

Ambos elementos de menú tienen un submenú con una lista de implementaciones adicionales de almacenamiento de proyectos (PostgreSQL, GeoPackage y Oracle). Al hacer clic en la acción se abrirá un cuadro de diálogo para elegir una conexión y un proyecto GeoPackage, una conexión, un esquema y un proyecto PostgreSQL o una conexión, un propietario y un proyecto Oracle.

Los proyectos almacenados en GeoPackage, PostgreSQL u Oracle también pueden cargarse a través del panel del navegador QGIS, haciendo doble clic sobre ellos o arrastrándolos al lienzo del mapa.

2.2 Manejo de rutas de archivo rotas

Al abrir un proyecto, QGIS puede fallar al contactar con algunas fuentes de datos debido a la no disponibilida del servicio/base de datos, o a un archivo renombrado o movido. QGIS entonces abre el control *Manejador de Capas No Disponibles*, referenciando las capas no encontradas. Puede:

- Hacer Doble-Click en el campo *Fuente de Datos*, ajuste la ruta de cada capa y haga click en *Aplicar cambios*;
- Seleccione una fila, presione *Explorar* para indicar la localización correcta y haga click en *Aplicar cambios*;
- Pulse *Búsqueda Automática* para navegar por as carpetas e intentar corregir automáticamente todas las ruta(s) rotas o las seleccionadas. Tenga en cuenta que la avegación puede llevar algún tiempo. Después pulse *Aplicar cambios*.
- Ignore el mensaje y abra su proyecto con la(s) ruta(s) rota(s) haciendo clic en *Mantener capas no disponibles*. Luego, su capa se muestra en el panel *Capas*, pero sin ningún dato hasta que arregle la ruta usando el **|indicadorBadLayer|** ¡Capa no disponible! junto a él en el panel *Capas*, o :guilabel:`Reparar fuente de datos… `en el menú contextual de la capa.

Con la herramienta *Reparar fuente de datos…* , una vez que una ruta de capa ha sido reparada, QGIS escanea todas las demás rutas rotas e intenta reparar automáticamente aquellas que tienen la misma ruta de archivo rota.

- *Borrar Capas No Disponibles* del proyecto.

Iniciar QGIS desde la línea de comandos utilizando la opción *--skipbadlayers* puede ayudarle a omitir el diálogo *Manejar capas no disponibles* al inicio.

2.3 Generando la salida

Hay varias formas de generar resultados de su sesión QGIS. Ya hemos discutido guardar como un archivo de proyecto en *Introducción a proyectos QGIS*. Otras formas de producir archivos de salida son:

- Creación de imágenes: *Proyecto* ► *Importar/Exportar* ► *Exportar Mapa a Imagen…* envía el lienzo del mapa renderizado a un formato de imagen (PNG, JPG, TIFF…) a escala, resolución y tamaño personalizados.

  Es posible incluir información georreferencial en la imagen exportada, simplemente active *Agregar información georreferencial (incrustada o a través de un archivo mundial)*. Ver *Exportando la vista de mapa* para más detalles.

- Exportar a archivos PDF: *Project* ► *Import/Export* ► *Export Map to PDF…* genera la representación del lienzo del mapa en PDF a escala personalizada, resolución y con algunas configuraciones avanzadas (simplificación, georreferenciación, …). Ver *Exportando la vista de mapa* para mas detalles.

- Exportando a archivos DXF: *Project* ► *Import/Export* ► *Export Project to DXF…* abre un diálogo donde puede definir “Symbology mode”, La “Symbology scale” y capas vectoriales que quiera exportar a DXF. Mediante el “Symbology mode”, Los símbolos de la simbología QGIS original se pueden exportar con alta fidelidad (consulte la sección *Creando nuevos archivos DXF*).

- Diseñando mapas: *Project* ► *New Print Layout…* abre un cuadro de diálogo donde puede diseñar e imprimir el lienzo del mapa actual (consulte la sección *Diseñando los mapas*).

CAPÍTULO 3

IGU QGIS

La interfaz gráfica de usuario (IGU) se muestra en la figura de abajo (los números de 1 a 5 en círculos amarillos indican elementos importantes de la IGU de QGIS, y se discuten abajo).

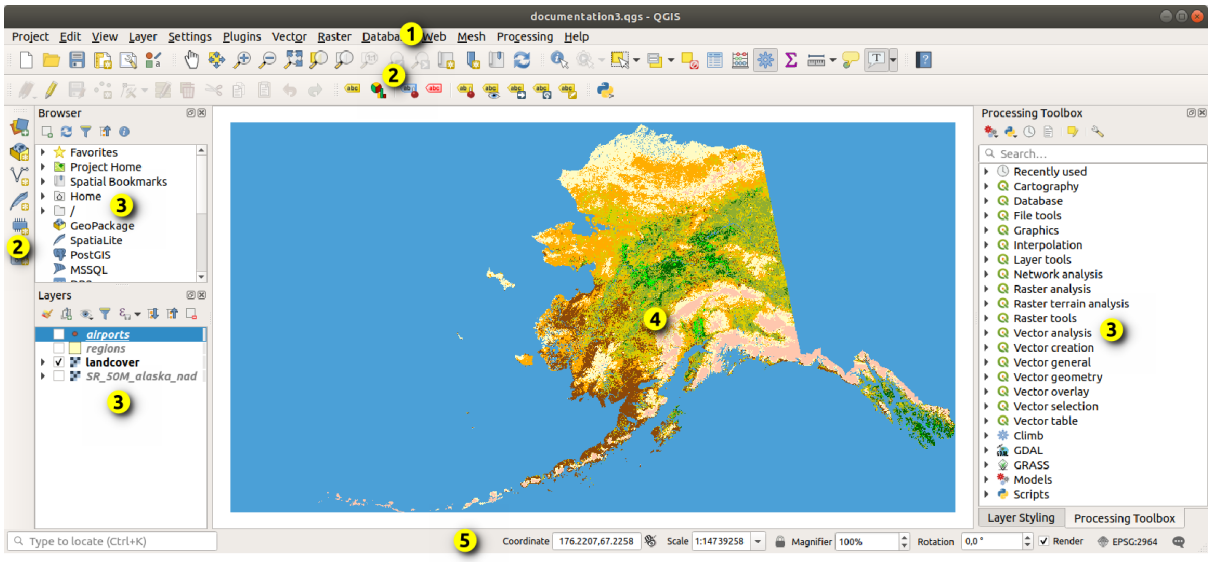


Figura 3.1: Interfaz Gráfica de Usuario de QGIS con datos de muestra de Alaska

Nota

Las decoraciones de las ventanas (barra de título, etc.) pueden ser distintas dependiendo de su sistema operativo y su gestor de ventanas.

La IGU principal de QGIS (Figura 3.1) consta de cinco componentes/tipos de componentes:

1. *Barra de menú*
2. *Barras de herramientas*
3. *Paneles*

4. *Vista de mapa*
5. *Barra de estado*

Desplazar hacia abajo para explicaciones detalladas de estos.

3.1 Barra de Menú

La barra de menú proporciona acceso a las funciones de QGIS usando menús jerárquicos estándar. Los menús, sus opciones, iconos asociados y teclas de acceso rápido se describen abajo. Las teclas de acceso rápido se pueden configurar (*Settings* ► *Keyboard Shortcuts*).

La mayoría de las opciones de menú tienen una herramienta correspondiente y viceversa. Sin embargo, los menús no están organizados exactamente como las barras de herramientas. La ubicación de las opciones de menú en las barras de herramientas, se indican abajo en una tabla. Los complementos pueden agregar nuevas opciones a los menús. Para más información acerca de herramientas y barras de herramientas, ver *Barras de herramientas*.

Nota

QGIS es una aplicación multi-plataforma. Las herramientas están disponibles de forma general en todas las plataformas, pero podrían estar ubicadas en diferentes menús, dependiendo de los sistemas operativos. Las listas de abajo muestran las ubicaciones más comunes, incluyendo variaciones conocidas.

3.1.1 Proyecto

La opción de menú *Projecto* proporciona acceso y puntos de salida para *archivos de proyecto*. Provee herramientas para:

- Crea un archivo de proyecto *New* desde cero o use algún archivo de proyecto como una plantilla (ver *Project files options* para configuración de plantillas)
- *Abrir…* un proyecto a partir de un archivo, un GeoPackage, una base de datos PostgreSQL u Oracle
- *Close* un proyecto o revertirlo a su último estado guardado
- *Guardar* un proyecto en formato `.qgs` o `.qgz`, ya sea como archivo o dentro de una base de datos GeoPackage, PostgreSQL u Oracle.
- Exporte el mapa de lieno a diferentes formatos o use *print layout* para salidas mas complejas
- Establezca las propiedades del proyecto y las opciones de ajuste para la edición de geometría.

Tabla 3.1: Las opciones del menú Proyecto

Menú Opción	Atajos	Barra de herramie	Referencia
New	`Ctrl+N`	*Project*	*Introducción a proyectos QGIS*
New from template ►			*Introducción a proyectos QGIS*
Open…	`Ctrl+O`	*Project*	*Introducción a proyectos QGIS*
Open from ►			*Introducción a proyectos QGIS*
► *GeoPackage…*			*Introducción a proyectos QGIS*
► *PostgreSQL…*			*Introducción a proyectos QGIS*
► *Oracle…*			*Introducción a proyectos QGIS*
Open Recent ►	`Alt+J + R`		*Introducción a proyectos QGIS*
► *Borrar lista*			*Introducción a proyectos QGIS*

continúe en la próxima página

Tabla 3.1 – proviene de la página anterior

Menú Opción	Atajos	Barra de herramie	Referencia
Close			*Introducción a proyectos QGIS*
Save	`Ctrl+S`	*Project*	*Introducción a proyectos QGIS*
Save As…	`Ctrl+Shift+S`	*Project*	*Introducción a proyectos QGIS*
Save to ►			*Introducción a proyectos QGIS*
► *Plantillas…*			*Introducción a proyectos QGIS*
► *GeoPackage…*			*Introducción a proyectos QGIS*
► *PostgreSQL…*			*Introducción a proyectos QGIS*
Revert…			
Propiedades…	`Ctrl+Shift+P`		*Propiedades del proyecto*
Snapping Options…			*Opciones de ajuste y digitalización*
Import/Export ►			
► *Export Map to Image…*			*Exportando la vista de mapa*
► *Export Map to PDF…*			*Exportando la vista de mapa*
► *Exportar Proyecto a DXF…*			*Creando nuevos archivos DXF*
► *Importar Capas desde DWG/DXF…*			*Importando un archivo DXF o DWG*
New Print Layout…	`Ctrl+P`	*Project*	*Diseñando los mapas*
New Report…			*Creando un informe*
Layout Manager…		*Project*	*Diseñando los mapas*
Layouts ►			*Diseñando los mapas*
Modelos ►			*El diseñador de modelos*
Exit QGIS	`Ctrl+Q`		

Under macOS, el comando *Exit QGIS* corresponde a *QGIS* ► *Quit QGIS* (`Cmd+Q`).

3.1.2 Editar

El menú *Editar* proporciona la mayoría de las herramientas nativas necesarias para editar atributos de capa o geometría. Para activar las opciones del menú *Editar* es necesario cambiar al modo de edición haciendo clic en Conmutar Edición (véase *Editando* para más detalles).

Tabla 3.2: Las opciones del menú Edición

Menú Opción	Atajos	Barra de herramietas	Referencia
Undo	`Ctrl+Z`	*Digitizing*	*Deshacer y rehacer*
Redo	`Ctrl+Shift+Z`	*Digitizing*	*Deshacer y rehacer*
Cut Features	`Ctrl+X`	*Digitizing*	*Cortar, copiar, y pegar objetos espaciales*
Copy Features	`Ctrl+C`	*Digitizing*	*Cortar, copiar, y pegar objetos espaciales*

continúe en la próxima página

Tabla 3.2 – proviene de la página anterior

Menú Opción	Atajos	Barra de herramietas	Referencia
Paste Features	`Ctrl+V`	*Digitizing*	*Cortar, copiar, y pegar objetos espaciales*
Pegar características como ►			*Trabajar con la tabla de atributos*
► *Nueva Capa Vectorial…*			*Trabajar con la tabla de atributos*
► *Capa Borrador Temporal…*	`Ctrl+Alt+V`		*Trabajar con la tabla de atributos*
Borrar Seleccionado		*Digitizing*	*Borrar objetos espaciales seleccionados*
Seleccionar ►			*Seleccionando objetos espaciales*
► *Select Feature(s)*		*Selección*	*Seleccionando objetos espaciales*
► *Select Features by Polygon*		*Selección*	*Seleccionando objetos espaciales*
► *Select Features by Freehand*		*Selección*	*Seleccionando objetos espaciales*
► *Select Features by Radius*		*Selección*	*Seleccionando objetos espaciales*
► *Select Features by Value…*	`F3`	*Selección*	*Seleccionando objetos espaciales*
► *Select Features by Expression…*	`Ctrl+F3`	*Selección*	*Seleccionando objetos espaciales*
► *Deselect Features from All Layers*	`Ctrl+Alt+A`	*Selección*	*Seleccionando objetos espaciales*
► *Deseleccionar Objetos de la Capa Activa Actual*	`Ctrl+Shift+A`	*Selección*	*Seleccionando objetos espaciales*
► *Reseleccionar Objetos*			*Seleccionando objetos espaciales*
► *Seleccionar todos los objetos*	`Ctrl+A`	*Selección*	*Seleccionando objetos espaciales*
► *Invertir Selección de Objetos*		*Selección*	*Seleccionando objetos espaciales*
Add Record	`Ctrl+.`	*Digitizing*	
Añadir Objeto Puntual	`Ctrl+.`	*Digitizing*	*Añadir objetos espaciales*
Agregar entidad de línea	`Ctrl+.`	*Digitizing*	*Añadir objetos espaciales*
Añadir Objeto Poligonal	`Ctrl+.`	*Digitizing*	*Añadir objetos espaciales*
Añadir Anotación ►			*Herramientas de anotaciones*
► *Anotación de Formulario*		*Anotaciones*	*Herramientas de anotaciones*
► *Anotación HTML*		*Anotaciones*	*Herramientas de anotaciones*

continúe en la próxima página

Tabla 3.2 – proviene de la página anterior

Menú Opción	Atajos	Barra de herramietas	Referencia
Editar Atributos ►			
► *Modificar atributos de entidades seleccionadas*		*Digitizing*	*Editar valores de atributo*
► *Fusionar atributos de entidades seleccionadas*		*Digitalización Avanzada*	*Combinar atributos de objetos espaciales*
Editar Geometría ►			
► *Mover Característica(s)*		*Digitalización Avanzada*	*Mover Entidad(es)*
► *Copiar y mover característica(s)*		*Digitalización Avanzada*	*Mover Entidad(es)*
► *Rotar Característica(s)*		*Digitalización Avanzada*	*Rotar objeto(s)*
► *Escalar Característica(s)*		*Digitalización Avanzada*	*Escala objeto espacial*
► *Simplificar Característica*		*Digitalización Avanzada*	*Simplificar objeto espacial*
► *Añadir Anillo*		*Digitalización Avanzada*	*Añadir anillo*
► *Añadir Parte*		*Digitalización Avanzada*	*Añadir parte*
► *Rellenar Anillo*		*Digitalización Avanzada*	*Rellenar anillo*
► *Borrar Anillo*		*Digitalización Avanzada*	*Borrar anillo*
► *Borrar Parte*		*Digitalización Avanzada*	*Borrar parte*
► *Volver a formar Características*		*Digitalización Avanzada*	*Remodelar objetos espaciales*
► *Curva de desplazamiento*		*Digitalización Avanzada*	*Desplazar curva*
► *Dividir Características*		*Digitalización Avanzada*	*Dividir objetos espaciales*
► *Dividir Partes*		*Digitalización Avanzada*	*Dividir partes*
► *Fusionar funciones seleccionadas*		*Digitalización Avanzada*	*Combinar objetos espaciales seleccionados*
► *Herramienta Vértice (Todas las capas)*		*Digitizing*	*Herramienta Vértices*
► *Herramienta Vértice (Capa actual)*		*Digitizing*	*Herramienta Vértices*
► *Línea inversa*		*Digitalización Avanzada*	*Línea inversa*

continúe en la próxima página

Tabla 3.2 – proviene de la página anterior

Menú Opción	Atajos	Barra de herramietas	Referencia
► *Función recortar/extender*		*Digitalización Avanzada*	*Cortar/Extender Objeto espacial*
Rotar Símbolos Puntuales		*Digitalización Avanzada*	*Rotar símbolos de puntos*
Símbolos de punto de desplazamiento		*Digitalización Avanzada*	*Símbolos de punto de desplazamiento*

Las herramientas que dependen del tipo de geometría de capa seleccionada, es decir, punto, polilínea o polígono, se activan en consecuencia:

Tabla 3.3: Los iconos basados en la geometría «Mover obejto espacial»

Menú Opción	Punto	Polilinea	Polígono
Mover Entidad(es)			
Copiar y Mover Objeto(s)			

3.1.3 Ver

El mapa se representa en vistas de mapa. Puede interactuar con estas vistas utilizando las herramientas *Vista*. Por ejemplo, puede:

- Crear nuevas vistas de mapa 2D o 3D próximo al lienzo de mapa principal
- *Zoom or pan* a algún lugar
- Consultar los atributos o la geometría de los elementos visualizados
- Mejore la vista del mapa con modos de vista previa, anotaciones o decoraciones
- Acceda a cualquier panel o barra de herramientas

El menú también le permite reorganizar la interfaz QGIS usando acciones como:

- *Toggle Full Screen Mode*: cubre toda la pantalla mientras oculta la barra de título
- *Conmutar la visibilidad de los paneles*: muestra u oculta los *paneles habilitados* - útil al digitalizar objetos espaciales (para una visibilidad máxima del lienzo) así como para presentaciones (proyectadas/grabadas) utilizando el lienzo principal de QGIS
- *Toggle Map Only*: hides panels, toolbars, menus and status bar and only shows the map canvas. Combined with the full screen option, it makes your screen display only the map.

Tabla 3.4: Las opciones del menú Vista

Menú Opción	Atajos	Barra de herramietas	Referencia
Nueva vista de mapa	`Ctrl+M`	*Navegación de Mapa*	*Vista de mapa 2D*
Vista de mapas en 3D ►			*Vista de Mapa 3D*
► *Nueva vista de mapa en 3D*	`Ctrl+Alt+M`	*Navegación de Mapa*	*Vista de Mapa 3D*
► *Gestionar Vista de Mapas 3D*			*Vista de Mapa 3D*

continúe en la próxima página

Tabla 3.4 – proviene de la página anterior

Menú Opción	Atajos	Barra de herramienta	Referencia
Mover Mapa		*Navegación de Mapa*	*Explorando la vista de mapa*
Desplazar mapa a selección		*Navegación de Mapa*	*Explorando la vista de mapa*
Acercarse	`Ctrl+Alt++`	*Navegación de Mapa*	*Explorando la vista de mapa*
Alejarse	`Ctrl+Alt+-`	*Navegación de Mapa*	*Explorando la vista de mapa*
Filtrado de datos ►			*Vista de mapa 2D*
► *Controlador Temporal*		*Navegación de Mapa*	*El panel del controlador temporal*
► *Controlador de elevación*			*El panel del controlador de elevación*
Identificar Características	`Ctrl+Shift+I`	*Atributos*	*Identificando entidades*
Medir ►		*Atributos*	*Mediciones*
► *Medir línea*	`Ctrl+Shift+M`	*Atributos*	*Mediciones*
► *Medir Área*	`Ctrl+Shift+J`	*Atributos*	*Mediciones*
► *Medir Orientación*		*Atributos*	*Mediciones*
► *Medir Ángulo*		*Atributos*	*Mediciones*
Resumen estadístico		*Atributos*	*Panel de resumen estadístico*
Perfil de elevación			*Vista del perfil de elevación*
Zoom completo	`Ctrl+Shift+F`	*Navegación de Mapa*	*Explorando la vista de mapa*
Zoom a la Selección	`Ctrl+J`	*Navegación de Mapa*	*Explorando la vista de mapa*
Zoom a la capa(s)		*Navegación de Mapa*	*Explorando la vista de mapa*
Zoom a resolución nativa (100%)		*Navegación de Mapa*	*Explorando la vista de mapa*
Ampliar último		*Navegación de Mapa*	*Explorando la vista de mapa*
Ampliar siguiente		*Navegación de Mapa*	*Explorando la vista de mapa*
Decoraciones ►	`Alt+V+D`		*Decorar el mapa*
► *Cuadrícula…*			*Cuadrícula*
► *Barra de Escala…*			*Barra de escala*
► *Imagen…*			*Imagen Ornamental*
► *Flecha Norte…*			*Flecha del Norte*
► *Etiqueta de Título…*			*Etiqueta de título*

continúe en la próxima página

Tabla 3.4 – proviene de la página anterior

Menú Opción	Atajos	Barra de herramienta	Referencia
► *Etiqueta de Copyright…*			*Etiqueta de Derechos de Copia*
► *Extensión de Diseño…*			*Extensión del diseño*
Modo de vista previa ►			
► *Normal*			
► *Simular monocromo*			
► *Simular daltonismo por acromatopsia (escala de grises)*			
► *Simular el daltonismo de Protanopia (sin rojo)*			
► *Simular el daltonismo por deuteranopía (sin verde)*			
► *Simular el daltonismo de Tritanopia (sin azul)*			
Mostrar sugerencias de mapas		*Atributos*	*Propiedades a mostrar*
Nuevo marcador espacial…	`Ctrl+B`	*Navegación de Mapa*	*Marcar extensiones en el mapa*
Mostrar marcadores espaciales	`Ctrl+Shift+B`	*Navegación de Mapa*	*Marcar extensiones en el mapa*
Mostrar administrador de marcadores espaciales			*Marcar extensiones en el mapa*
Actualizar	`F5`	*Navegación de Mapa*	
Visibilidad de capa ►			*Panel de capas*
► *Mostrar Todas las Capas*	`Ctrl+Shift+U`		*Panel de capas*
► *Ocultar Todas las Capas*	`Ctrl+Shift+H`		*Panel de capas*
► *Mostrar Capas Seleccionadas*			*Panel de capas*
► *Ocultar Capas Seleccionadas*			*Panel de capas*
► *Alternar capas seleccionadas*			*Panel de capas*
► *Toggle Selected Layers Independently*			*Panel de capas*
► *Ocultar capas no seleccionadas*			*Panel de capas*
Paneles ►			*Paneles y Barras de Herramientas*
Toolbars ►			*Paneles y Barras de Herramientas*
Conmutar Modo Pantalla Completa	`F11`		
Conmutar Panel Visibilidad	`Ctrl+Tab`		
Conmutar Solo Mapa	`Ctrl+Shift+Tab`		

Bajo Linux KDE, *Paneles* ►, *Barras de Herramientas* ► y *Conmutar Modo Pantalla Completa* están en el menú *Configuración*.

3.1.4 Capa

El menú *Capa* proporciona un amplio conjunto de herramientas para *crear* nuevas fuentes de datos, *añadir* estas a un proyecto o *guardar modificaciones* en estas. Usando las mismas fuentes de datos, puede además:

- *Duplicar* una capa para generar una copia donde se puede modificar el nombre, estilo (simbología, etiquetas, …), uniones, … La copia utiliza la misma fuente de datos que el original.
- *Copiar* y *Pegar* capas o grupos de un proyecto a otro como una nueva instancia cuyas propiedades se pueden modificar de forma independiente. En cuanto a * Duplicar *, las capas todavía se basan en la misma fuente de datos.
- o *Embed Layers and Groups…* de otro proyecto, como copias de solo lectura que no puede modificar (consulte *Incorporación de capas de proyectos externos*)

El menú *Layer* también contiene herramientas para configurar, copiar o pegar propiedades de capa (estilo, escala, CRS …).

Tabla 3.5: Las opciones del menú Capa

Menú Opción	Atajos	Barra de herramietas	Referencia
Data Source Manager	`Ctrl+L`	*Administrador Fuentes Datos*	*Opening Data*
Create Layer ►			*Creando nuevas capas vectoriales*
► *Nueva capa de GeoPackage…*	`Ctrl+Shift+N`	*Administrador Fuentes Datos*	*Creando una nueva capa de GeoPackage*
► *Nueva capa de Shapefile…*		*Administrador Fuentes Datos*	*Crear una nueva capa de archivo shape*
► *Nueva capa SpatiaLite…*		*Administrador Fuentes Datos*	*Crear una nueva capa SpatiaLite*
► *Nueva capa temporal vacía…*		*Administrador Fuentes Datos*	*Crear una nueva capa borrador temporal*
► *Nueva capa de malla…*		*Administrador Fuentes Datos*	*Crear una nueva capa de malla*
► *Nueva capa GPX…*		*Administrador Fuentes Datos*	*Crear una nueva capa GPX*
► *Nueva capa virtual…*		*Administrador Fuentes Datos*	*Creando capas virtuales*
Add Layer ►			*Abriendo Datos*
► *Agregar capa vectorial……*	`Ctrl+Shift+V`	*Administrar Capas*	*Cargando una capa desde archivo*
► *Agregar capa ráster…*	`Ctrl+Shift+R`	*Administrar Capas*	*Cargando una capa desde archivo*
► *Agregar capa de malla…*		*Administrar Capas*	*Cargando una capa de malla*
► *Agregar capa de texto delimitado…*	`Ctrl+Shift+T`	*Administrar Capas*	*Importando un archivo de texto delimitado*
► *Add PostgreSQL Layer…*	`Ctrl+Shift+D`	*Administrar Capas*	*Herramientas relacionadas con la base de datos*

continúe en la próxima página

Tabla 3.5 – proviene de la página anterior

Menú Opción	Atajos	Barra de herramietas	Referencia
► *Agregue la capa SpatiaLite…*	`Ctrl+Shift+L`	*Administrar Capas*	*Capas SpatiaLite*
► *Añadir capa MS SQL Server…*		*Administrar Capas*	*Herramientas relacionadas con la base de datos*
► *Agregar capa espacial de Oracle…*		*Administrar Capas*	*Herramientas relacionadas con la base de datos*
► *Agregar SAP HANA Spatial Layer…*		*Administrar Capas*	*Herramientas relacionadas con la base de datos*
► *Agregar/editar capa virtual…*		*Administrar Capas*	*Creando capas virtuales*
► *Agregar la capa WMS/WMTS…*	`Ctrl+Shift+W`	*Administrar Capas*	*Cargando capas WMS/WMTS*
► *Añadir capa XYZ…*			*Usando servicios de Teselas XYZ*
► *Agregar capa WCS…*		*Administrar Capas*	*WCT Cliente*
► *Añadir WFS / OGC API - Capa de objetos espaciales…*		*Administrar Capas*	*Cliente WFS y WFS-T*
► *Agregar ArcGIS REST Server Layer…*		*Administrar Capas*	*Uso de servidores REST de ArcGIS*
► *Añadir Capa Vectorial Teselada…*			*Usando de Servicios de Teselas Vectoriales*
► *Añadir Capa de Nube de Puntos…*			*Trabajar con Puntos Nubes*
► *Añadir capa GPX…*			*Introducción a los datos GNSS/GPS*
Incrustar capas y grupos…			*Incorporación de capas de proyectos externos*
Añadir desde archivo de definición de capa…			*Archivo de definición de capa*
Georeferenciador…			*Georreferenciador*
Copiar Estilo			*Guardar y compartir Propiedades*
Pegar Estilo			*Guardar y compartir Propiedades*
Copiar Capa			
Pegar Capa/Grupo			
Abrir tabla de atributos	`F6`	*Atributos*	*Trabajar con la tabla de atributos*
Filtrar Tabla de Atributos ►			*Trabajar con la tabla de atributos*
► *Abrir tabla de atributos (Objetos espaciales seleccionados)*	`Shift+F6`	*Atributos*	*Trabajar con la tabla de atributos*

continúe en la próxima página

Tabla 3.5 – proviene de la página anterior

Menú Opción	Atajos	Barra de herramietas	Referencia
► *Abrir tabla de atributos (Objetos espaciales visibles)*	`Ctrl+F6`	*Atributos*	*Trabajar con la tabla de atributos*
► *Abri tabla de atributos (Objetos espaciales editados y nuevos)*		*Atributos*	*Trabajar con la tabla de atributos*
Conmutar edición		*Digitizing*	*Digitalizando una capa existente*
Guardar Capas Editadas		*Digitizing*	*Guardar capas editadas*
Ediciones actuales ►		*Digitizing*	*Guardar capas editadas*
► *Guardar para la capa(s) seleccionada*		*Digitizing*	*Guardar capas editadas*
► *Revertir para la capa seleccionada(s)*		*Digitizing*	*Guardar capas editadas*
► *Cancelar para la capa seleccionada(s)*		*Digitizing*	*Guardar capas editadas*
► *Guardar todas las capas*		*Digitizing*	*Guardar capas editadas*
► *Retroceder para todas las capas*		*Digitizing*	*Guardar capas editadas*
► *Cancelar todas Capas*		*Digitizing*	*Guardar capas editadas*
Guardar Como…			*Creando nuevas capas desde una capa existente*
Guardar como archivo de definición de capa…			*Archivo de definición de capa*
Remove Layer/Group	`Ctrl+D`		
Duplicar Capa(s)			
Establecer visibilidad de escala de capa(s)			*Selector de escala de visibilidad*
Establecer SRC de capa(s)	`Ctrl+Shift+C`		*Sistemas de Coordenadas de Capas*
Establecer proyecto SRC desde capa			*Sistema de Coordenadas de Referencia del Proyecto*
Propiedades de Capa			• *vector* • *raster* • *malla* • *nube de puntos* • *teselas vectoriales*
Filtrar…	`Ctrl+F`		*Constructor de Consulta*
Etiquetado			*Propiedades de etiquetas*
Mostrar en Resumen			*Panel de información general*
Show All in Overview			*Panel de información general*
Hide All from Overview			*Panel de información general*

3.1.5 Configuración

Tabla 3.6: Las opciones del menú Configuración

Menú Opción	Referencia
Perfiles de usuario ►	*Trabajando con perfiles de usuario*
► *predeterminado*	*Trabajando con perfiles de usuario*
► *Abrir carpeta de perfil activo*	*Trabajando con perfiles de usuario*
► *Nuevo perfil…*	*Trabajando con perfiles de usuario*
Administrador de estilo…	*El administrador de Estilo*
Proyecciones personalizadas…	*Sistema de referencia de coordenadas personalizada*
Atajos de teclado…	*Atajos de teclado*
Personalización de la interfaz…	*Personalización*
Opciones…	*Opciones*

Bajo Linux KDE, encontrará herramientas en el menú *Settings* así como *Panels* ►, *Toolbars* ► y *Toggle Full Screen Mode*.

3.1.6 Complementos

Tabla 3.7: Las opciones del menú Complementos

Menú Opción	Atajos	Barra de herramietas	Referencia
Manage and Install Plugins…			*El diálogo de complementos*
Consola Python	`Ctrl+Alt+P`	*Plugins*	*Consola Python de QGIS*

Cuando inicie QGIS por primera vez no se cargan todos los complementos principales.

3.1.7 Vectorial

The *Vector* menu provides tools for analyzing, managing and manipulating vector datasets. By default, it is filled with a set of predefined *Processing* algorithms, grouped by sub-menus. This provides shortcuts for many common vector-based GIS tasks from core providers. If not all these sub-menus are available, enable the *Processing* plugin in *Plugins* ► *Manage and Install Plugins…*.

Nota

The list of tools and their menu can be modified/extended with any Processing algorithms (read *Configurando el marco de procesamiento*) or by some third-party *plugins*.

This is what the *Vector* menu looks like if all the core plugins are enabled.

Tabla 3.8: Las opciones del menú Vector

Menú Opción	Atajos	Barra de herramie	Referencia
Comprobar geometrías…			*Complemento Verificador de Geometría*
Comprobador de topología		*Vectorial*	*Complemento Comprobador de Topología*
Herramientas de geoprocesamiento ►	`Alt+O + G`		
► *Búfer…*			*Buffer*
► *Cortar…*			*Cortar*
► *Envolvente convexa…*			*Casco convexo*
► *Diferencia…*			*Diferencia*
► *Disolver…*			*Disolver*
► *Intersection…*			*Intersección*
► *Diferencia Simétrica…*			*Diferencia simétrica*
► *Unión…*			*Unión*
► *Eliminar Polígonos Seleccionados…*			*Eliminar polígonos seleccionados*
Geometry Tools ►	`Alt+O + E`		
► *Centroides…*			*Centroides*
► *Recoger Geometrías…*			*Coleccionar geometrías*
► *Extraer Vértices…*			*Extraer vértices*
► *Multiparte a Monoparte…*			*Multipartes a partes sencillas*
► *Polígonos a Líneas…*			*Polígonos a líneas*
► *Simplificar…*			*Simplificar*
► *Comprobar validez…*			*Comprobar validez*
► *Triangulación de Delaunay…*			*Triangulación Delaunay*
► *Densificar por Recuento…*			*Densificar por conteo*
► *Añadir Atributos de Geometría…*			*Agregar atributos de geometría*
► *Líneas a Polígonos…*			*Lineas a polígonos*
► *Polígonos de Voronoi…*			*Polígonos Voronoi*
Herramientas de Análisis ►	`Alt+O + A`		
► *Intersección de línea…*			*Intersecciones de línea*
► *Coordenada(s) media(s)…*			*Coordenada(s) Media*
► *Estadísticas básicas para campos…*			*Estadísticas básicas para campos*
► *Cuenta Puntos en Polígono…*			*Recuento de puntos en polígono*
► *Matriz de distancia…*			*Matriz distancia*
► *Listar Valores Únicos…*			*Lista de valores únicos*
► *Análisis del vecino más próximo…*			*Análisis de vecino mas próximo*

continúe en la próxima página

Tabla 3.8 – proviene de la página anterior

Menú Opción	Atajos	Barra de herramie	Referencia
► *Suma de Longitud de Líneas…*			*Suma de la longitud de las líneas*
Herramientas de Administración de Datos ►	`Alt+O + D`		
► *Fusionar Capas Vectoriales…*			*Mezclar capas vectoriales*
► *Reproyectar Capa…*			*Capa reproyectada*
► *Crear Índice Espacial…*			*Crear índice espacial*
► *Unir Atributos por Localización…*			*Unir atributos por localización*
► *Dividir Capa Vectorial…*			*Dividir capa vectorial*
Herramientas de Investigación ►	`Alt+O + R`		
► *Crear cuadrícula…*			*Crear cuadrícula*
► *Extraer la extensión de la capa…*			*Extraer extensión de la capa*
► *Puntos aleatorios en extensión…*			*Puntos aleatorios en extensión*
► *Puntos aleatorios en polígonos…*			*Puntos aleatorios en polígonos*
► *Puntos aleatorios en líneas…*			*Puntos aleatorios sobre líneas*
► *Seleccionar por ubicación…*			*Seleccionar por ubicacion*
► *Seleccionar dentro de la distancia…*			*Seleccione dentro de la distancia*
► *Puntos aleatorios en los límites de la Capa…*			*Puntos aleatorios en límites de capa*
► *Puntos aleatorios dentro de polígonos…*			*Puntos aleatorios dentro de polígonos*
► *Selección aleatoria…*			*Selección aleatoria*
► *Selección aleatoria dentro de subconjuntos…*			*Selección aleatoria con subconjuntos*
► *Puntos regulares…*			*Puntos regulares*

3.1.8 Ráster

The *Raster* menu provides tools for analyzing, managing and manipulating raster datasets. By default, it is filled with a set of predefined *Processing* algorithms, grouped by sub-menus. This provides shortcuts for many common raster-based GIS tasks from core providers. If not all these sub-menus are available, enable the *Processing* plugin in *Plugins* ► *Manage and Install Plugins…*.

Nota

The list of tools and their menu can be modified/extended with any Processing algorithms (read *Configurando el marco de procesamiento*) or by some third-party *plugins*.

This is what the *Raster* menu looks like if all the core plugins are enabled.

Tabla 3.9: Los elementos del menú Raster

Menú Opción	Atajos	Barra de herram	Referencia
Raster calculator…			*Calculadora Ráster*
Alinear rásters…			*Alinear rásters*
Analysis ►			
► *Aspecto…*			*Aspecto*
► *Rellenar nodata…*			*Rellenar NoData*
► *Cuadrícula (media móvil)…*			*Cuadrícula(promedio móvil)*
► *Cuadrícula (métrica de datos)…*			*Cuadrícula (datos métricos)*
► *Cuadrícula (distancia inversa a una potencia)…*			*Cuadrícula (distancia inversa a una fuente)*
► *Cuadrícula (Vecino más cercano)…*			*Cuadrícula (IDW con búsqueda de vecino mas próximo)*
► *Casi negro…*			*Negro cercano*
► *Mapa de sombras…*			*Sombreado*
► |proximidad| *Proximidad (Distancia Raster)…*			*Proximidad (distancia ráster)*
► *Rugosidad…*			*Rugosidad*
► *Filtrado…*			*Tamiz*
► *Pendiente…*			*Pendiente*
► *Índice de Posición Topográfica (IPT)…*			*Índice de Posición Topográfica (TPI)*
► *Índice de rugosidad del terreno (TRI)…*			*Indice de Rugosidad del Terreno (TRI)*
Projections ►			
► *Asignar Proyección…*			*Asignar proyección*
► *Extraer Proyección…*			*Extraer poyección*
► *Combar (Reproyectar)…*			*Deformación (reproyectar)*
Miscellaneous ►			
► *Construir Raster Virtual…*			*Construir ráster virtual*
► *Información Raster…*			*Información ráster*
► *Fusionar…*			*Mezclar*
► *Construir vistas generales (pirámides)…*			*Construir vista general (pirámides)*
► *Índice de teselas…*			*El índice*
Extraction ►			
► *Cortar trama por extensión…*			*Cortar ráster por extensión*
► *Cortar Raster por Capa de Máscara…*			*Cortar ráster por capa de máscara*
► *Contorno…*			*Curvas de nivel*

continúe en la próxima página

Tabla 3.9 – proviene de la página anterior

Menú Opción	Atajos	Barra de herram	Referencia
Conversion ►			
► *PCT a RGB…*			*PCT a RGB*
► *Poligonizar (Raster a Vector)…*			*Poligonizar (ráster a vectorial)*
► *Rasterizar (Vector a Raster)…*			*Rasterizar (vectorial a ráster)*
► *RGB a PCT…*			*RGB a PCT*
► *Traducir (Convertir Formato)…*			*Traducir (convertir formato)*

3.1.9 Base de datos

The *Database* menu provides tools for Database interaction. If no database-related plugins are enabled, there will be no *Database* menu. This is what the *Database* menu looks like if all the core plugins are enabled.

Tabla 3.10: Elementos del menú Base de datos

Menú Opción	Atajos	Barra de herramietas	Referencia
Offline editing…	`Alt+D` + `O`		*Complemento de Edición Fuera de Línea*
► *Convert to Offline Project…*		*Database*	*Complemento de Edición Fuera de Línea*
► *Synchronize*		*Database*	*Complemento de Edición Fuera de Línea*
Query History…		*Database*	*Creating SQL Query Layers*
DB Manager…		*Database*	*Complemento de Administración de BBDD*

Cuando inicie QGIS por primera vez no se cargan todos los complementos principales.

3.1.10 Web

Esto es como el menú *Web* parece si todos los complementos nucleares están activos. Si no hay complementos web activos, no habrá menú *Web*.

Tabla 3.11: Los elementos del menú Web

Menú Opción	Atajos	Barra de herramietas	Referencia
MetaSearch ►	`Alt+W` + `M`		*Cliente de Catálogo de metasearch*
► *Metasearch*		*Web*	*Cliente de Catálogo de metasearch*
► *Ayuda*			*Cliente de Catálogo de metasearch*

Cuando inicie QGIS por primera vez no se cargan todos los complementos principales.

3.1.11 Malla

El menú *Malla* proporciona las herramientas necesarias para manipular *capas malla*. Los complementos de terceros pueden añadir elementos a este menú.

Tabla 3.12: Los elementos del menú Malla

Menú Opción	Atajos	Barra de herramietas	Referencia
Calculadora de Malla…			*Calculadora de malla*
Volver a indexar Lados y Vértices			*Volviendo a indexar mallas*

3.1.12 Procesado

Tabla 3.13: Las opciones del menú Procesos

Menú Opción	Atajos	Barra de herramie	Referencia
Caja de Herramientas	`Ctrl+Alt+T`		*La caja de Herramientas*
Diseñador de modelos…	`Ctrl+Alt+G`		*El diseñador de modelos*
Historial…	`Ctrl+Alt+H`		*El administrador del historial*
Visor de Resultados	`Ctrl+Alt+R`		*Configurar aplicaciones externas*
Editar Objetos in-situ			*El modificador de capa de procesamiento in situ*

Cuando inicie QGIS por primera vez no se cargan todos los complementos principales.

3.1.13 Ayuda

Tabla 3.14: Los elementos del menú Ayuda

Menú Opción	Atajos	Barra de herramietas	Referencia
Contenido de la Ayuda	`F1`	*Ayuda*	
Documentación API			
Complementos ►			
Reporta un Problema			
Necesita soporte comercial?			
QGIS Home Page	`Ctrl+H`		
Comprobar Versión QGIS			
Sobre			
QGIS Sustaining Members			

3.1.14 QGIS

Este menú está solo disponible bajo macOS y contiene algunos comandos relacionados OS.

Tabla 3.15: Los elementos del menú QGIS

Menú Opción	Atajos	Referencia
Sobre QGIS		
Preferencias	`Cmd+,`	*Opciones*
Ocultar QGIS	`Cmd+H`	
Mostrar todo		
Ocultar Otros	`Shift+Cmd+H`	
Quitar QGIS	`Cmd+Q`	

:guilabel:Preferencias` corresponde a *Configuración* ► *Opciones*, :guilabel:Acerca de QGIS` corresponde a *Ayuda* ► *Acerca de* y :guilabel:Salir de QGIS` corresponde a *Proyecto* ► *Salir de QGIS* para otras plataformas.

3.2 Paneles y Barras de Herramientas

Desde el menú *Cer* (o *Configuración*), puede cambiar widgets de QGIS (*Paneles* ►) y activar y desactivar barras de herramientas (*Barras de Herramientas* ►). Para (des)activar alguna de ellas, click-derecho en la barra de menú o barra de herramientas y escoja el elemento que quiera. Paneles y barras de herramientas se puede mover y ubicar donde quiera dentro de la interfaz de QGIS. La lista se puede ampliar con la activación de los *Núcleo o complementos externos*.

3.2.1 Barras de herramientas

Las barras de herramientas proporcionan acceso a la mayoría de las funciones en los menús, además de herramientas adicionales para interactuar con el mapa. Cada elemento de la barra de herramientas tiene ayuda emergente disponible. Pase el mouse sobre el elemento y se mostrará una breve descripción del propósito de la herramienta.

Las barras de herramientas disponibles son:

Tabla 3.16: Barras de herramientas QGIS

Nombre de barra de herramienta	Referencia principal para herramientas
Digitalización Avanzada	*Digitalización avanzada*
Anotaciones	*Herramientas de anotaciones*
Atributos	*Trabajar con la tabla de atributos*, *Herramientas generales*
Administrador Fuentes Datos	*Administrar el origen de datos*
Database	*Complemento de Administración de BBDD*
Digitizing	*Digitalizando una capa existente*
GPS	*Seguimiento de GPS en Vivo*
GRASS	*Integracion GRASS SIG*
Ayuda	
Etiqueta	*La Barra de Herramientas Etiqueta*
Administrar Capas	*Abriendo Datos*
Navegación de Mapa	*Explorando la vista de mapa*
Digitalización de malla	*Editando la capa de malla*
Plugins	*Complementos*
Project	*Trabajando con Archivos de Proyecto*, *Diseñando los mapas*, *La librería Style*
Algoritmos de procesamiento	*Configurando el marco de procesamiento*
Raster	*Complementos*
Selección	*Seleccionando objetos espaciales*
Digitalización de formas	*Digitalización de formas*
Snapping	*Configuración de la tolerancia de ajuste y el radio de búsqueda*
Vectorial	*Complementos*
Web	*Complementos*, *Cliente de Catálogo de metasearch*

Nota

Los complementos de terceros pueden ampliar la barra de herramientas predeterminada con sus propias herramientas o proporcionar su propia barra de herramientas.

Truco

Restauración de barras de herramientas

Si accidentalmente ha ocultado una barra de herramientas, puede restaurarla usando *View* ► *Toolbars* ► (o *Settings* ► *Toolbars* ►). Si, por alguna razón, una barra de herramientas (o algún otro widget) desaparece totalmente de la interfaz, encontrará consejos para recuperarlo en *restoring initial GUI*.

3.2.2 Paneles

QGIS proporciona muchos paneles. Los paneles son widgets especiales con los que puede interactuar (seleccionar opciones, marcar casillas, completar valores …) para realizar tareas más complejas.

A continuación se muestra una lista de los paneles predeterminados proporcionados por QGIS:

Tabla 3.17: Paneles QGIS

Nombre de panel	Atajos	Referencia
Digitalización Avanzada	`Ctrl+4`	*El panel de Digitalización Avanzada*
Navegador	`Ctrl+2`	*El panel Navegador*
Navegador (2)		*El panel Navegador*
Herramientas de depuración/desarrollo	`F12`	*Panel de herramientas de desarrollo/depuración*
Perfil de elevación		*Vista del perfil de elevación*
Validación de geometría		*Propiedades de Digitalizado*
Información GPS	`Ctrl+0`	*Seguimiento de GPS en Vivo*
Herramientas GRASS		*Integracion GRASS SIG*
Orden de capa	`Ctrl+9`	*Panel de orden de capa*
Estilo de capa	`Ctrl+3`	*Panel de Estilizado de Capa*
Capas	`Ctrl+1`	*Panel de capas*
Mensajes del registro		*Panel de mensajes de registro*
Vista general	`Ctrl+8`	*Panel de información general*
Caja de herramientas Procesos		*La caja de Herramientas*
Visor de resultados		*La caja de Herramientas*
Opciones de ajuste y digitalización		*Configuración de la tolerancia de ajuste y el radio de búsqueda*
:guilabel:`Administrador de marcadores espaciales	`Ctrl+7`	*Marcar extensiones en el mapa*
Estadísticas	`Ctrl+6`	*Panel de resumen estadístico*
Controlador temporal		*El panel del controlador temporal*
Escala de tesela		*Conjunto de teselas*
Deshacer/Rehacer	`Ctrl+5`	*Panel de deshacer/rehacer*
Editor de vértice		*El Panel de Editor de Vértices*

3.3 Barra de Estado

La barra de estado le proporciona información general sobre la vista del mapa y las acciones procesadas o disponibles, y le ofrece herramientas para administrar la vista del mapa.

3.3.1 Barra de localización

En el lado izquierdo de la barra de estado, la barra de localizador, un complemento de búsqueda rápida, le ayuda a encontrar y ejecutar cualquier funcionalidad u opción en QGIS:

1. Click en el complemento de texto para activar la barra de búsqueda de localizador o presionando `Ctrl+K`.
2. Escriba un texto asociado al elemento que está buscando (nombre, etiqueta, palabra clave, …). De forma predeterminada, los resultados se devuelven a los filtros de localizadores habilitados, pero puede limitar la búsqueda a un cierto alcance colocando un prefijo en su texto con prefijo *filtros de localizador*, p.ej. escribiendo `l cad` devolverá solo las capas cuyo nombre contiene `cad`.

   El filtro puede también ser seleccionada con un doble-click en el menú mostrado al acceder al complemento del localizador.
3. Click en un resultado para ejecutar la acción correspondiente, dependiendo del tipo de elemento.

Truco

Limitar la búsqueda a campo(s) particulares de la capa activa

Por defecto, una búsqueda con el filtro «características de la capa activa» (`f`) recorre la tabla de atributos de la capa entera. Puede limitar la búsqueda a un campo en particular usando el prefijo `@`. P.Ej. `f @name sal` o `@name sal` devuelve solo las entidades cuyo atributo «name» contiene "sal". El autocompletado de texto está activado al escribir y la sugerencia puede ser aplicada usando la tecla `Tab`.

Es posible un control más avanzado sobre los campos consultados desde la capa *Fields* tab. Read *Propiedades de campos* para mostrar detalles

La búsqueda se maneja mediante hilos, de modo que los resultados siempre estén disponibles lo más rápido posible, incluso si se instalan filtros de búsqueda lentos. También aparecen tan pronto como son detectados por un filtro, lo que significa que, p. Ej. un filtro de búsqueda de archivos mostrará los resultados uno por uno a medida que se escanea el árbol de archivos. Esto garantiza que la interfaz de usuario siempre responda, incluso si hay un filtro de búsqueda muy lento (por ejemplo, uno que utiliza un servicio en línea).

Nota

La herramienta de localización de Nominatim puede comportarse de manera diferente (sin búsqueda de autocompletado, retraso en la obtención de resultados, …) con respecto a la política de uso de OpenStreetMap Nominatim. <https://operations.osmfoundation.org/policies/nominatim/>`_.

Truco

Acceso rápido a los ajustes del localizador

Click en el icono dentro del complemento del localizador en la barra de estado para mostrar la lista de filtros que puede usar y una entrada *Configurar* que abre la pestaña *Localizador* del menú *AJustes* ► *Opciones…*.

3.3.2 Reporte de acciones

En el área junto a la barra de ubicación, se mostrará un resumen de las acciones que ha llevado a cabo cuando sea necesario (como seleccionar entidades en una capa, eliminar una capa, desplazar la distancia y la dirección) o una descripción larga de la herramienta que está desplazando. over (no disponible para todas las herramientas).

En el caso de operaciones prolongadas, como la recopilación de estadísticas en capas ráster, la ejecución de algoritmos de procesamiento o la representación de varias capas en la vista de mapa, se muestra una barra de progreso en la barra de estado.

3.3.3 Controlar el lienzo del mapa

La opción *Coordinate* muestra la posición actual del mouse, siguiéndola mientras se mueve por la vista del mapa. Puede configurar las unidades (y la precisión) en la pestaña *Project* ► *Properties…* ► *General*. Click en el botón pequeño a la izquierda del cuadro de texto para alternar entre la opción coordinar y la opción *Extents* que muestra las coordenadas de las esquinas actuales inferior izquierda y superior derecha de la vista de mapa en unidades de mapa.

Junto a la pantalla de coordenadas, encontrará la pantalla *Scale*. Muestra la escala de la vista del mapa. Hay un selector de escala, que le permite elegir entre *predefined and custom scales*.

En la parte derecha de la pantalla de la escala, pulse el botón para bloquear la escala y utilizar la lupa para acercar

o alejar la imagen. La lupa permite ampliar un mapa sin alterar la escala del mapa, lo que facilita ajustar con precisión las posiciones de las etiquetas y los símbolos. El nivel de ampliación se expresa en porcentaje. Si la *Lupa* tiene un nivel de 100%, el mapa actual no se amplía, es decir, se representa a escala exacta en relación con la resolución del monitor (PPP). Puede definirse un valor de ampliación por defecto en *Configuración* ► *Opciones* ► *Renderizado* ► *Comportamiento de Renderizado*, que resulta muy útil en pantallas de alta resolución para ampliar símbolos pequeños. Además, un ajuste en *Configuración* ► *Opciones* ► *Lienzo y Leyenda* ► *PPP* controla si QGIS respeta los PPP físicos de cada monitor o utiliza los PPP lógicos del sistema.

A la derecha de la herramienta de lupa puede definir una rotación actual en sentido horario para su vista de mapa en grados.

En la parte derecha de la barra de estado, la casilla *Renderizar* se puede utilizar para suspender temporalmente la renderización de la vista del mapa (ver sección *Control de la representación de mapas*).

A la derecha de la función *Renderizar*, se encuentra el botón *EPSG:code* que muestra el CRS actual del proyecto. Al hacer clic en él, se abre el cuadro de diálogo *Propiedades del proyecto* que permite reproyectar la vista del mapa o ajustar cualquier otra propiedad del proyecto.

Truco

Calcular la escala correcta de su lienzo de mapa

Cuando inicia QGIS, el CRS predeterminado es `WGS 84 (EPSG 4326) ''` y las unidades son grados. Esto significa que QGIS interpretará cualquier coordenada en su capa como se especifica en grados. Para obtener valores de escala correctos, puede cambiar manualmente esta configuración en la pestaña :guilabel:`General` bajo :menuselection:`Project --> Properties...` (p. ej. a metros), o puede usar el icono |projectionEnabled| :sup:`EPSG:code` mostrado encima. En el último caso, las unidades se configuran según lo que especifica la proyección del proyecto (p. Ej., ``+units=us-ft).

Tenga en cuenta que la elección de SRC en el arranque puede establecerse en *Configuración* ► *Opciones* ► *Manejo de SRC*.

3.3.4 Mensajería

El botón Messages al lado de él abre el *Log Messages Panel* que tiene información sobre procesos subyacentes (inicio de QGIS, carga de complementos, herramientas de procesamiento …)

CAPÍTULO 4

El panel Navegador

- *Recursos que se pueden abrir/ejecutar desde el navegador*
- *Entradas de nivel superior del panel del navegador*
  - *Favoritos*
  - *Marcadores espaciales*
  - *Proyecto Inicio*
  - *Unidades y sistema de archivos*
  - *Tipo de datos*
  - *Entradas de la base de datos*
  - *Mosaicos y servicios web*
  - *Proyecto y tipos de archivo compatibles*

El panel del navegador QGIS es una gran herramienta para navegar, buscar, inspeccionar, copiar y cargar recursos QGIS. Solo los recursos que QGIS sabe cómo manejar se muestran en el navegador.

Usando el panel del Navegador puede localizar, inspeccionar y agregar datos, como se describe en *El panel Navegador*. Además, el panel del navegador admite la función de arrastrar y soltar muchos recursos de QGIS, como archivos de proyecto, scripts de Python, scripts de procesamiento y modelos de procesamiento.

Los scripts de Python, los scripts de procesamiento y los modelos de procesamiento también se pueden abrir para editarlos en un editor externo y en el modelador gráfico.

You can drag and drop layers from the *Layers* panel to the *Browser* panel, for instance into a GeoPackage or a PostgreSQL database.

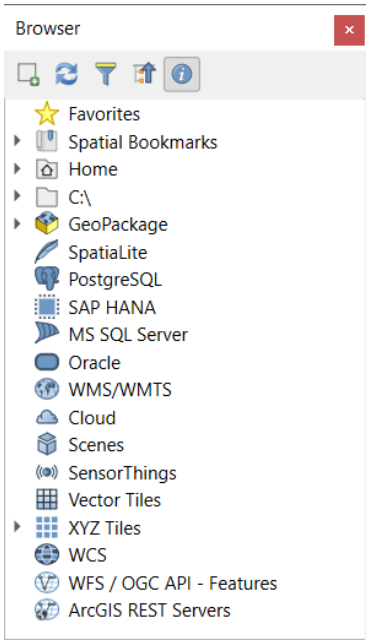


Figura 4.1: El panel Navegador

El panel del navegador (Figura 4.1) está organizado como una jerarquía expandible con algunas entradas fijas de nivel superior que organizan los recursos manejados por el navegador. Las entradas de nodo se expanden haciendo click en a la izquierda del nombre de la entrada. Una rama se contrae haciendo click en . El botón Contraer todo contrae todas las entradas de nivel superior.

En *Ajustes* ► *Personalización de Interface* es posible deshabilitar recursos. Si usted, por ejemplo, no desearía mostrar los scripts Python en el navegador, usted puede quitar la selección de la entrada *Navegador* ► *py* y si desea deshacerse de su carpeta de inicio en el navegador usted puede quitar la selección de la entrada *Navegador* ► *special:Inicio*.

Se puede utilizar un filtro (Filter Browser) para buscar en función de los nombres de las entradas (tanto las entradas de hoja como las entradas de nodo en la jerarquía). Usando el menú desplegable **|opciones|** Opciones junto al campo de texto del filtro, puede

- cambiar búsqueda *sensible a mayúsculas y minúsculas*
- establecer *Sintaxis del patrón de filtro* a uno de
  - *Normal*
  - *Comodín(es)*
  - *Expresiones Regulares*

El *widget de Propiedades*, muestra información útil sobre algunas entradas/recursos, se puede habilitar/deshabilitar usando el botón **|metadatos|** Activar/desactivar el widget de propiedades. Cuando está habilitado, se abre en la parte inferior del panel del navegador, como se muestra en Figura 4.2.

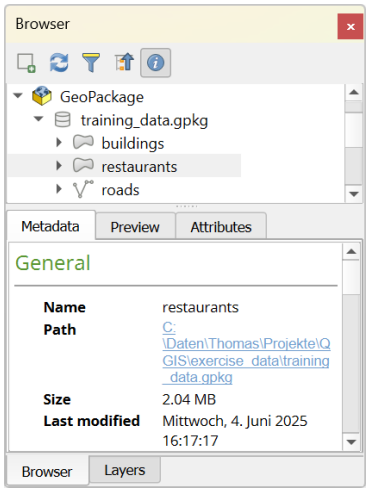


Figura 4.2: El Widget Propiedades

Se puede abrir un segundo panel navegador activando el panel *Explorador (2)* en *Ver* ► *Paneles*. Teniendo dos paneles de exploración puede ser útil al copiar capas de recursos localizados dentro de diferentes ramas de la jerarquía del explorador.

4.1 Recursos que se pueden abrir/ejecutar desde el navegador

Se pueden lograr muchas cosas en el panel del navegador

- Agregar capas vectoriales, ráster y de malla a su mapa haciendo doble click, arrastrándolo al lienzo del mapa o haciendo click en el botón Agregar capas seleccionadas (después de seleccionar capas)
- Ejecutar Scripts de python (incluidos algoritmos de Procesamiento) haciendo doble click o arrastrando sobre el mapa del lienzo
- Ejecutar modelos haciendo doble clic o arrastrándolos al lienzo del mapa
- *Extraer Símbolos…* de archivos del Proyecto QGIS usando el menú contextual
- Abrir archivos con sus aplicaciones por defecto (*Abrir <file type> Externamente…* en el menú contextual). Ejemplos: archivos HTML, hojas de calculo, imagenes, PDFs, archivos de texto, …
- Copiar entradas
- Cambiar el nombre y eliminar (múltiples) capas (Menú de contexto: *Manage* ►)
- Abra una ventana del explorador de archivos y seleccione directamente el archivo *Mostrar en Archivos*.

Las acciones específicas de recursos se enumeran para los diferentes grupos de recursos ordenados en las entradas de nivel superior que se enumeran a continuación.

4.2 Entradas de nivel superior del panel del navegador

4.2.1 Favoritos

Las ubicaciones del sistema de archivos más utilizadas se pueden etiquetar como favoritas. Las que hayas etiquetado aparecerán aquí.

Además de las opercaiones descritas en *Home*, el menú contextual le permite *Renombrar Favorito…* y *Borrar Favorito*.

4.2.2 Marcadores espaciales

This is where you will find your *spatial bookmarks*, divided into *Project Bookmarks* and *User Bookmarks*.

Tabla 4.1: available actions for spatial bookmarks

Nivel	Contextual menu
Spatial Bookmarks	*New Spatial Bookmark…*
	Show the Spatial Bookmark Manager
	Import Spatial Bookmarks…
	Export Spatial Bookmarks…
Bookmark group (*Project Bookmarks* and *User Bookmarks*)	*New Spatial Bookmark…*
	Import Spatial Bookmarks…
	Export Spatial Bookmarks…
Individual bookmark	*Zoom to Bookmark*
	Edit Spatial Bookmark…
	Delete Spatial Bookmark

4.2.3 Proyecto Inicio

Disponible una vez que se ha guardado el archivo del proyecto, la entrada *Project home* es una carpeta que contiene datos y otros contenidos (guiones, modelos, texto, …) que se pueden utilizar dentro del proyecto actual. Mostrado en el panel *Browser*, permite acceder rápidamente a los datos y otros archivos del proyecto.

Por defecto es la carpeta del archivo del proyecto, pero se puede cambiar a través de la opción *Project* ► *Properties…* ► *General* ► *Project home*, o haciendo clic con el botón derecho en *Project Home `del panel Navegador y seleccionando :guilabel:`Set project home…*. Personalizar esa carpeta es especialmente útil en contextos donde los proyectos de QGIS no se almacenan en la carpeta raíz de un "project" organizacional, junto con conjuntos de datos.

4.2.4 Unidades y sistema de archivos

Los siguientes elementos del panel *Browser* dependen del sistema operativo en uso y se refieren a las entradas de nivel superior de su sistema de archivos.

Son principalmente:

- La carpeta *Home*, que apunta a la carpeta de inicio del usuario actual
- en máquinas basadas en Unix, la carpeta raíz `/`
- las unidades conectadas, ya sean locales o de red. Según el sistema operativo, se enumeran directamente (p. ej., `C:\`, `D:\`) o a través de la entrada `/Volumes`.

Desde el menú contextual de cada una de estas carpetas o unidades, puedes:

- refrescar los contenidos
- cree un subelemento *New* ► que sea un conjunto de datos con formato *Directory*, *GeoPackage* o ESRI *Shapefile*

- ocultar el directorio (*Ocultar del Navegador*)
- *Set color*: personalizar el color del icono de la carpeta, lo que ayuda en la navegación rápida del navegador de estructuras de carpetas complejas
- habilitar *Scanning*:
  - *Monitor for changes*: permite controlar manualmente si un directorio en particular debe ser monitoreado y actualizado automáticamente. Esta configuración se aplica al directorio seleccionado y todos los subdirectorios. Esto significa que puede habilitar manualmente la supervisión de unidades de red si sabe que no hay ningún problema, o cancelar manualmente la supervisión de directorios grandes que no desea monitorear por otros motivos. De forma predeterminada, las unidades remotas o de red no se supervisan automáticamente.
  - *Escanear rápido este directorio*
- abra el directorio en su administrador de archivos (*Open Directory…*)
- abrir el directorio en una ventana de terminal (*Open in Terminal…*)
- inspeccionar las *Properties…* o las *Directory Properties…*

4.2.5 Tipo de datos

Tabla 4.2: Available actions for plain layer files

Level and Context menu	Tipos de capas Raster (*Reference*)	Vector (*Reference*)	Point clouds (*Reference*)	Mesh (*Reference*)
File				
Eliminar archivo <dataset name>	✓	✓		
Mostrar en archivos	✓	✓	✓	✓
Administrar	✓	✓	✓	✓
File Properties…	✓	✓	✓	✓
Open <file type> Externally…	✓			
Layer				
Añadir capa a proyecto	✓	✓	✓	✓
Exportar capa	✓	✓		
Layer Properties…	✓	✓	✓	✓
Open with Data Source Manager…	✓	✓		
Execute SQL…		✓		

4.2.6 Entradas de la base de datos

Según su sistema operativo y los controladores instalados, es posible que tenga acceso a diferentes tipos de bases de datos para usar en QGIS. A continuación se enumeran las diferentes entradas del menú contextual en cada nivel del árbol del conjunto de datos.

Tabla 4.3: Available actions for database entries

Level and Context menu	Tipo de base de datos

continúe en la próxima página

Tabla 4.3 – proviene de la página anterior

	GeoPackage ([1])	SpatiaLite	PostgreSQL	SAP HANA	MS SQL Server	Oracle
Top menu						
Crear una *New Connection…* a una base de datos existente	☑	☑	☑	☑	☑	☑
Crear base de datos…	☑	☑				
Crear base de datos y capa…	☑					
Guardar Conexiones… detalles a un archivo			☑	☑	☑	
Cargar Conexiones…			☑	☑	☑	
Connection / Database						
Refrescar una conexión			☑	☑	☑	☑
Editar conexión… configuración			☑	☑	☑	☑
Duplicar conexión			☑	☑	☑	☑
Quitar conexión…	☑	☑	☑	☑	☑	☑
Import Vector Layer…	☑	☑	☑	☑	☑	
Nuevo dominio de campo ► Nuevo rango de dominio	☑					
Nuevo dominio de campo ► Nuevo dominio de valores codificados.	☑					
Nuevo Dominio de Campo ► Nuevo Dominio Glob.	☑					
Delete <database_name>	☑	☑				
Base de Datos Compacta (VACÍO)	☑					
Crear un *New Schema…*			☑	☑	☑	
Crear un *New Table…*	☑	☑	☑	☑		
Ejecutar SQL… consulta	☑	☑	☑	☑		
Schema						
Refrescar un esquema			☑	☑	☑	
Operaciones de esquema ► Cambiar nombre de esquema…			☑	☑	☑	
Operaciones de esquema ► Eliminar esquema…			☑	☑	☑	
Crear un *New Table…*			☑	☑		
Ejecutar SQL… consulta			☑	☑		
Table / Layer						
Operaciones de tabla ► Cambiar nombre de tabla…			☑	☑	☑	

continúe en la próxima página

Tabla 4.3 – proviene de la página anterior

Operaciones de tabla ► Truncar tabla…			☑		☑
Ejecutar SQL… consulta	☑	☑	☑		
Exportar capa ► A archivo…	☑	☑	☑	☑	☑
Administrar ► Cambiar nombre de capa<layer_name>…	☑	☑			☑
Administrar ► Eliminar capa<layer_name>…	☑	☑	☑	☑	☑
Administrar –> Eliminar capa	☑	☑	☑	☑	☑
Administrar ► Agregar capa al proyecto	☑	☑	☑	☑	☑
Administrar ► Agregar capas seleccionadas al proyecto	☑	☑	☑	☑	☑
Manage ► Move to Another Schema			☑		☑
Abrir el cuadro de diálogo *Propiedades de la capa…*	☑	☑	☑	☑	☑
Abra el cuadro de diálogo *Propiedades del archivo…*	☑				
Abrir con el Gestor de Fuentes de Datos	☑	☑			
Fields					
Añadir nuevo campo…	☑	☑	☑	☑	
Field					
Configurar dominio de campo	☑				
Rename Field ([2])	☑		☑		☑
Configurar alias…	☑				
Configurar comentario…	☑		☑		
Eliminar campo…	☑	☑	☑	☑	

Nota

Al cargar capas vectoriales en QGIS, los campos con  *Dominios de campo* (como los definidos en un GeoPackage o en una Geodatabase de archivos ESRI) se detectan automáticamente. Estos dominios son restricciones a nivel de base de datos, lo que significa que son impuestas por la propia base de datos y se aplican a través de diferentes aplicaciones, no sólo QGIS.

[1] Las diferentes entradas también pueden estar disponibles para algunos formatos de archivo *vectorial* soportados por GDAL <https://gdal.org/en/latest/drivers/vector/index.html>`_ como ESRI File Geodatabase, FlatGeobuf, GeoParquet, NetCDF, … cuando sean compatibles.

[2] For PostgreSQL connections: only supported with ALTER TABLE privilege.

Importing vector layers into databases using Browser panel

QGIS allows you to import vector layers directly into a database from the *Browser* panel. This provides a simple way to transfer datasets into supported database formats without using DB Manager (see *Complemento de Administración de BBDD*).

You can start the import in two ways:

- Drag and drop a single vector layer from the Browser panel or from the Project into a database connection or schema item.
- Right-click on a database connection or schema item and choose *Import Vector Layer…* from the context menu.

Nota

When multiple layers are dragged into a database, they are imported immediately without further options. All imports run in the background, so the QGIS interface remains responsive during the process.

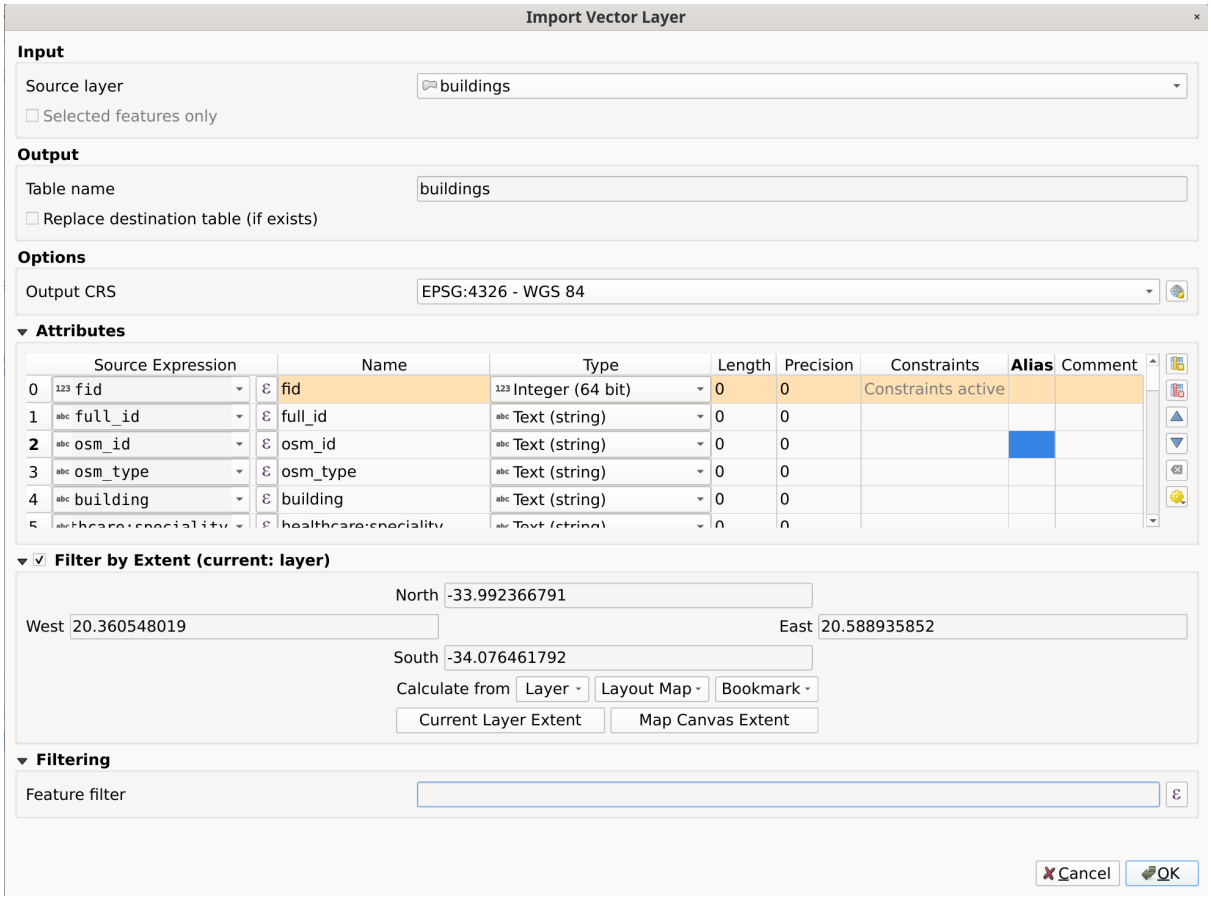


Figura 4.3: Import Vector Layer dialog

To add a layer using *Import Vector Layer…* follow these steps:

1. Add *Source layer*: the layer to import, chosen from the current QGIS project. To import only selected features of the source layer use the *Selected features only* option.
2. Set the *Table name*: the name of the destination table in the database. By default, it is taken from the source layer name but can be modified. You can also use the *Replace destination table (if exists)* to overwrite an existing table.

3. Define the *Output CRS*: the coordinate reference system of the destination table. By default, it is the CRS of the source layer, but it can be changed using the *CRS selector*.
4. Configure the *Attributes*:
   - *Source expression*: expression used to populate the field. By default, this corresponds to the original source field.
   - *Name*: name of the field in the destination table.
   - *Type*: field data type (integer, text, real, etc.).
   - *Length* and *Precision*: length and precision of the field, where applicable.
   - *Constraints*: rules to enforce on the field values (e.g. NOT NULL, UNIQUE, DEFAULT). Support depends on the destination database.
   - *Alias*: an alternate display name for the field.
   - *Comment*: a description or note attached to the field.

   You can use buttons on the right side to:
   -  Add new field to add a new field to the destination table.
   -  Delete selected field to remove a field from the destination table.
   -  Move selected field up or  Move selected field down to reorder fields.
   -  Reset all fields to clear all fields and start from scratch.
   -  Convert All Fields To Lowercase or  Convert All Fields To Uppercase, to change the case of all field names.
5. Optionally, apply a spatial filter with *Filter by Extent*. This restricts the import to features within a defined bounding box.
6. Optionally, apply an attribute filter with *Feature filter*. This allows you to use an expression to select which features are imported.

4.2.7 Mosaicos y servicios web

Tabla 4.4: Available actions for web services (1)

Level and Context menu	Tipo de servicios WMS / WMTS (*Reference*)	XYZ Tiles (*Reference*)	WCS (*Reference*)	WFS / OGC API - Features (*Reference*)	ArcGIS REST Servers (*Reference*)
Top menu					
Crear un *New Connection…*	☑	☑	☑	☑	☑
Guardar Conexiones… detalles a un archivo	☑	☑	☑	☑	☑
Cargar Conexiones…	☑	☑	☑	☑	☑
Connection					
Refrescar conexión	☑	☑	☑	☑	☑
Editar conexión… configuración	☑	☑	☑	☑	☑
Duplicar conexión	☑	☑	☑	☑	☑
Quitar conexión…	☑	☑	☑	☑	☑

continúe en la próxima página

Tabla 4.4 – proviene de la página anterior

Level and Context menu	Tipo de servicios				
	WMS / WMTS (*Reference*)	XYZ Tiles (*Reference*)	WCS (*Reference*)	WFS / OGC API - Features (*Reference*)	ArcGIS REST Servers (*Reference*)
Ver información del servicio en el navegador web					☑
Table / Layer					
Exportar capa ► Al Archivo…	☑	☑	☑	☑	☑
Añadir capa a proyecto	☑	☑	☑	☑	☑
Add Filtered Layer to Project…					☑
Abrir el cuadro de diálogo *Propiedades de capa…*	☑	☑	☑	☑	☑
Ver información del servicio en el navegador web					☑

Tabla 4.5: Available actions for web services (2)

Level and Context menu	Tipo de servicios				
	Vector Tiles (*Reference*)	Scene (*Reference*)	Cloud (*Reference*)	SensorThings (*Reference*)	STAC (*Reference*)
Top menu					
Crear un *New Connection…*		☑	☑	☑	☑
Crear un *New Generic Connection…*	☑				
Crear un *New ArcGIS Vector Tile Service Connection…*	☑				
Save Connections… details to a file	☑	☑	☑	☑	☑
Cargar Conexiones…	☑	☑	☑	☑	☑
Connection					
Editar conexión… configuración	☑	☑	☑	☑	☑
Duplicar conexión	☑	☑	☑	☑	
Delete Connection…	☑	☑	☑	☑	☑
Refresh					☑
Details…					☑
Table / Layer					
Exportar capa ► Al Archivo…				☑	
Añadir capa a proyecto	☑	☑		☑	
Abrir el cuadro de diálogo *Propiedades de capa…*	☑	☑		☑	
Refresh					☑
Download assets…					☑

continúe en la próxima página

Tabla 4.5 – proviene de la página anterior

Level and Context menu	Tipo de servicios				
	Vector Tiles (*Reference*)	Scene (*Reference*)	Cloud (*Reference*)	SensorThings (*Reference*)	STAC (*Reference*)
Details…					☑

4.2.8 Proyecto y tipos de archivo compatibles

Tabla 4.6: Available actions for various supported files :class: longtable

Tipo de archivo	Menú de contexto
Archivo de proyecto de QGIS (.qgz, .qgs)	*Abrir proyecto*
	Extraer símbolos… (abre el Administrador de estilos para exportar los símbolos a formato XML, añadirlos al estilo predeterminado o exportarlos como PNG/SVG)
	Propiedades del archivo…
Definición de capa de QGIS (.qlr)	*Exportar capa ► Al Archivo…*
	Añadir capa al proyecto
	Propiedades de la capa…
Modelo de procesamiento (.model3)	*Ejecutar modelo…*
	Editar modelo…
Plantilla de Composición de Impresión (.qpt)	*Nuevo diseño a partir de plantilla*
Script de Python (.py)	*Ejecutar script…*
	Abrir en editor externo

CAPÍTULO 5

Configuración QGIS

QGIS es altamente configurable. Mediante el menú *Ajustes*, proporciona diferentes herramientas para:

- *Administrador de estilo…*: crear y administrar *símbolos, estilos y rampas de color*.
- *Proyecciones personalizadas…*: crear tus propios :ref:`coordinate reference systems `.
- *Atajos de teclado…*: define tu propio conjunto de *keyboard shortcuts*. Además, pueden ser anulados durante cada sesión de QGIS por el *project properties* (acessible en el menú *Proyecto*).
- *Personalización de la Interfaz…*: configure la *application interface*, ocultando cuadros de diálogo o herramientas que quizás no necesite.
- *Opciones…*: establece las *options* globales a aplicar en diferentes áreas del software. Estas preferencias se guardan en los ajustes activos *User profile* y se aplica por defecto cada vez que abre un nuevo proyecto con este perfil.

5.1 Opciones

Algunas opciones básicas para QGIS se pueden seleccionar usando el diálogo *Opciones*. Selecciona la opción de menú *Ajustes*► *Opciones*. Puede modificar las opciones según sus necesidades. Algunos de los cambios pueden requerir un reinicio de QGIS antes de que sean efectivos.

Las pestañas donde puede personalizar sus opciones se describen a continuación.

> **Nota**
>
> **Los complementos pueden incrustar su configuración dentro del cuadro de diálogo Opciones**
>
> Si bien solo se presentan las configuraciones principales a continuación, tenga en cuenta que esta lista puede ampliarse *installed plugins* implementando sus propias opciones en el cuadro de diálogo Opciones estándar. Esto evita que cada complemento tenga su propio diálogo de configuración con elementos de menú adicionales solo para ellos …

5.1.1 Configuración general

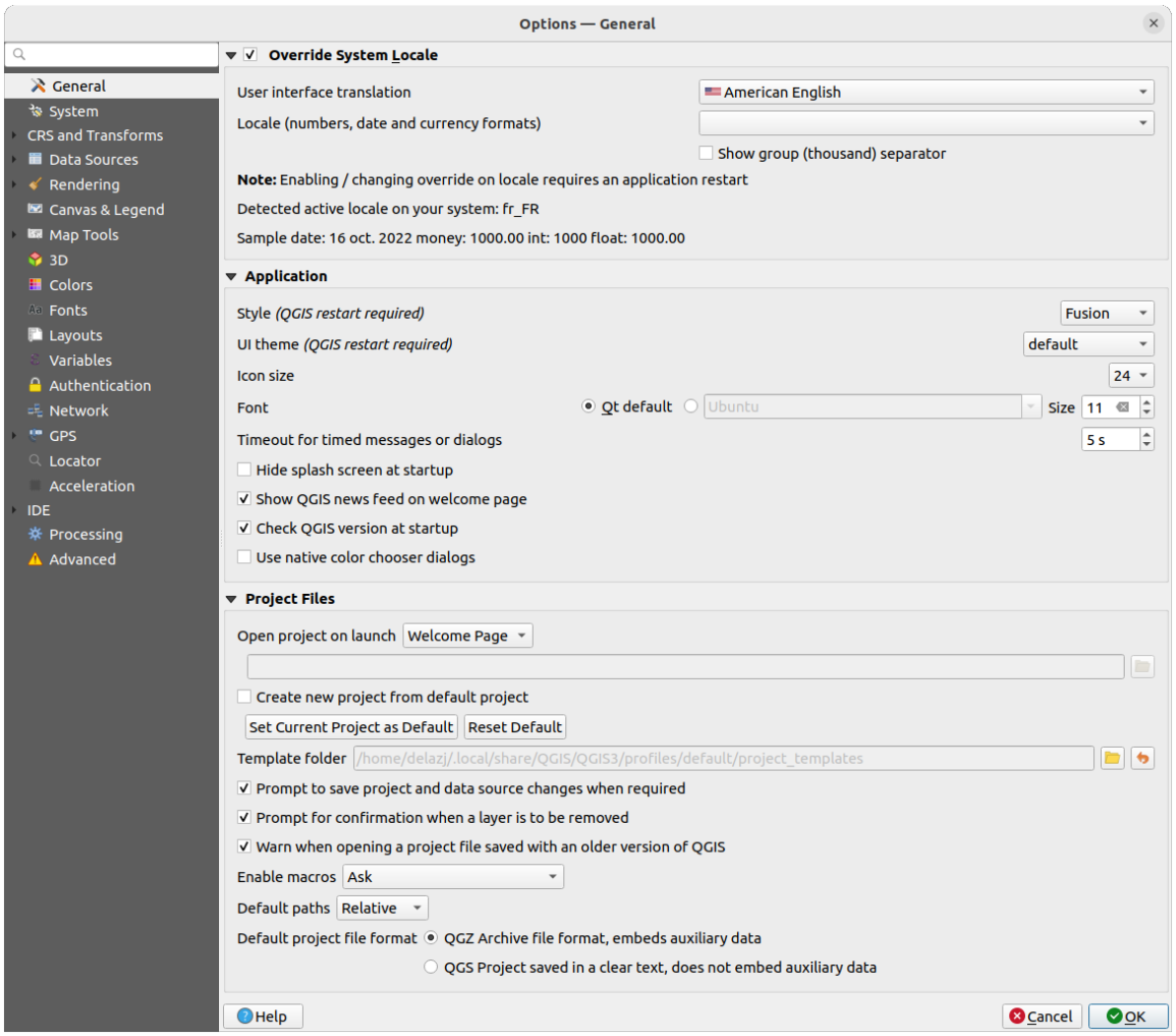


Figura 5.1: Configuración general

Anular configuración regional del sistema

Por defecto, QGIS se basa en la configuración de su sistema operativo para establecer el idioma y manipular los valores numéricos. Habilitar este grupo le permite personalizar el comportamiento.

- Seleccione en *Traducción de interfaz de usuario* el idioma para aplicar a la GUI
- Seleccione en *Local (número, formatos de fecha y moneda)* el sistema en el que deben ingresarse y representarse los valores de fecha y numéricos
- *Mostrar separador de grupo(mil)*

En la parte inferior del cuadro se muestra un resumen de la configuración seleccionada y cómo se interpretarían.

Aplicación

- Seleccione la *Estilo (requiere reinicio de QGIS)* p.ej., Los widgets se ven y se colocan en diálogos. Los valores posibles dependen de su sistema operativo.
- Define el *tema UI (requiere reinicio de QGIS)* . Puede ser “default”, “Night Mapping”, o “Blend of Gray”
- Define el *Tamaño de icono*

- Define la *Fuente* y su *Tamaño*. La fuente puede ser *Qt default* o una definida por el usuario
- Cambie el *Tiempo de espera para mensajes o diálogos cronometrados*
- *Ocultar pantalla de inicio al iniciar*
- *Mostrar noticias feed de QGIS en la página de bienvenida*: muestra una fuente de noticias feed de QGIS curada en la página de bienvenida, que le brinda una forma directa de estar al tanto de las noticias del proyecto (fecha y resumen de las reuniones de usuarios/desarrolladores, encuestas de la comunidad, anuncios de lanzamientos, varios consejos …)
- *Verifique la versión de QGIS al inicio* para mantenerse informado si se lanza una versión más nueva
- *Usar cuadros de diálogo de selección de color nativos* (ver *Selector de color*)

Los archivos de proyecto

- *Abrir proyecto al iniciar*
  - La “Página de Bienvenida” (default): puede mostrar las «Noticias» feed, la plantilla(s) del proyecto y los proyectos mas recientes (con miniaturas) del *user profile*. Ningún proyecto se abre por defecto.
  - “Nueno”: abre un nuevo proyecto, absado en una plantilla por defecto
  - “Mas recientes”: reabre el último proyecto guardado
  - y “Specifico”: abre un proyecto en concreto . Use el … botón para definir el proyecto a usar por defecto.
- *Crear nuevo proyecto desde el proyecto predeterminado*. Tiene la posibilidad de presionar *Establecer el actual proyecto como predeterminado* o sobre *Restablecer el predeterminado*. Puede navegar a través de sus archivos y definir un directorio donde se encuentra las plantillas definidas por el usuario. Esto se añadirá a *Proyecto* ► *Nueva plantilla de formulario*. Si activa primero *Crear nuevo proyecto desde proyecto predeterminado* y entonces guarde un proyecto en l la carpeta de las plantillas de proyecto.
- *Solicitar guardar los cambios del proyecto y la fuente de datos cuando sea necesario* para evitar la pérdidas de cambios realizados.
- *Pedir confirmación cuando se va a eliminar una capa*
- *Avisar al abrir un archivo de proyecto guardado con una versión anterior de QGIS*. Siempre puede abrir proyectos creados con una versión anterior de QGIS, pero una vez que se guarda el proyecto, intentar abrir con una versión anterior puede fallar debido a las características no disponibles en esa versión.
- *Habilitar código Python embebido en el proyecto* . Esta opción gestiona la ejecución de macros que se escriben para realizar una acción sobre eventos del proyecto, así como funciones personalizadas de Python que se utilizarán como expresiones. Puede elegir entre “Nunca”, “Preguntar”, “Sólo para esta sesión” y “Siempre (no recomendado)”.
- *Default paths*: define si las rutas a los archivos y las capas utilizadas en nuevos proyectos se almacenan como “Absolutas” o “Relativas” al archivo del proyecto. Esta configuración se puede sobrescribir a nivel de proyecto.
- *formato de archivo de proyecto predeterminado*
  - *archivo de formato QGZ Archive, incrustar datos auxiliares* (ver *auxiliary data*)
  - *Proyecto QGS guardado en un texto limpio, no incrustar datos auxiliares*: los datos auxiliares son almacenados en un archivo separado `.qgd` junto al archivo del proyecto.

5.1.2 Ajustes del sistema

rutas SVG

Añada o borre *Ruta(s) para buscar símbolos Scalable Vector Graphic (SVG)*. Estos archivos SVG están además disponibles para simbolizar o etiquetas las entidades o decorar tu composición de mapa.

También lea *Selector de archivos remoto o incrustado* para conocer las diferentes formas de referirse a los archivos svg en una ruta QGIS.

Rutas de complemento

Añadir o borrar *Ruta(s) para buscar librerías de complementos adicionales de C++*.

Rutas de Documentación

Añadir o quitar *Ruta(s) de documentación* a utilizar para la ayuda de QGIS. Por defecto, se añade un enlace al Manual de Usuario oficial en línea correspondiente a la versión que se está utilizando (es decir, `https://docs.qgis.org/$qgis_short_version/$qgis_locale/docs/user_manual/`). No obstante, se pueden añadir otros enlaces y priorizarlos de arriba a abajo: cada vez que se pulsa un botón *Ayuda* en un diálogo, se comprueba el enlace situado más arriba y, si no se encuentra la página correspondiente, se intenta con la siguiente, y así sucesivamente.

> **Nota**
>
> La documentación está versionada y traducida solo para QGIS Long Term Releases (LTR), lo que significa que si está ejecutando una versión regular (por ejemplo, QGIS 3.0), el botón de ayuda abrirá de manera predeterminada la siguiente página del manual de LTR (es decir, 3.4 LTR), que puede contener una descripción de las características de las versiones más recientes (3.2 y 3.4). Si no hay documentación LTR disponible, se utiliza el documento *testing*, con características de versiones más recientes y de desarrollo.

Configuraciones

Le ayuda a *Restablecer la interfaz de usuario a la configuración predeterminada(se requiere reinicio)* si realizó alguna *customization*.

Entorno

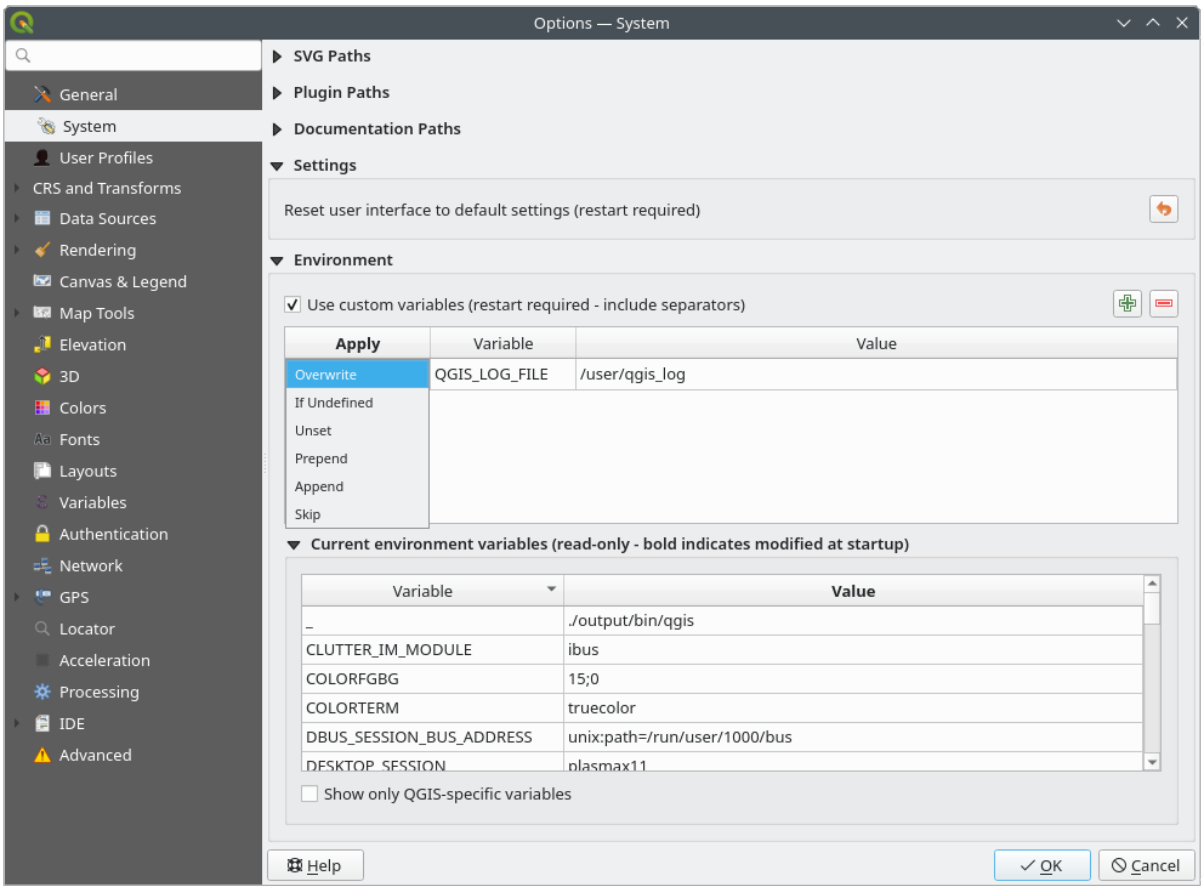


Figura 5.2: Variables de entorno del sistema

System environment variables can be viewed, and many configured, in the **Environment** group. This is useful for platforms, such as Mac, where a GUI application does not necessarily inherit the user's shell environment. It's also useful for setting and viewing environment variables for the external tool sets controlled by the Processing toolbox (e.g., GRASS), and for turning on debugging output for specific sections of the source code.

Marque *Usar variables personalizadas (requiere reinicio - incluye separadores)* y puede Añadir y Quitar variables de entorno. Por cada nuevo elemento, usted puede configurar un nombre de *Variable* , su *Valor* y el método de *Aplicar* a usar, entre los cuales:

- *Sobrescribir*: reemplaza cualquier valor anterior de la variable
- *Si no está definida*: utilice este valor para la variable si no está ya definida a un nivel superior (por ejemplo, a nivel de sistema operativo o de aplicación).
- *Desarmado*: quita la variable del entorno (el parámetro *Valor* no se usa)
- *Anteponer*: antepone el valor al valor preexistente de la variable
- *Adjuntar*: adjunta el valor al valor preexistente de la variable
- *Saltar*: se mantiene el elemento en la lista para referencias futuras pero no se utiliza

Las variables de entorno ya definidas se muestran en *Variables de entorno actuales* y es posible filtrarlas activando *MOstrar solo las variables específicas de QGIS.*

5.1.3 Configuración de los perfiles de usuario

> **Nota**
>
> Para más información sobre cómo gestionar los perfiles de usuario, lea la sección dedicada en *Trabajando con perfiles de usuario*.

5.1.4 Configuración de CRS y Transformaciones

> **Nota**
>
> Para obtener más información sobre cómo QGIS maneja la proyección de capas, lea la sección dedicada en *Trabajar con Proyecciones*.

Manejar CRS

En la pestaña *Manejar CRS* puede configurar que CRS será usado para un nuevo proyecto o capa.

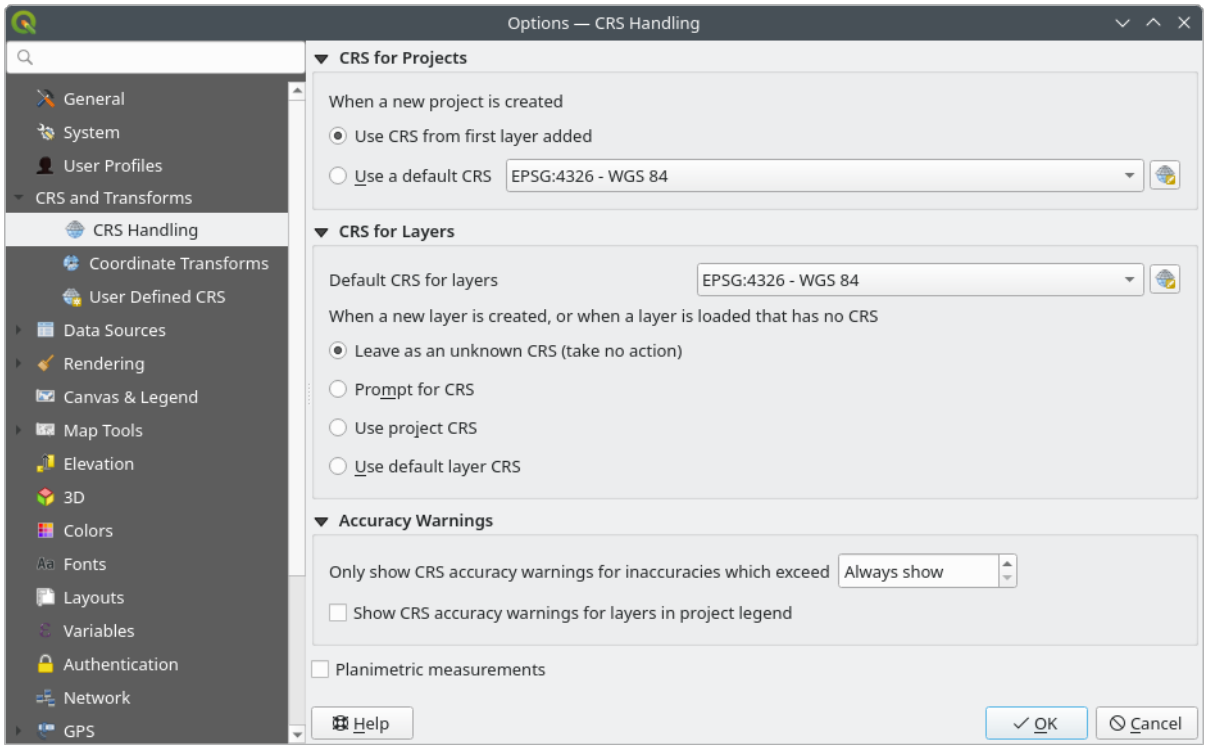


Figura 5.3: Configuraciones SRC

CRS para Proyectos

Hay una opción para ajustar automáticamente el CRS de nuevos proyectos:

- *Usar CRS de la primera capa agregada*: el CRS del proyecto se establecerá en el CRS de la primera capa cargada en él
- *Utilice un CRS predeterminado*: un CRS preseleccionado se aplica por defecto a cualquier proyecto nuevo y no se modifica al agregar capas al proyecto.

La opción se guardará para su uso en sesiones posteriores de QGIS. El Sistema de referencia de coordenadas del proyecto todavía puede anularse desde la pestaña *Proyecto* ► *Propiedades…* ► *CRS*.

CRS para Capas

:guilabel:`CRS predeterminado para capas": seleccione un CRS Preestablecido a usar cuando cree una capa

También puede definir la acción a tomar cuando se crea una nueva capa, o cuando se carga una capa sin un CRS.

- *Dejar como CRS desconocido (no hacer nada)*
- *Solicitud de CRS*
- *Usar CRS del proyecto*
- *Usar CRS de capa predeterminado*

Advertencias de Precisión

Only show CRS accuracy warnings for inaccuracies which exceed a given distance: occurs when you are explicitly creating or modifying a dataset and select a CRS based on a datum ensemble with lower accuracy. The default is to `Always show` the warning if any inaccuracy.

Mostrar aviso de precisión de CRS para capas en la leyenda del proyecto: Si está marcada, cualquier capa con un SRC con problemas de precisión (es decir, un SRC dinámico sin época de coordenadas disponible, o un SRC basado en un conjunto de datos con una inexactitud inherente que exceda el límite establecido por el usuario) tendrá el icono de advertencia en el panel *Capas* reflejando que se trata de una capa de baja precisión.

¡Esto está diseñado para su uso en ingeniería, BIM, gestión de activos y otros campos donde las imprecisiones del nivel de metro/submetro son potencialmente muy peligrosas o costosas!

Mediciones planimetricas: crea el predeterminado para *planimetric measurements* propiedad para nuevos proyectos creados..

Transformación de Coordenadas

La pestaña *Transformación de Coordenadas* le ayuda a establecer las transformaciones y operaciones de coordenadas a aplicar cuando cargue una capa a un proyecto o reproyecte una capa.

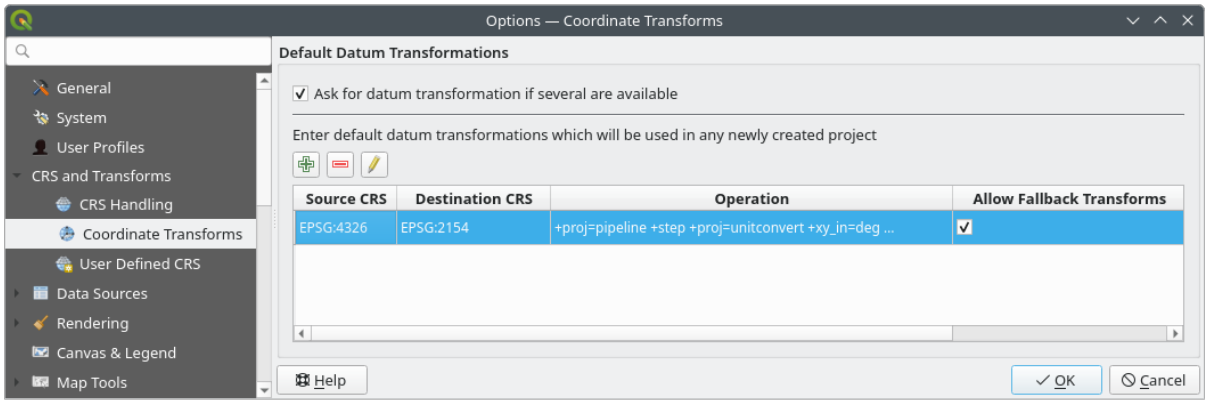


Figura 5.4: Ajustes de transformaciones

Transformaciones de Datum Predeterminadas

Aquí puede controlar si las capas proyectadas a otro CRS deberían ser:

- procesado automáticamente usando la configuración de transformaciones predeterminadas de QGIS;

- y/o mayor control por usted con preferencias personalizadas como:
  - *Pedir transformación de datos si hay varios disponibles*
  - Una lista predefinida de transformaciones de referencia para aplicar por defecto. Ver *Transformaciones de Datum* for more details.

Puede Añadir, Quitar o Editar transformaciones, que serán usadas en el proyecto recientemente creado.

CRS Definido por el Usuario

La pestaña *CRS Definido por el Usuario* le ayuda a definir un CRS personalizado que debe ser conforme al formato de cadena WKT o Proj.

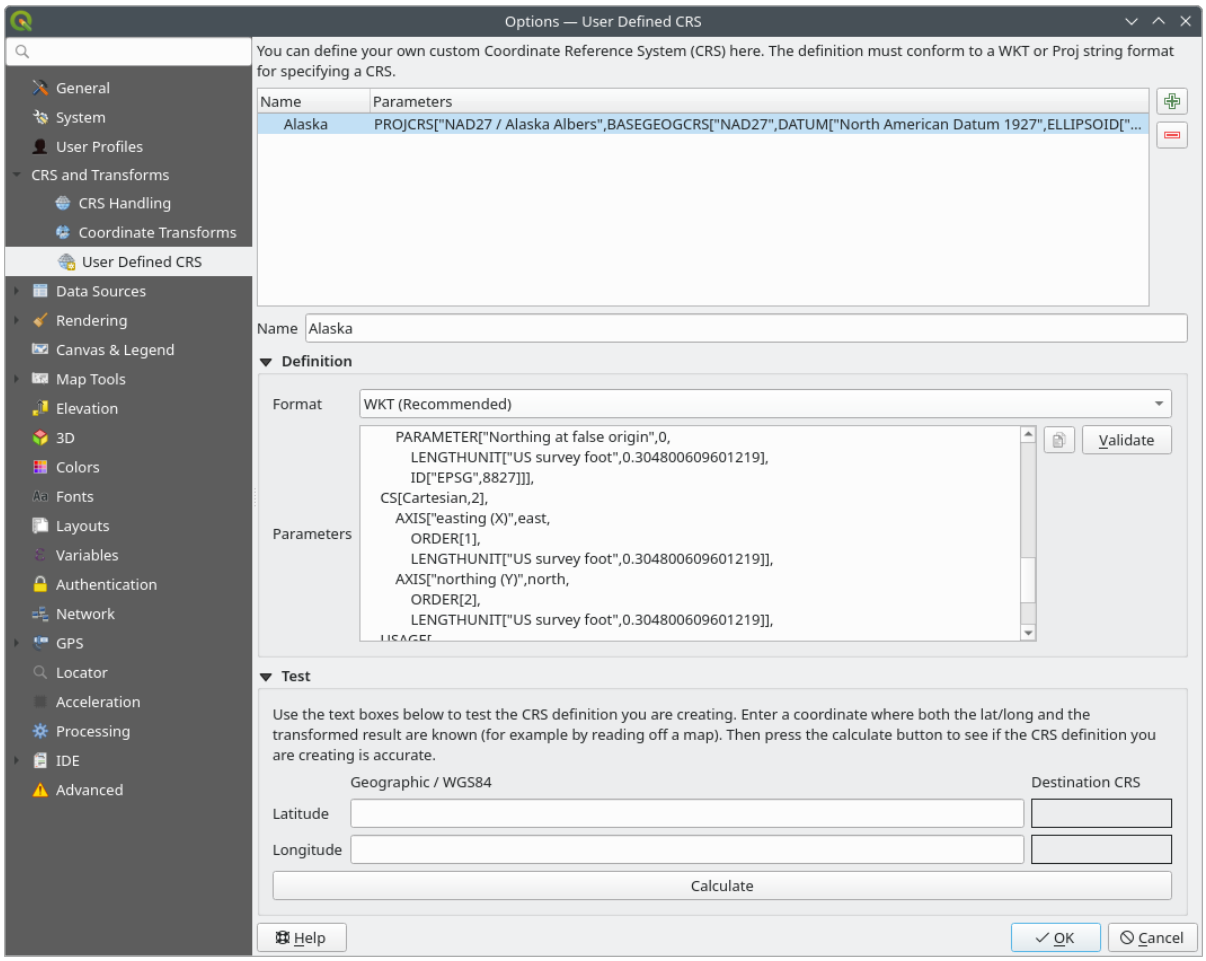


Figura 5.5: CRS Definido por el Usuario

Establezca un *Nombre* y use Añadir nuevo CRS. Si desea borrar uno existente puede usar Borrar CRS.

Definición

- ***Formato***
  - WKT (Recomendado)
  - Cadena Proj (Legado - No Recomendado)
- ***Parámetros***

- Copiar los parámetros de un CRS existente.
- *Validar* prueba si su expresión es correcta.

Probar

Aquí puede probar su definición de CRS creada por Latitud y Longitud. Use una coordenada conocida para controlar si su definición es precisa.

5.1.5 Ajustes de Fuentes de Datos

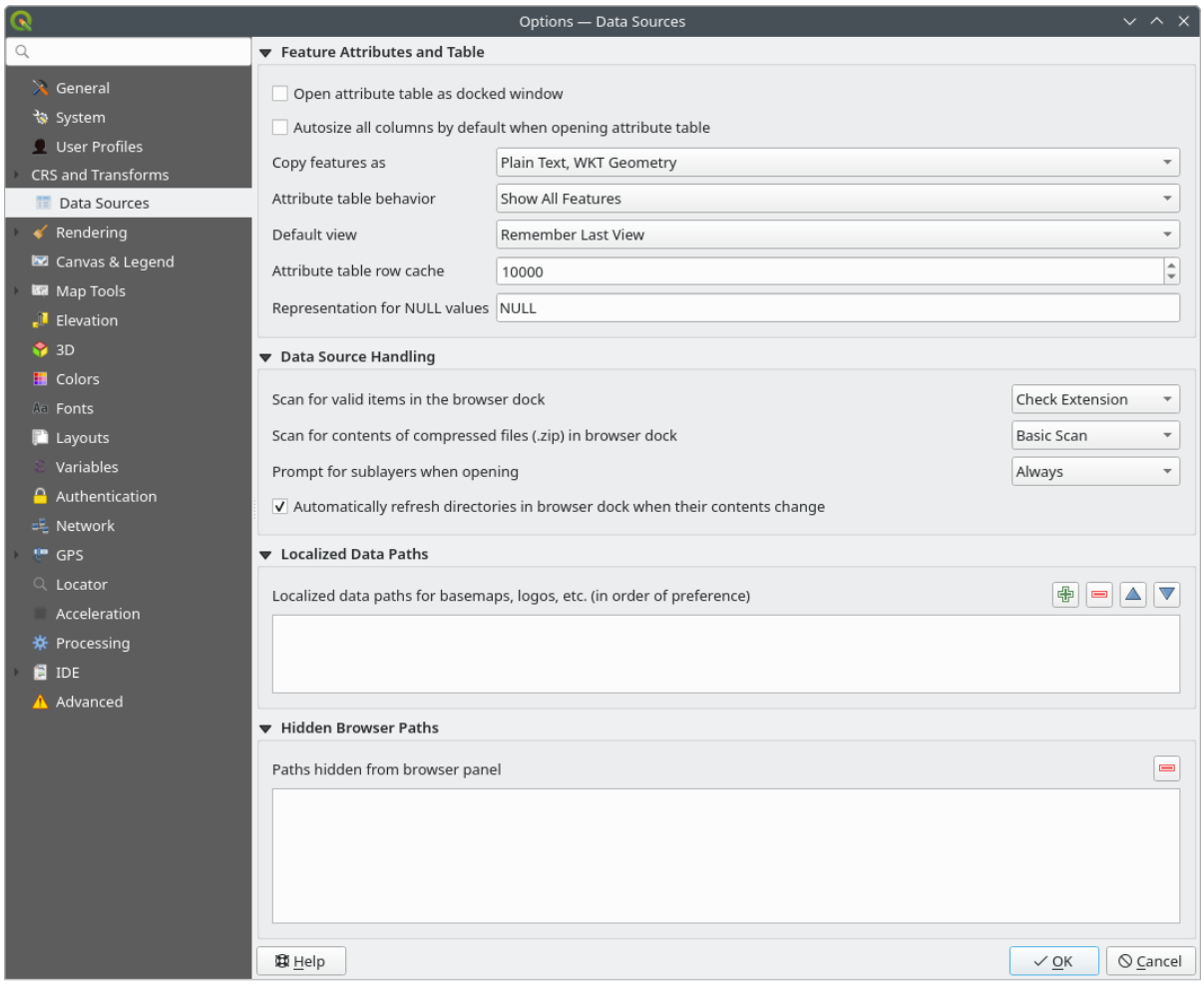


Figura 5.6: Ajustes de Fuentes de Datos

Atributos de entidades espaciales y tabla

- *Abrir la tabla de atributos como ventana acoplada*
- *Autodimensionar todas las columnas por defecto al abrir tabla de atributos.*
- *Copiar objetos espaciales como*: define el formato que se utilizará para los datos al pegar características en otras aplicaciones.
  - *Texto sin geometría*: los atributos se copian como texto pero se omite la geometría
  - *Texto, geometría WKT*: los atributos se copian como texto y la geometría se devuelve en WKT

  - *Texto, geometría WKB*: los atributos se copian como texto y la geometría se devuelve en WKB (formato hexadecimal)
  - *GeoJSON*: los atributos y la geometría se formatean como datos GeoJSON

- *Comportamiento de la tabla de atributos*  : establecer filtro en la tabla de atributos en la apertura. Hay tres posibilidades: “Mostrar todas las entidades”, “Mostrar entidades seleccionadas” y “Mostrar entidades visibles en el mapa”.
- *Vista predeterminada*: define el modo de vista de la tabla de atributos en cada apertura. Puede ser “Recordar última vista”, “Vista de tabla” o “Vista de formulario”.
- *Tabla de atributos fila caché* 

. Este caché de filas permite guardar las N últimas filas de atributos cargados para que trabajar con la tabla de atributos sea más rápido. El caché se eliminará al cerrar la tabla de atributos.
- *Representación de valores NULOS*. Aquí, puede definir un valor para los datos de campos que tienen un valor NULO.

Truco

Mejorar la apertura de la tabla de atributos de big data

Al trabajar con capas con gran cantidad de registros, la apertura de la tabla de atributos puede ser lenta ya que el diálogo solicita todas las filas de la capa. Al configurar *Atributo del comportamiento de la tabla* en **Mostrar características visibles en el mapa** hará que QGIS solicite solo las características en el lienzo del mapa actual al abrir la tabla, permitiendo una carga rápida de datos.

Tenga en cuenta que los datos en esta instancia de tabla de atributos siempre estarán vinculados a la extensión del lienzo con la que se abrió, lo que significa que al seleccionar **Mostrar todas las entidades** dentro de dicha tabla no se mostrarán nuevas entidades. Sin embargo, puede actualizar el conjunto de entidades mostradas cambiando la extensión del lienzo y seleccionando la opción **Mostrar entidades visibles en el mapa** en la tabla de atributos.

Manejo de fuente de datos

- *Escanear en busca de elementos válidos en el dock del navegador*  . Puede elegir entre “Verificar extensión” y “Verificar contenido del archivo”.
- :guilabel:`Escanear el contenido de los archivos comprimidos(.zip) en el dock del navegador”  define cuan detallada es la información del widget en la parte inferior del panel del navegador cuando se consultan dichos archivos. “No”, “Escaneo básico” y “Escaneo completo” son las posibles opciones.
- *Solicitar subcapas al abrir*. Algunos rásteres admiten subcapas — se llaman subconjuntos de datos en GDAL. Un ejemplo son los archivos netCDF — sí hay muchas variables netCDF, GDAL ve cada variable como un subconjunto de datos. La opción le permite controlar como tratar con las subcapas cuando se abre un archivo con subcapas. Tiene las siguientes opciones:
  - ‘Siempre’: Siempre preguntar (Si hay subcapas existentes)
  - ‘Si es necesario’: Preguntar si la capa no tiene bandas, pero tiene subcapas
  - ‘Nunca’: Nunca preguntar, no se cargará nada
  - ‘Cargar todo’: Nunca preguntar, pero cargar todas las subcapas
-  :guilabel:` Actualizar automáticamente los directorios en el panel del navegador cuando cambie su contenido`: Allows you to manually opt-out of monitoring directories in the *Browser* de forma predeterminada (por ejemplo, para evitar una posible ralentización debido a la latencia de la red).

Rutas de datos localizadas

Es posible usar rutas localizadas para cualquier tipo de archivos de la fuente de datos. Son una lista de rutas usadas para abstraer la ubicación de la fuente de datos. Por ejemplo,si `C:\my_maps` es listada en las rutas localizadas, una capa que tiene `C:\my_maps\my_country\ortho.tif` como fuente de datos será guardada en el proyecto usando `localized:my_country\ortho.tif`.

Las rutas son listadas por orden de preferencia, en otras palabras QGIS buscará primero el archivo en la primera ruta, luego en la segunda, etc.

Rutas ocultas del navegador

Este widget enumera todas las carpetas que eligió ocultar del *Browser panel*. Al eliminar una carpeta de la lista, estará disponible en el panel *Browser*.

Configuración de GDAL

GDAL es una biblioteca de intercambio de datos geoespaciales que soporta un gran número de formatos vectoriales y raster. Proporciona controladores para leer y (a menudo) escribir datos en estos formatos. La pestaña *GDAL* expone los controladores para formatos raster y vectoriales con sus capacidades.

Controladores vectoriales y de trama GDAL

las pestañas *Raster Drivers* y *Vector Drivers* le permite definir qué controlador GDAL está habilitado para leer y/o escribir archivos, ya que en algunos casos hay más de un controlador GDAL disponible.

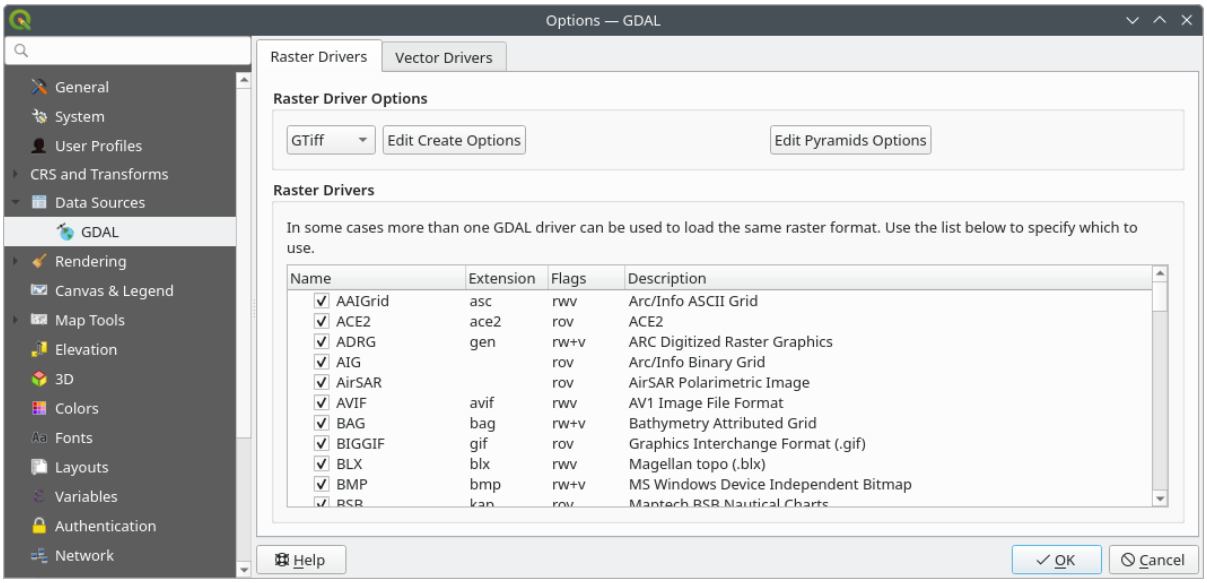


Figura 5.7: Configuración de GDAL - Controladores de trama

> **Truco**
>
> Hacer doble clic en un controlador ráster que permita el acceso de lectura y escritura (`rw+(v)`) abre el diálogo *Editar Opciones de Creación* para personalización.

Opciones de controladores de trama

Este marco proporciona modos de personalizar el comportamiento de drivers ráster que soportan el acceso lectura y escritura:

- *Edit create options*: allows you to edit or add different profiles of file transformation, i.e., a set of predefined combinations of parameters (type and level of compression, BIGTIFF support, blocks size, overview, colorimetry, alpha…) to use when outputting raster files. The parameters depend on the driver.

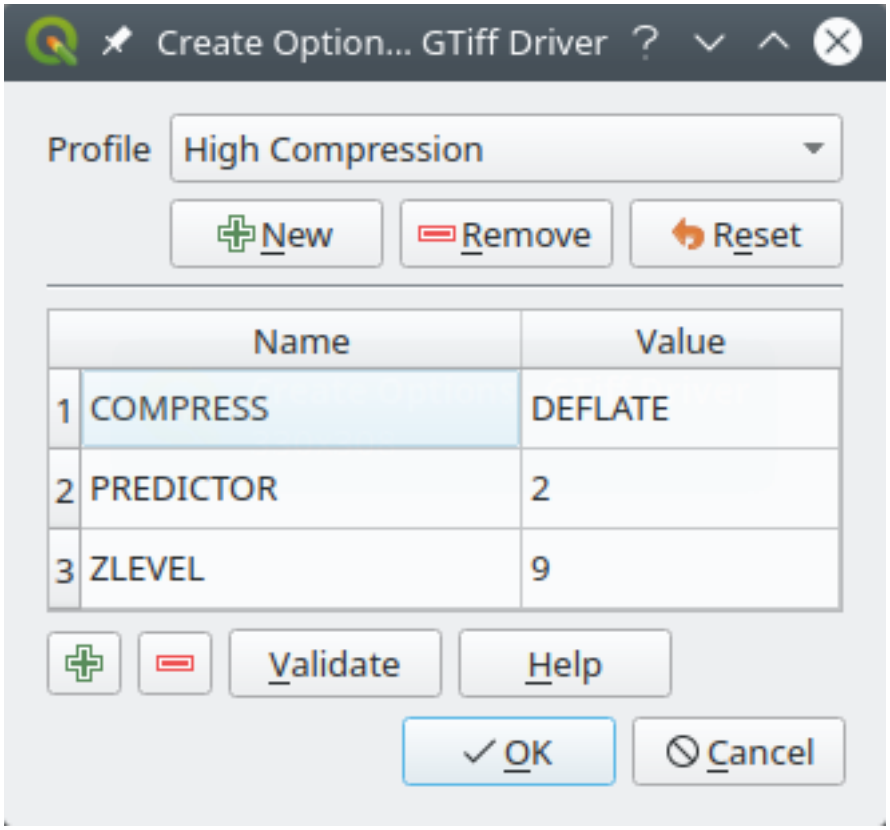


Figura 5.8: Muestra de opciones de creación de perfil (para GeoTiff)

La parte superior del cuadro de diálogo enumera los perfiles actuales y le permite agregar nuevos o eliminar cualquiera de ellos. También puede restablecer el perfil a sus parámetros predeterminados si los ha cambiado. Algunos controladores (por ejemplo, GeoTiff) tienen una muestra de perfiles con los que puede trabajar.

En la parte inferior del diálogo:

- El botón le permite añadir filas a rellenar con el nombre del parámetro y valor
- El botón borra el parámetro seleccionado
- Click en el botón *Validate* para probar que las opciones de creación entradas para el formato dado son válidas
- Utilice el botón *Ayuda* para encontrar los parámetros a utilizar, o consulte la Documentación de controladores raster de GDAL.

- *Opciones Editar Pirámides*

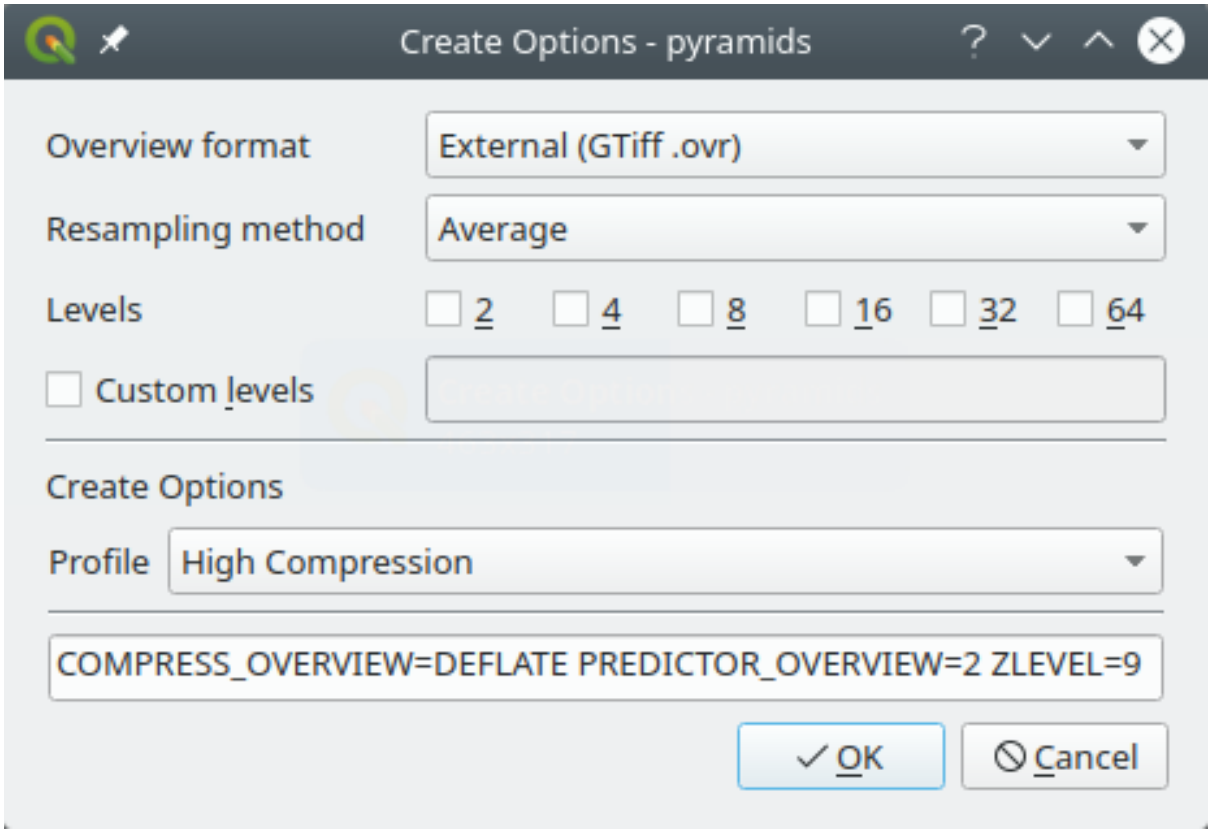


Figura 5.9: Muestra de perfil de pirámides

5.1.6 Ajustes de Renderización

La pestaña *Representación* proporciona configuraciones para controlar la representación de capas en el lienzo del mapa.

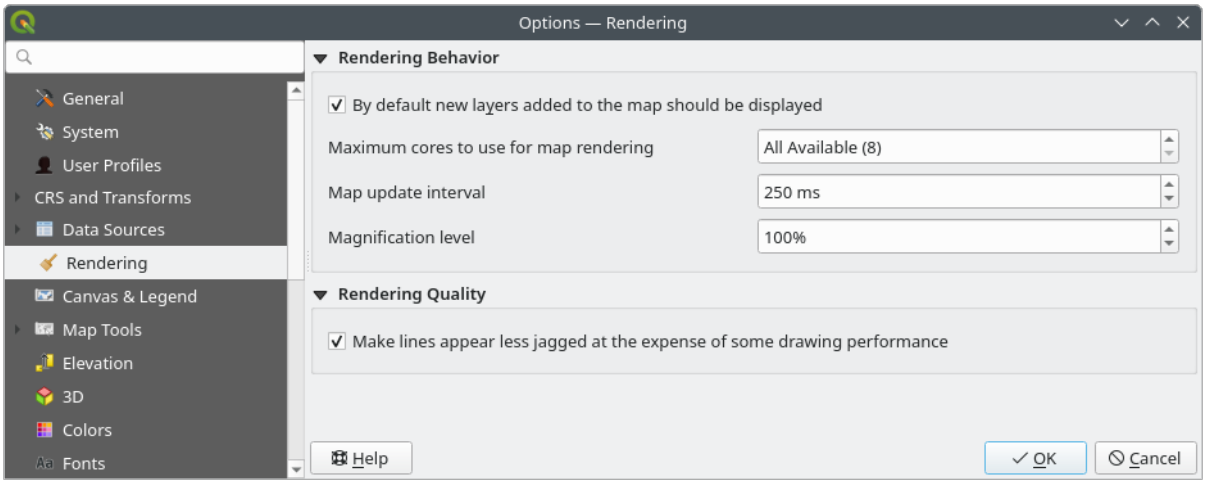


Figura 5.10: Ajustes de renderizado

Comportamiento de Renderizado

- *Por defecto, deben mostrarse las nuevas capas agregadas al mapa*: desmarcar esta opción puede ser útil al cargar varias capas para evitar que cada capa nueva se represente en el lienzo y ralentizar el proceso
- Establecer los *Núcleos máximos para usar en la representación de mapas*
- El lienzo del mapa se representa en el formado en una imagen separada y en cada *Intervalo de actualización del mapa* (de modo predeterminado a 250 ms), el contenido de esta imagen (fuera de la pantalla) será tomada para actualizar la representación visible en la pantalla. Sin embargo, sí el renderizado finaliza mas rápido que esta duración será mostrado instantáneamente.
- *Nivel de magnificación* (vea *magnifier*)

Calidad de Representación

- *Hacer que las líneas se muestren menos quebradas a expensas del rendimiento de la representación*

Configuración de renderizado vectorial

La pestaña *Vector* contiene ajustes específicos para el renderizado de las capas vectoriales.

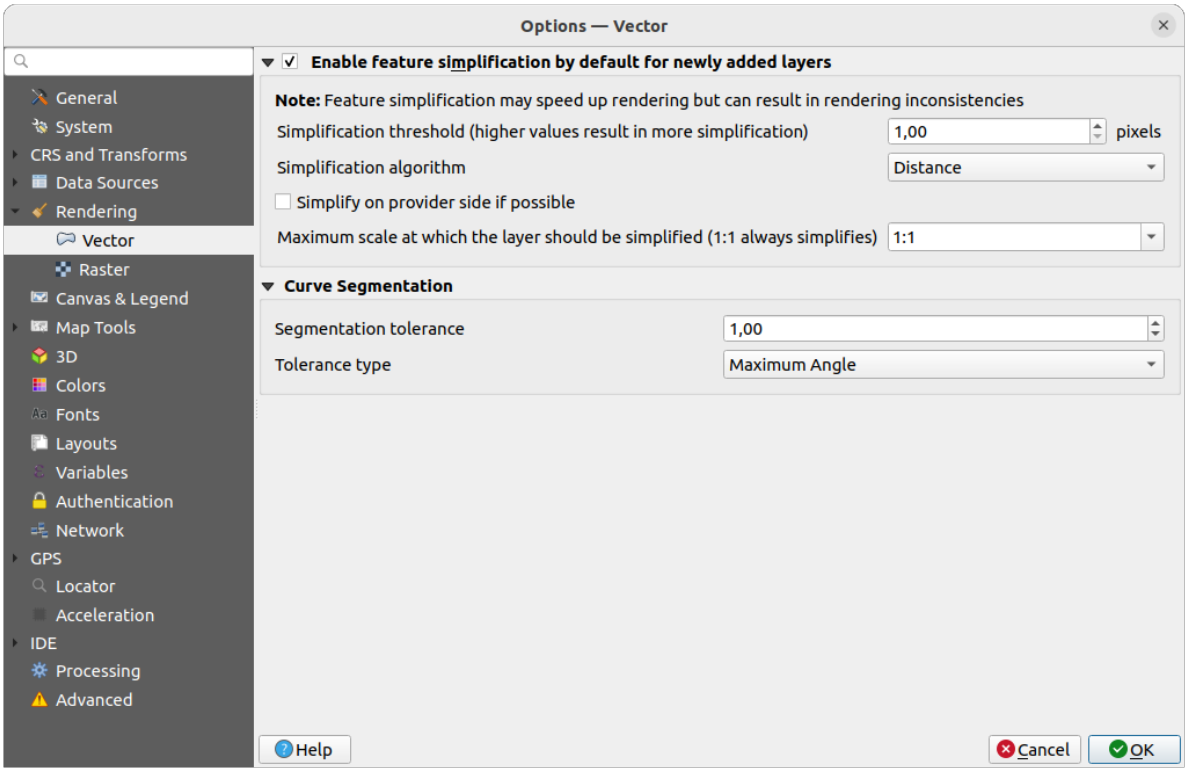


Figura 5.11: Configuración de renderizado vectorial

- *Habilitar la Simplificación de Funciones de Forma Predeterminada para las Capas Recién Agregadas*: simplifica la geometría de las funciones (menos nodos) y como resultado se visualizan más rápidamente. Tenga en cuenta que esto puede causar incoherencias en la representación. Los ajustes disponibles son:
  - *Umbral de simplificación (valores mas altos llevan a más simplificación)*
  - *Algoritmo de simplificación*: Esta opción realiza una simplificación local «sobre la marcha» de los elementos y acelera la representación de la geometría. No cambia la geometría obtenida de los proveedores de datos. Esto es importante cuando se tienen expresiones que utilizan la geometría de la característica (por ejemplo, cálculo de área) - asegura que estos cálculos se realizan sobre la geometría original, no sobre la simplificada. Para este propósito, QGIS proporciona tres algoritmos: “Distancia” (por defecto), “SnapToGrid” y “Visvalingam”.

  - *Simplify on provider side if possible*: the geometries are simplified by the provider (PostgreSQL, Oracle…) and unlike the local-side simplification, geometry-based calculations may be affected
  - *Escala máxima en la que la capa debería ser simplificada (1:1 simplifica siempre)*

> **Nota**
>
> Además de la configuración global, se puede configurar la simplificación de características para cualquier capa específica desde su menú *Propiedades de Capa* ► *Renderizado*

- *Segmentación de Curvas*
  - *Tolerancia de segmentación*: esta configuración controla la forma en que se representan los arcos circulares. **EL menor** ángulo máximo (entre los dos vértices consecutivos y el centro de la curva, en grados) o la diferencia máxima (distancia entre el segmento de los dos vértices y la línea de la curva, en unidades de mapa), los segmentos **más rectos** se usarán durante el renderizado.
  - *Tipo de Tolerancia*: puede ser *Máximo ángulo* o *Maxima diferencia* entre aproximación y curva.

Configuración de renderizado ráster

La pestaña *Ráster* contiene configuraciones específicas para renderizar capas ráster.

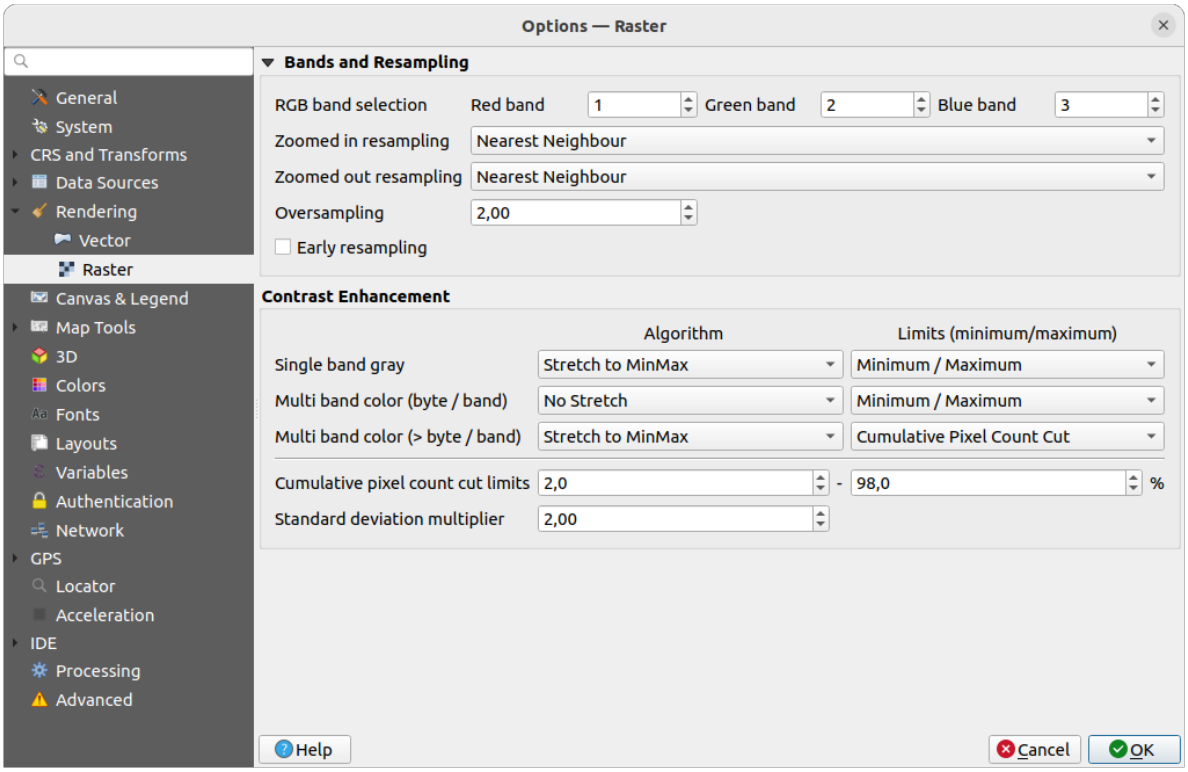


Figura 5.12: Configuración de renderizado ráster

Bajo *Bandas y Remuestreo*:

- Con *Selección de la banda RGB*, puede definir el numero para la banda Roja, Verde y Azul.
- Se pueden definir los métodos *Remuestreo con zoom* y *Remuestreo con zoom*. Para *Remuestreo con zoom* se puede elegir entre tres métodos de remuestreo: “Vecino más cercano”, “Bilineal (núcleo 2x2)” y “Cúbico (núcleo 4x4)”. Para *Remuestreo con zoom* puede elegir entre “Vecino más cercano” y “Promedio”. También puede

establecer el valor de *Sobremuestreo* (entre 0.0 y 99.99 - un valor grande significa más trabajo para QGIS - el valor predeterminado es 2.0).

- *Remuestreo temprano*: permite calcular la representación ráster a nivel de proveedor donde la resolución de la fuente se conoce y asegura un mejor zoom en el renderizado con el estilo personalizado QGIS. Realmente conveniente para rásteres de teselas cargados usando un *método de interpretación*. La opción se puede establecer también a nivel capa (:guilabel:`Propiedades de simbología)

Las opciones *Mejora de Contraste* se pueden aplicar a *Banda única gris*, *Color multibanda (byte/banda)* o *Color multibanda (>byte/banda)*. Para cada uno puede establecer:

- el *Algoritmo* a usar, cuyos valores pueden ser “Sin estiramiento”, “Estirar a MinMax”, “Estirar y recortar a MinMax” o “Recortar a MinMax”
- Los *Limites(minimo/maximo)* a plicar, con valores como “Corte de recuento de píxeles acumulativo”, “Mínimo/Máximo”, “Media +/- desviación estándar”.

Las opciones de *Mejora de Contraste* también incluyen:

- *Límite para corte del conteo acumulativo de píxeles*
- *Multiplicador de la desviación estándar*

5.1.7 Ajustes de Lienzo y Leyenda

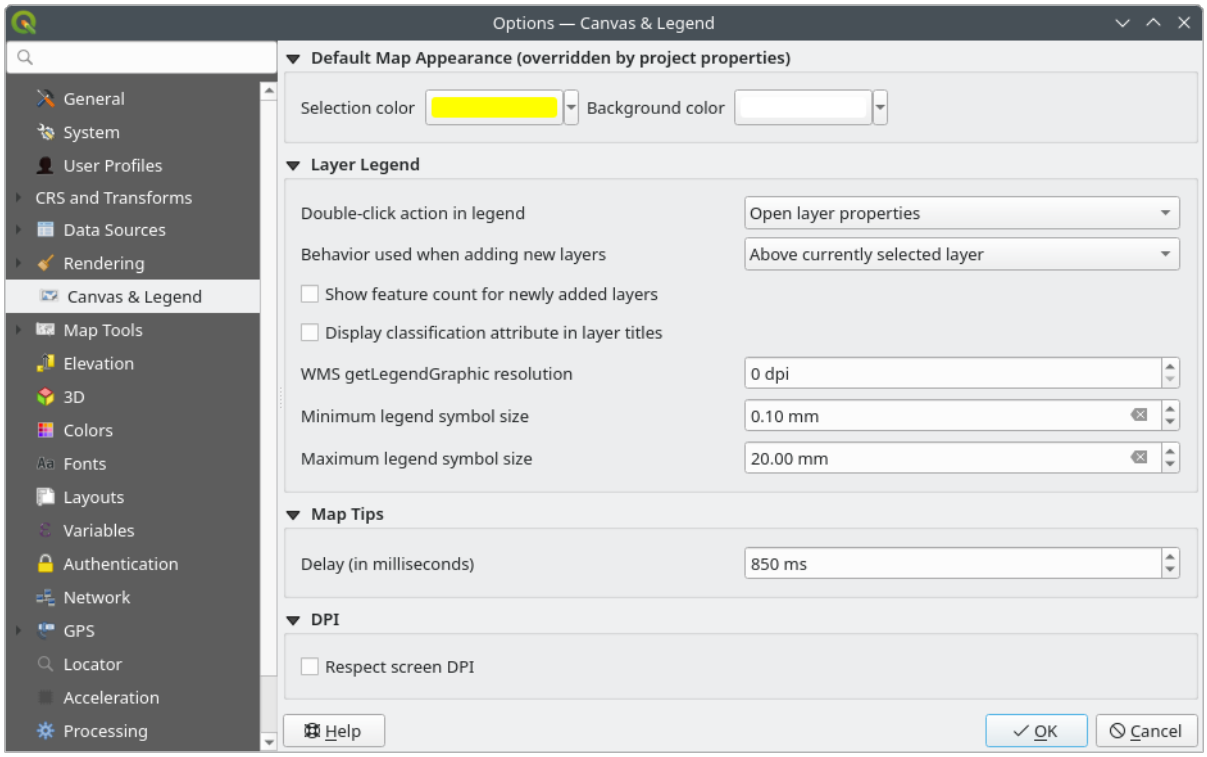


Figura 5.13: Configuración de Lienzo y Leyendas

Estas propiedades le permiten establecer:

- **Aspecto Preestablecido del mapa (anulado por las propiedades del proyecto)**: el *Color de selección* y :guilabel:` Color de fondo`.
- **Leyenda de Capa** interacción:
  - *Acción de doble clic en la leyenda*: si al hacer doble clic en una capa se debe *Abrir propiedades de capa* (por defecto), *Abrir tabla de atributos* o *Abrir muelle de estilo de capa*.

  - *Comportamiento utilizado al añadir nuevas capas*: determina dónde se colocan las capas en el panel *Capas* cuando se cargan en QGIS. Puede ser:
    - *Sobre la capa seleccionada actualmente*.
    - *Siempre en la parte superior del árbol de capas*
    - *Índice óptimo dentro del grupo actual del árbol de capas*: A diferencia de las otras opciones que ordenan las nuevas capas entre ellas y las colocan como una pila en la ubicación deseada, esta opción extiende la lógica de ordenación a todo el árbol de capas (o al grupo activo) e inserta las nuevas capas de forma «óptima» asegurando que las capas de puntos se sitúan encima de las capas de puntos, seguidas de las capas de líneas encima de las líneas, seguidas de las capas de polígonos, etc.
  - *Mostrar recuento de características para capas recién agregadas*: muestra en el panel *Capas* el número de entidades junto al nombre de la capa. El recuento de características de las clases, sí las hay, también se muestra. Puede con el botón derecho sobreuna capa poner/quitar el recuento de características.
  - *Mostrar nombres de atributos de clasificación* en el panel Capas, por ejemplo, al aplicar un renderizador categorizado o basado en reglas (véase menú_de_estilo_vectorial para más información).
  - la *WMS getLegendGraphic Resolution*
  - *Mínimo* y *Máximo tamaño de símbolo de leyecda* para controlas en tamaño de símbolo mostrado en el panel *Capas*
- la visualización *Retardo* en milisegundos de las capas *map tips*
- Si QGIS debe *Respetar los PPP de la pantalla*: Si se activa, QGIS intentará mostrar el lienzo con una escala físicamente precisa en pantalla, dependiendo de los PPP físicos del monitor. La simbología con el tamaño de pantalla especificado también se mostrará con precisión, por ejemplo, un símbolo de 10mm se mostrará como 10mm en pantalla. Sin embargo, los tamaños de fuente de las etiquetas en el lienzo pueden diferir de los de la interfaz de usuario de QGIS u otras aplicaciones. Si esta opción está desactivada, QGIS utilizará los PPP lógicos del sistema operativo, que serán coherentes con los de otras aplicaciones del sistema. Sin embargo, la escala del lienzo y el tamaño de la simbología pueden ser físicamente inexactos en la pantalla. En particular, en pantallas de alta ppp, es probable que la simbología aparezca demasiado pequeña.

  Para una mejor experiencia, se recomienda habilitar *Respect screen DPI*, especialmente cuando utilice múltiples o diferentes monitores y prepare mapas de alta calidad visual. Deshabilitar *Respect screen DPI* generará una salida que puede ser más adecuada para el mapeo destinado solo para uso en pantalla, especialmente donde los tamaños de fuente deben coincidir con otras aplicaciones.

> **Nota**
>
> La representación en los diseños no se ve afectada por la configuración de *Respect screen DPI*; siempre respeta el DPI especificado para el dispositivo de salida de destino. También tenga en cuenta que esta configuración utiliza la pantalla física DPI según lo informado por el sistema operativo, lo que puede no ser preciso para todas las pantallas.

5.1.8 Ajustes de herramientas de mapa

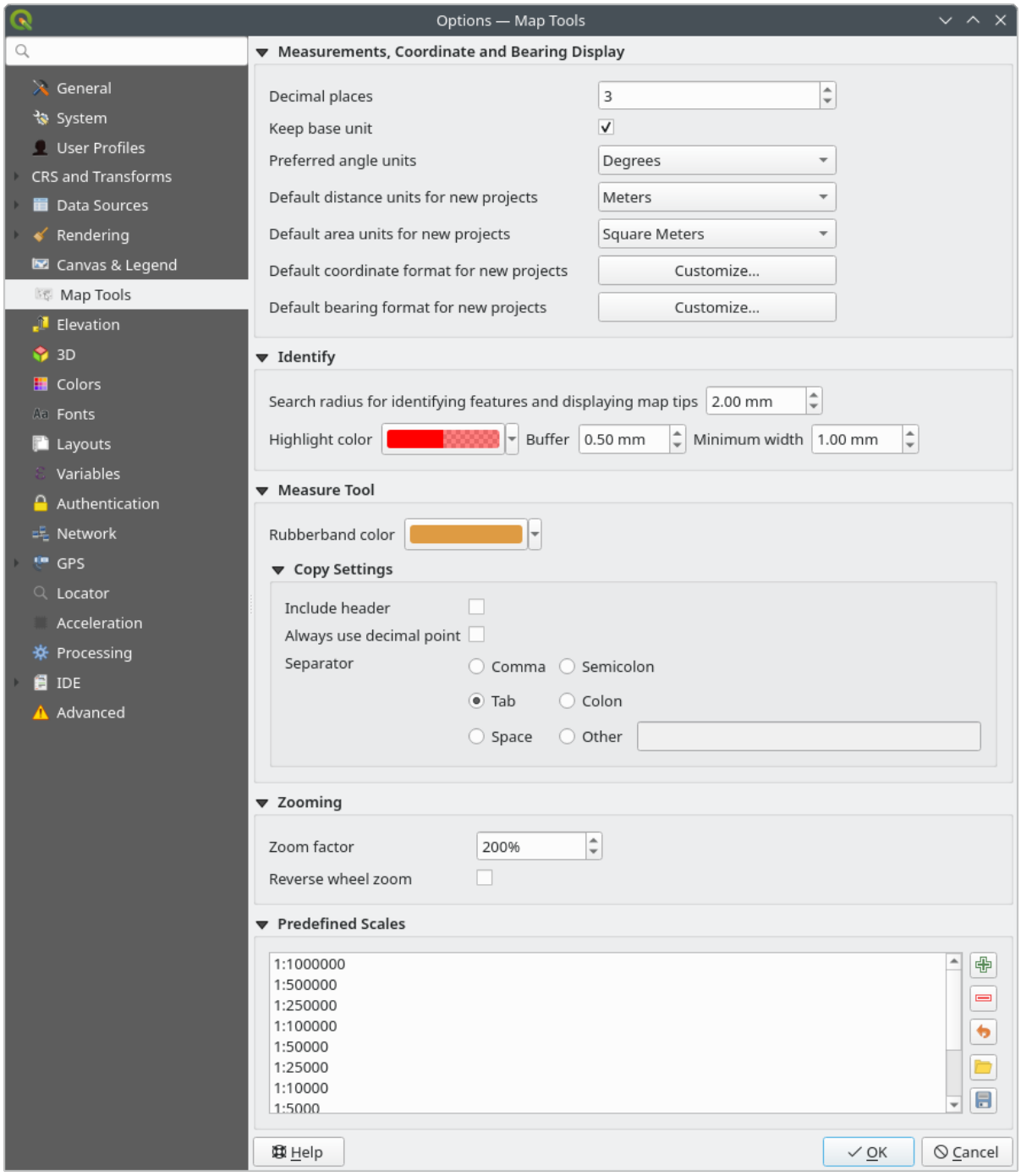


Figura 5.14: Configuración de herramientas de mapa

Measurements, Coordinate and Bearing Display

This section provides ways to configure default units parameters:

- Definir *Lugares decimales*
- *mantener unidades de base* para no convertir automáticamente números largos (p.ej., metros a kilometros)
- *Unidades angulares preferidas*: las opciones son *Grados*, *Radianes*, *Gon/gradianes*, *Minutos de arco*, *Segundos de arco*, *Giros/revoluciones*, miliradianes (definición SI) o mil (OTAN/definición militar)

- *Default distance units for new projects*: options are “Meters”, “Kilometers”, “Feet”, “Yards”, “Miles”, “Nautical Miles”, “Centimeters”, “Millimeters”, “Inches”, “Degrees” or “Map Units” (automatically matches the project’s CRS units)
- *Preferred area units for new projects*: options are “Square meters”, “Square kilometers”, “Square feet”, “Square yards”, “Square miles”, “Hectares”, “Acres”, “Square nautical miles”, “Square centimeters”, “Square millimeters”, “Square inches”, “Square degrees” or “Map Units” (automatically matches the project’s CRS units)
- *Formato predeterminado de coordenadas para nuevos proyectos*, como se muestra en la caja *Coordenadas* en la barra de estado de QGIS y en los resultados de la herramienta *Derived* section of the  Identificar características
- *Formato de rodamiento predeterminado para nuevos proyectos*, como se muestra en la barra de estado para la dirección de desplazamiento panorámico del lienzo del mapa y por la herramienta  Medir rumbo.

The global options can be overridden at the project level, *here* for measurement units and *here* for coordinate and bearing formatting.

Identify tool

This group offers some options regarding the behavior of the *Identify tool*.

- :guilabel:El *radio de búsqueda para identificar entidades y mostrar sugerencias de mapa* es una distancia de tolerancia dentro de la cual la herramienta de identificación representará los resultados, siempre que haga click dentro de esta tolerancia.
- *Color de resaltado* le permite elegir con que color se deben resaltar las entidades identificadas.
- *Zona de influencia* determina una distancia de zona de influencia a ser representada desde el borde del resaltado identificador.
- *Mínima anchura* determina como de gruesa debe ser la linea exterior de un objeto resaltado.

Herramienta de medición

- Definir *Color de la banda de medida* para herramienta de medida

Configuración de la Copia de la Herramienta de Medición

Estos ajustes le permiten controlar el comportamiento del botón *Copiar* de la ventana **|medir|** Línea de medida. Puede elegir  *Incluir cabecera* para mantener los nombres de las columnas, y también puede seleccionar *Separador* de su elección. También puede elegir  *Usar siempre punto decimal* para mantener sus coordenadas copiadas en el portapapeles con un punto como separador decimal, incluso si su configuración de idioma en las opciones de QGIS está configurada para usar una coma. En este caso, si no tiene marcada la opción *Usar siempre punto decimal*, no podrá seleccionar la coma como separador de campos.

Mover y zoom

- Define un *Factor de Zoom* para las herramientas zoom o rueda del ratón
-  *Invertir el zoom de la rueda* permite ajustar la dirección de desplazamiento de la rueda del ratón.

Escalas predefinidas

Aquí encontrará una lista de escalas predefinidas para mostrar por defecto en los widgets desplegables relacionados con la escala, como la barra de estado *Scale*, el selector de escalas de visibilidad o los ajustes secundarios de la vista del mapa 2D,… Con los botones  y  puedes añadir o eliminar tus escalas personales. También puede importar o exportar escalas desde/a un archivo `.XML`. Tenga en cuenta que aún tiene la posibilidad de eliminar sus cambios y volver a la lista predefinida.

Desde el cuadro de diálogo de propiedades del proyecto, también puede establecer su propia lista de escalas, anulando esta global en los widgets.

Ajustes de digitalización

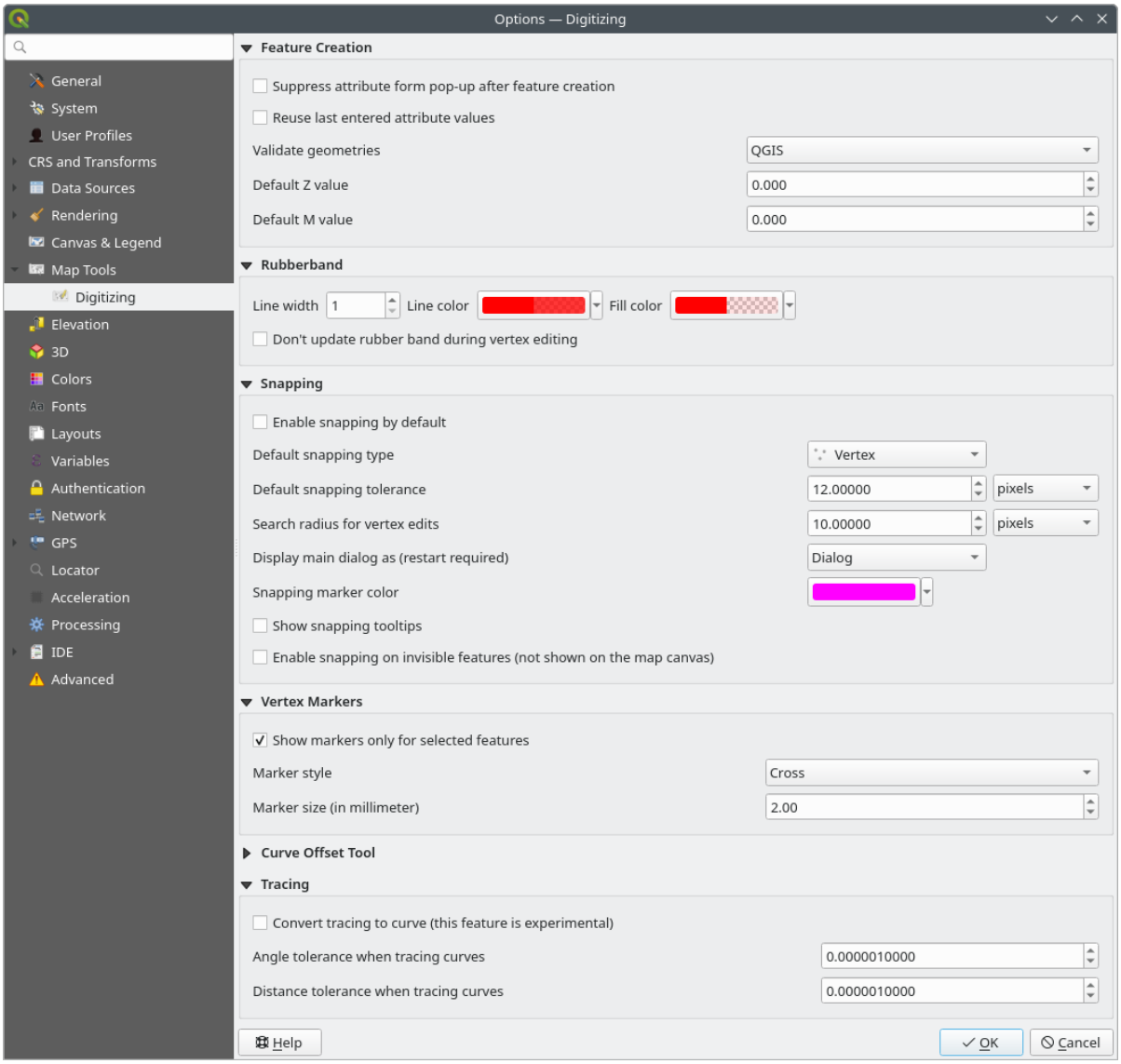


Figura 5.15: Ajustes de digitalización

Esta pestaña te ayuda a configurar ajustes generales al *editing vector layer* (atributos y geometría).

Creación de entidades espaciales

- *Suprime la ventana emergente del formulario de atributo después de la creación de entidades*: esta opción se puede anular en el cuadro de diálogo de propiedades de cada capa.
- *Reutilizar los últimos valores de atributo introducidos*: recuerda el último valor utilizado de cada atributo y lo utiliza por defecto para la siguiente característica que se digitalice. Funciona por capa. Este comportamiento también puede controlarse por campo (véase *Configurar el comportamiento del campo*).
- *Validate geometries*. Editing complex lines and polygons with many nodes can result in very slow rendering. This is because the default validation procedures in QGIS can take a lot of time. To speed up rendering, it is possible to select GEOS geometry validation or to switch it off. GEOS geometry validation is much faster, but the disadvantage is that only the first geometry problem will be reported.

  Tenga en cuenta que, según la selección, los informes de errores de geometría pueden diferir (ver *Tipos de mensajes de error y sus significados*)

- *Valor Z por defecto* para usar al crear nuevas características 3D.

Banda de medición

- Definir banda elástica *Ancho de línea*, *Color de línea* y *Color de relleno*.
- *No actualizar la banda elástica durante la edición de vértices*.

Autoensamblado

- *Activar ajuste por defecto* activa el ajuste cuando se abre un proyecto
- Definir *Modo de ajuste predeterminado* (“Vértice”, “Segmento”, “Centroide”, “Mitad de segmentos”, “Puntos finales de línea”, “Área”)
- Definir *Tolerancia de autoensamblado predeterminado* en unidades de mapa o píxeles
- Definir el *Radio de búsqueda para edición de vértices* en unidades de mapa o píxeles
- *Mostrar cuadro de diálogo principal como (se requiere reiniciar)*: establezca si el cuadro de diálogo Autoensamblado avanzado debe mostrarse como “Diálogo” o “Dock”.
- *Color de marcador de autoensamblado*
- *Mostrar información sobre herramientas de ajuste* como el nombre de la capa cuya entidad está a punto de ajustar. Útil cuando se superponen varias funciones.
- *Habilitar el ajuste de características invisibles (no se muestra en el lienzo del mapa)*

Marcar vértices

- *Mostrar marcadores sólo para los objetos espaciales seleccionados*
- Definir vértice *Estilo de marcador* (“Cruz” (predeterminado), “Círculo semitransparente” o “Nada”)
- Definir vértice *Tamaño del marcador (en milímetros)*

Herramienta de desplazamiento de curva

Las siguientes opciones se refieren a la herramienta Curva de desplazamiento en *Digitalización avanzada*. A través de los distintos ajustes, es posible influir en la forma del desplazamiento de la línea.

- *Estilo de unión*: especifica si se deben utilizar uniones “Redondas”, “Ingletes” o “Bisel” al desplazar las esquinas.
- *Segmentos de cuadrante*: controla el número de segmentos de línea que se utilizarán para aproximar un cuarto de círculo al crear desplazamientos redondeados.
- *Límite de inglete*: Establece la distancia máxima desde la geometría de desplazamiento que se utilizará al crear una unión a inglete como factor de la distancia de desplazamiento (sólo aplicable para el estilo de unión a inglete).
- *Estilo de fin de línea*: controla cómo se tratan los finales de línea; pueden ser “Redondo”, “Plano” o “Cuadrado”.

Trazado

Activando *Convertir trazado a curva* puede crear segmentos curvos mientras digitaliza. Tenga en cuenta que su proveedor de datos debe admitir esta función.

5.1.9 Ajustes de elevación

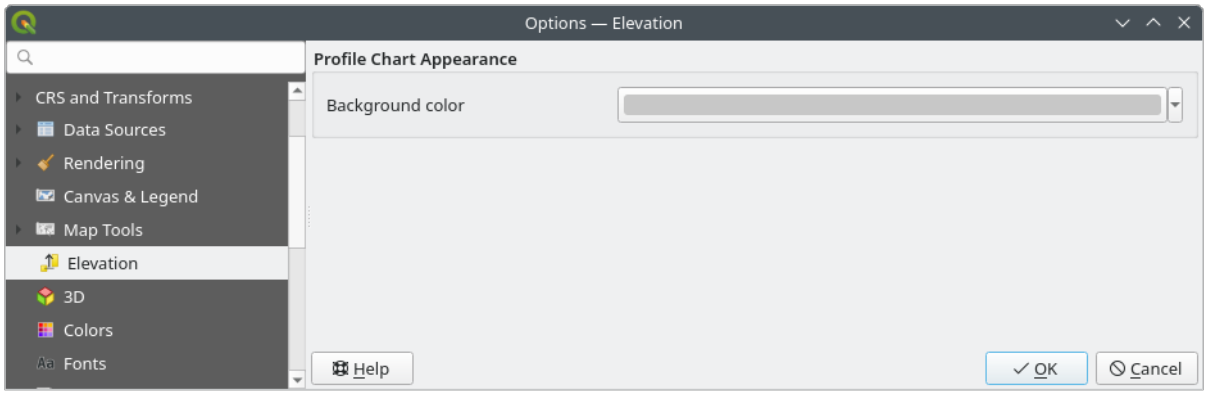


Figura 5.16: Ajustes de elevación

En el menú *Elevación*, puede establecer un color específico para utilizarlo como *Color de fondo* para los perfiles de *elevación*. Esto puede hacer que el gráfico sea más legible para ciertos conjuntos de datos, por ejemplo, nubes de puntos con coloración RGB, donde el color de fondo por defecto es demasiado similar a los colores de los puntos para ser fácilmente discernible. Si no se modifica, los perfiles de elevación seguirán mostrándose con el color de fondo estándar del sistema.

5.1.10 Configuración 3D

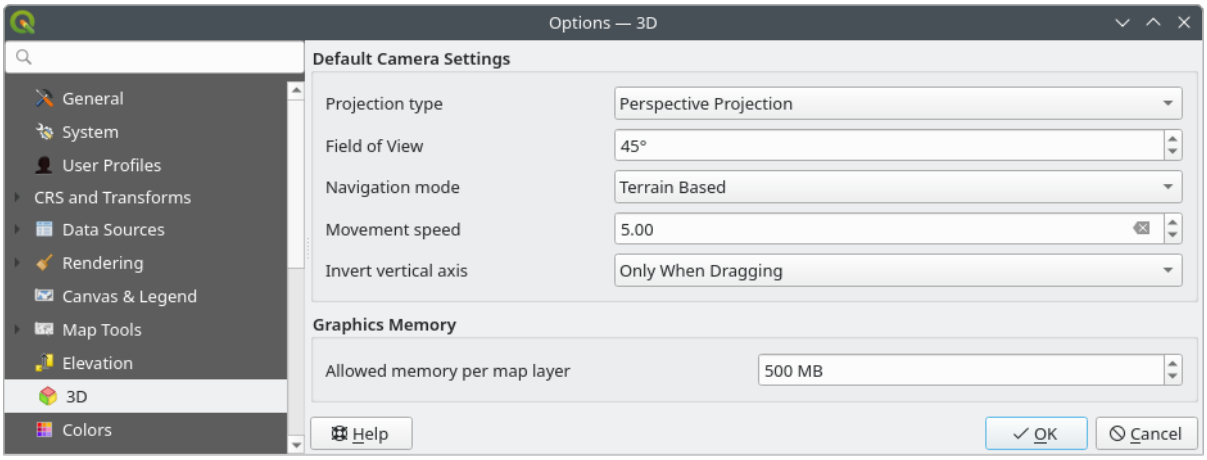


Figura 5.17: Configuración 3D

El menú *3D* le ayuda a configurar algunos parámetros predeterminados para utilizar en cualquier vista de *3D Mapa*. Estos pueden referirse a:

- *Configuración predeterminada de la Cámara*:
  - *Projection type*: permitiendo ver la escena 3D en un:
    - *Perspective projection* (default): Líneas paralelas parecen encontrarse en la distancia. Los objetos parecen encogerse cuanto más lejos están de la cámara.
    - o una *Orthogonal projection*: Las líneas paralelas parecen paralelas. Los objetos aparecen del mismo tamaño independientemente de la distancia.

  - de la cámara *Field of view*: solo relevante en la proyección en perspectiva, especifica el campo de visión vertical actual en grados y determina qué parte de la escena es visible para la cámara. El valor predeterminado es 45°.
  - *Navigation mode*: proporciona diferentes medios para interactuar con la escena 3D. Los modos disponibles son:
    * *Terrain based*: la cámara sigue una posición fija en la superficie del terreno a medida que se navega por la escena.
    * *Modo caminar (primera persona)*

    Según el modo seleccionado, los *navigation commands* difieren.
  - *Velocidad de movimiento*
  - *Invert vertical axis*: Controla si los movimientos del eje vertical deben invertirse de su comportamiento normal. Sólo afecta al movimiento en el *Modo andar*. Se puede configurar para:
    * *Nunca*
    * *Only when dragging*: hace que el movimiento vertical se invierta solo cuando se realiza una rotación de la cámara haciendo clic y arrastrando
    * y *Always*: hace que los movimientos se inviertan cuando se hace clic y se arrastra y cuando el movimiento de la cámara se bloquea en el cursor (a través de una pulsación de tecla ~)

- En *Memoria gráfica*, la opción *Memoria permitida por capa* permite establecer la configuración del límite de memoria de la GPU en cada capa. Esto es útil para los usuarios que utilizan grandes escenas 3D que agotan los recursos disponibles de memoria de la GPU. Cuando se alcanza un límite, también se muestra una advertencia, lo que debería ayudar en la solución de problemas de grandes escenas.

5.1.11 Configuración de colores

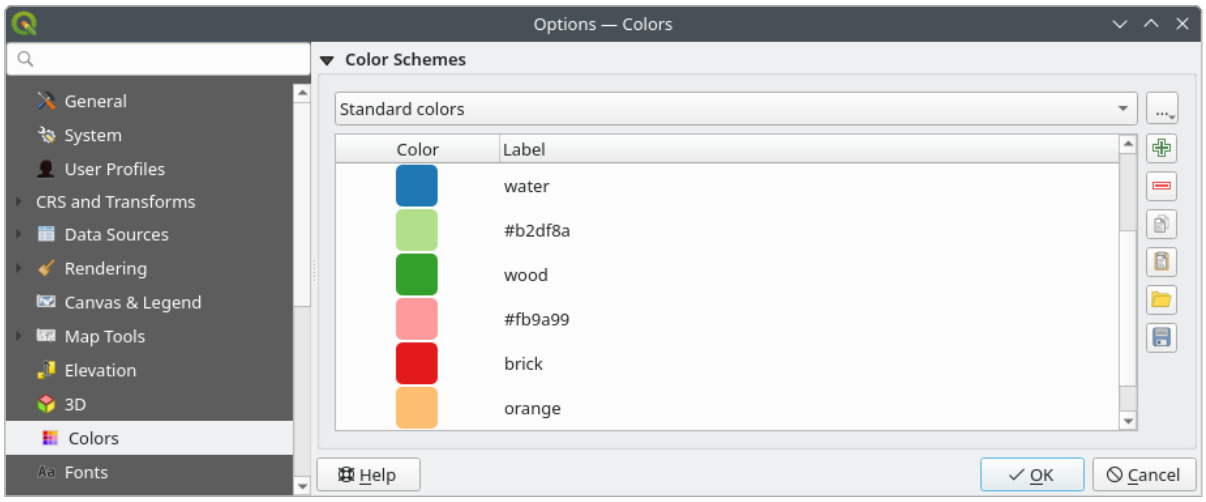


Figura 5.18: Configuración de colores

Este menú le permite crear o actualizar paletas de colores utilizados en toda la aplicación en el *color selector widget*. Puede elegir desde:

- *Colores recientes* mostrando los colores usados recientemente
- *Colores Estándar*, la paleta de colores por defecto
- *Colores de Proyecto*, un conjunto específico de colores para el proyecto actual (ver *Propiedades de Estilos* para mas detalles)

- *Nuevos colores de capa*, un conjunto de colores para usar de forma predeterminada cuando se agregan nuevas capas a QGIS
- o paletas personalizadas que puede crear o importar usando el botón … al lado del cuadro combinado de la paleta.

Por defecto, *Colores recientes*, :guilabel:` Colores estándar` y *Paletas de colores de proyecto* no se pueden eliminar y están configuradas para aparecer en el menú desplegable de botones de color. También se pueden agregar paletas personalizadas a este widget gracias a la opción *Mostrar en botones de colores*.

Para cualquiera de las paletas, puede administrar la lista de colores utilizando el conjunto de herramientas al lado del marco, es decir:

- *Añadir* o *Borrar* color
- *Copiar* o *Pegar* color
- *Importar* o *Exportar* el conjunto de colores de/a archivo `.gpl`.

Haga doble clic en un color de la lista para modificarlo o reemplazarlo en el diálogo *Color Selector*. También puede cambiarle el nombre haciendo doble click en la columna *Etiqueta*.

5.1.12 Configuración de fuentes

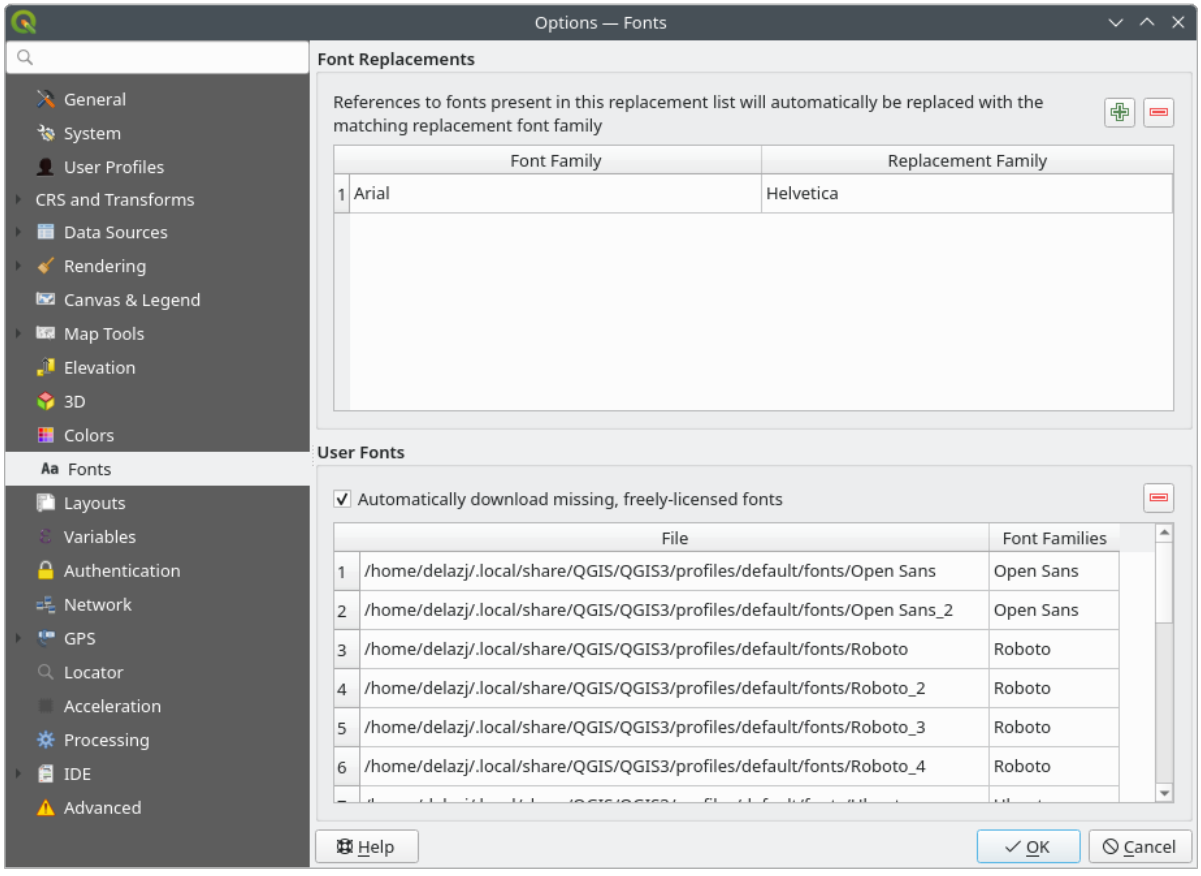


Figura 5.19: Configuración de fuentes

La pestaña *Fonts* permite gestionar las fuentes utilizadas en todos los proyectos:

- *Sustituciones de fuentes*: ofrece la posibilidad de rellenar una lista de sustituciones automáticas de fuentes para aplicarlas al cargar proyectos o estilos, lo que permite una mejor compatibilidad con proyectos y estilos que se utilicen en distintos sistemas operativos (por ejemplo, sustituir «Arial» por «Helvetica»).
- *Fuentes de usuario*: Permite colocar fuentes TTF u OTF en la subcarpeta `fonts` del *perfil de usuario*. Estas fuentes pueden cargarse automáticamente al iniciar QGIS. Esto proporciona una forma de utilizar las fuentes sin necesidad de instalarlas en el sistema operativo, lo que a menudo está bloqueado en entornos empresariales. El panel enumera todas las fuentes de usuario instaladas y permite gestionar (es decir, eliminar) las fuentes de usuario previamente instaladas.

  También es posible  *Descargar automáticamente fuentes sin licencia*: Por ejemplo, si abre un proyecto o estilo, o intenta cargar una capa de mosaico vectorial que hace referencia a fuentes que no están disponibles actualmente, se consulta una lista codificada de fuentes con licencia libre para descargar mediante URL para determinar si es posible descargar automáticamente la fuente al directorio de fuentes del perfil de usuario (con notificación de la licencia de la fuente).

5.1.13 Ajustes de Composiciones

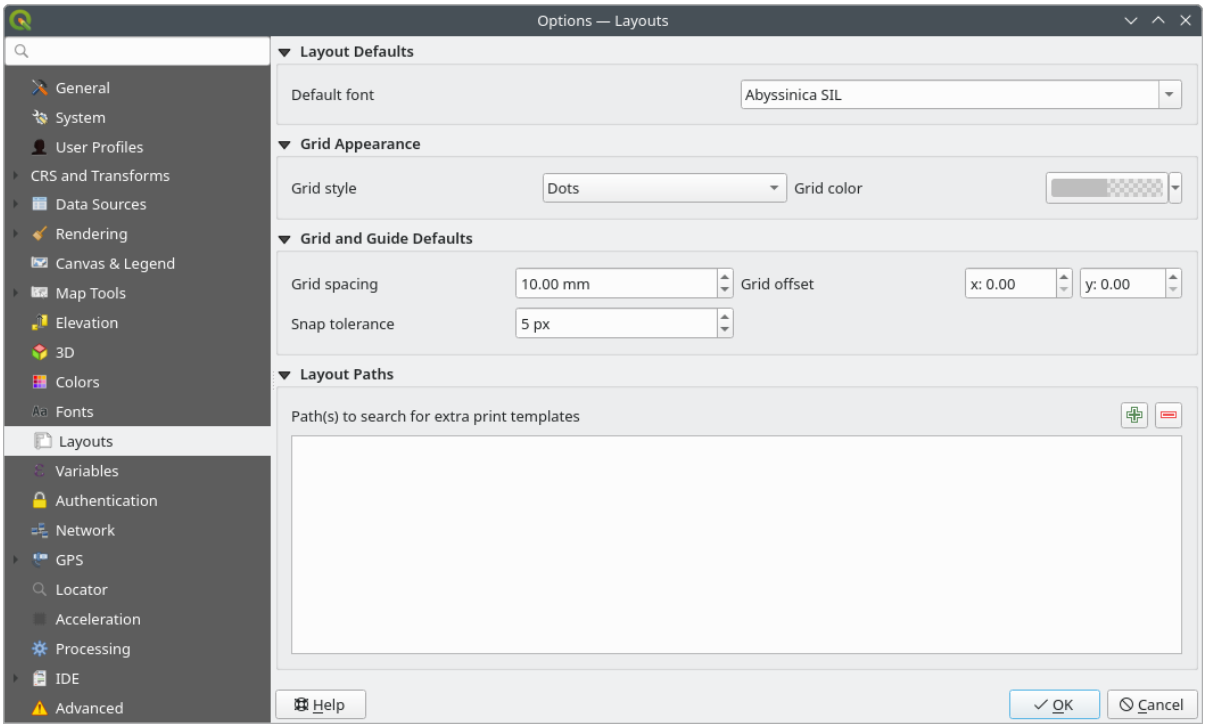


Figura 5.20: Ajustes de Composiciones

Predeterminados de la composición

Puede definir la *Fuente `predeterminada* usada dentro del *print layout*.

Apariencia de la cuadrícula

- Definir el estilo *Grid* (“Sólido”, “Puntos”, “Cruces”)
- Definir el *Color de la cuadrícula*

Valores predeterminados de la cuadrícula y guía

- Definir el *Espaciado entre cuadrículas*
- Definir el *Desplazamiento de cuadrícula* para X e Y
- Defina la *Tolerancia de autoensamblado*.

Rutas de diseño

- Define las *Ruta(s) para buscar plantillas extra de impresión*: una lista de carpetas con plantillas de diseño personalizadas a usar al crear uno nuevo.

5.1.14 Configuración de variables

La pestaña *Variables* lista atodas las variables disponibles a nivel global.

También permite al usuario gestionar variables de nivel global. Haga clic en el botón para agregar una nueva variable personalizada de nivel global. Del mismo modo, seleccione una variable personalizada de nivel global de la lista y haga clic en el botón para eliminarla.

Mas información sobre variables en la sección *Almacenando valores en variables*

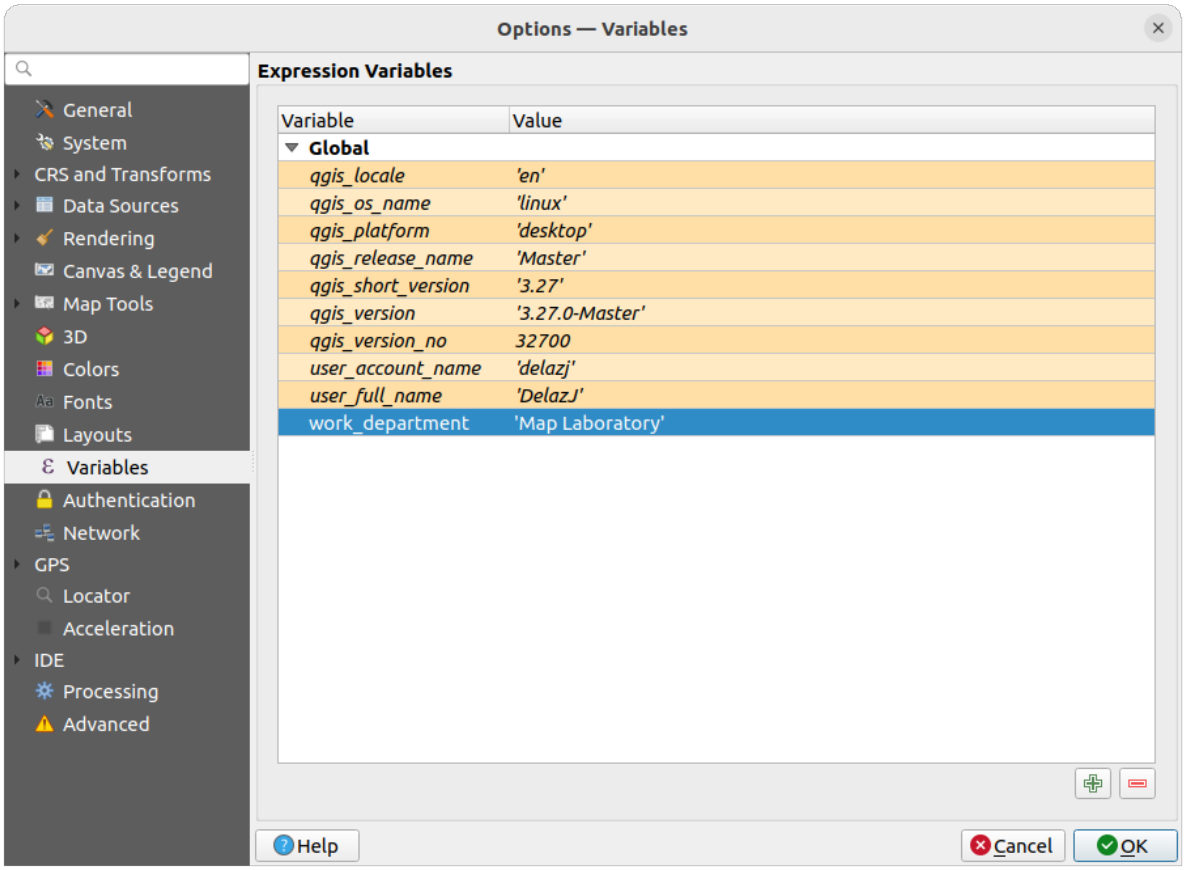


Figura 5.21: Configuración de variables

5.1.15 Configuración de la autenticación

En la pestaña *Authentication* puede establecer las configuraciones de autenticación y administrar certificados PKI. Ver *Sistema de autenticación* para mas detalles.

Para gestionar las autenticaciones, puede utilizar la lista de herramientas situada junto al marco, es decir

- Añadir nueva configuración de autenticación.
- Eliminar la configuración de autenticación seleccionada.

- Editar la configuración de autenticación seleccionada.

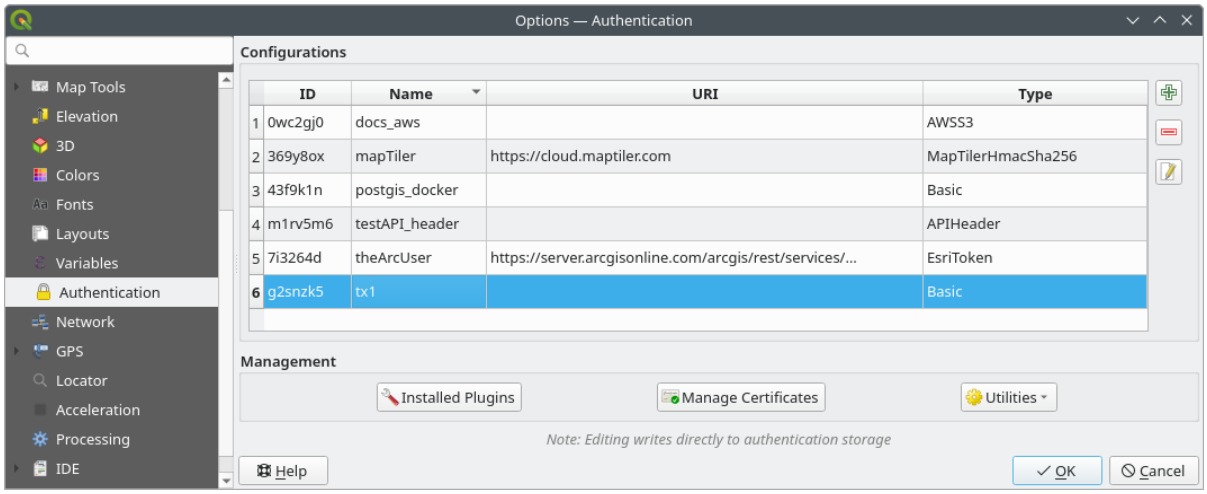


Figura 5.22: Configuración de la autenticación

5.1.16 Configuración de la red

General

- Definir *Expiró el tiempo para solicitudes de red* - por omisión 60000
- Define el *Periodo de expiración por defecto para WMS Capabilities (horas)* - por defecto es 24
- Definir *Periodo de expiración predeterminado para teselas WMS-C/WMTS (horas)* - predeterminado es 24
- Define el *Máximo número de intentos en caso de errores de solicitud de archivo o entidad*
- Define *Prefijo de agente de usuario* que por defecto es `Mozilla/5.0`. Este valor se añadirá tanto a QGIS como a la versión del sistema operativo, por ejemplo, `Mozilla/5.0 QGIS/33801/Ubuntu 22.04.4 LTS` para dar forma al agente de usuario.

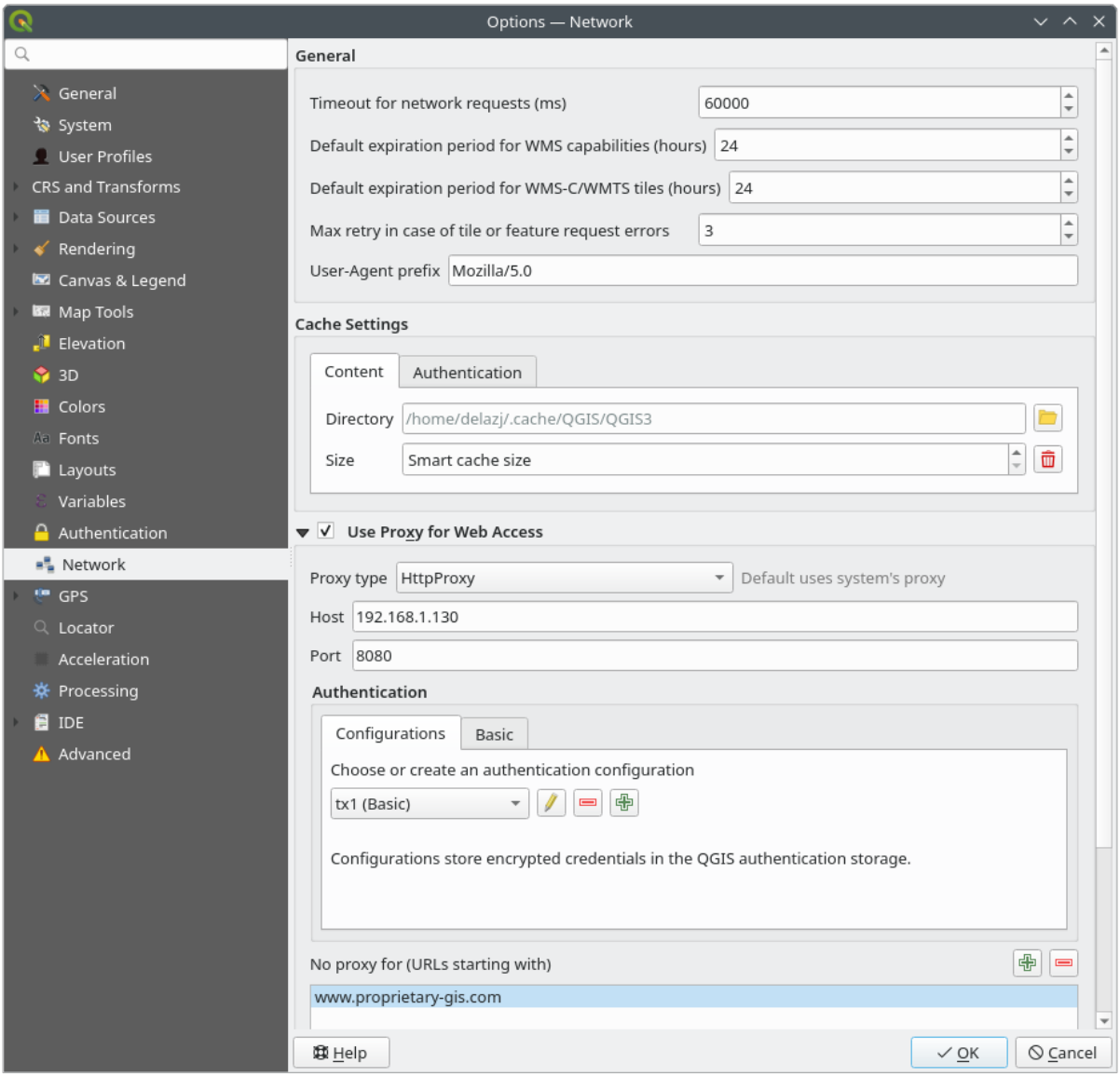


Figura 5.23: Configuración de red y proxy

Configuración de caché

Define el *Directorio* y un *Tamaño* para la caché especificado en megabytes. Puede borrar el valor para activar *Tamaño inteligente de la caché*, que establece el tamaño máximo de la caché basándose en el espacio disponible en disco. También ofrece herramientas para *borrar automáticamente la caché de autenticación de conexión en errores SSL (recomendado)*.

Proxy para acceso web

- *Usar proxy para acceso web*
- Ver el *tipo de Proxy* que se ajuste a sus necesidades y defina “Host” y “Puertoo”. Los tipos de proxy disponibles son:
  - *Proxy Predeterminado*: El proxy se determina en función del proxy del sistema
  - *Socks5Proxy*: Proxy genérico para cualquier tipo de conexión. Admite TCP, UDP, enlace a un puerto (conexiones entrantes) y autenticación.
  - *HttpProxy*: Implementado con el comando «CONNECT», sólo admite conexiones TCP salientes; admite la autenticación.

- *HttpCachingProxy*: Implementando el uso de comandos HTTP normales, es útil sólo en el contexto de peticiones HTTP.
- *FtpCachingProxy*: Implementar el uso de un proxy FTP, es útil sólo en el contexto de las peticiones FTP.

Las credenciales del proxy se configuran utilizando *authentication widget*.

La exclusión de algunas URL se puede agregar al cuadro de texto debajo de la configuración del proxy (ver Figura 5.23). No se utilizará ningún proxy si la URL de destino comienza con una de las cadenas enumeradas en este cuadro de texto.

Si necesita más información sobre los diferentes ajustes del proxy, consulte el manual de la documentación de la librería QT subyacente en https://doc.qt.io/archives/qt-5.9/qnetworkproxy.html#ProxyType-enum

Truco

Utilizar proxies

Usar proxies a veces puede ser complicado. Es útil proceder mediante "prueba y error" con tipos de proxy anteriores, para verificar si tienen éxito en su caso.

5.1.17 Ajustes del GPS

El diálogo *GPS* le ayuda a configurar las conexiones y propiedades de los dispositivos GPS en QGIS. También proporciona ajustes para el seguimiento GPS y la digitalización de datos.

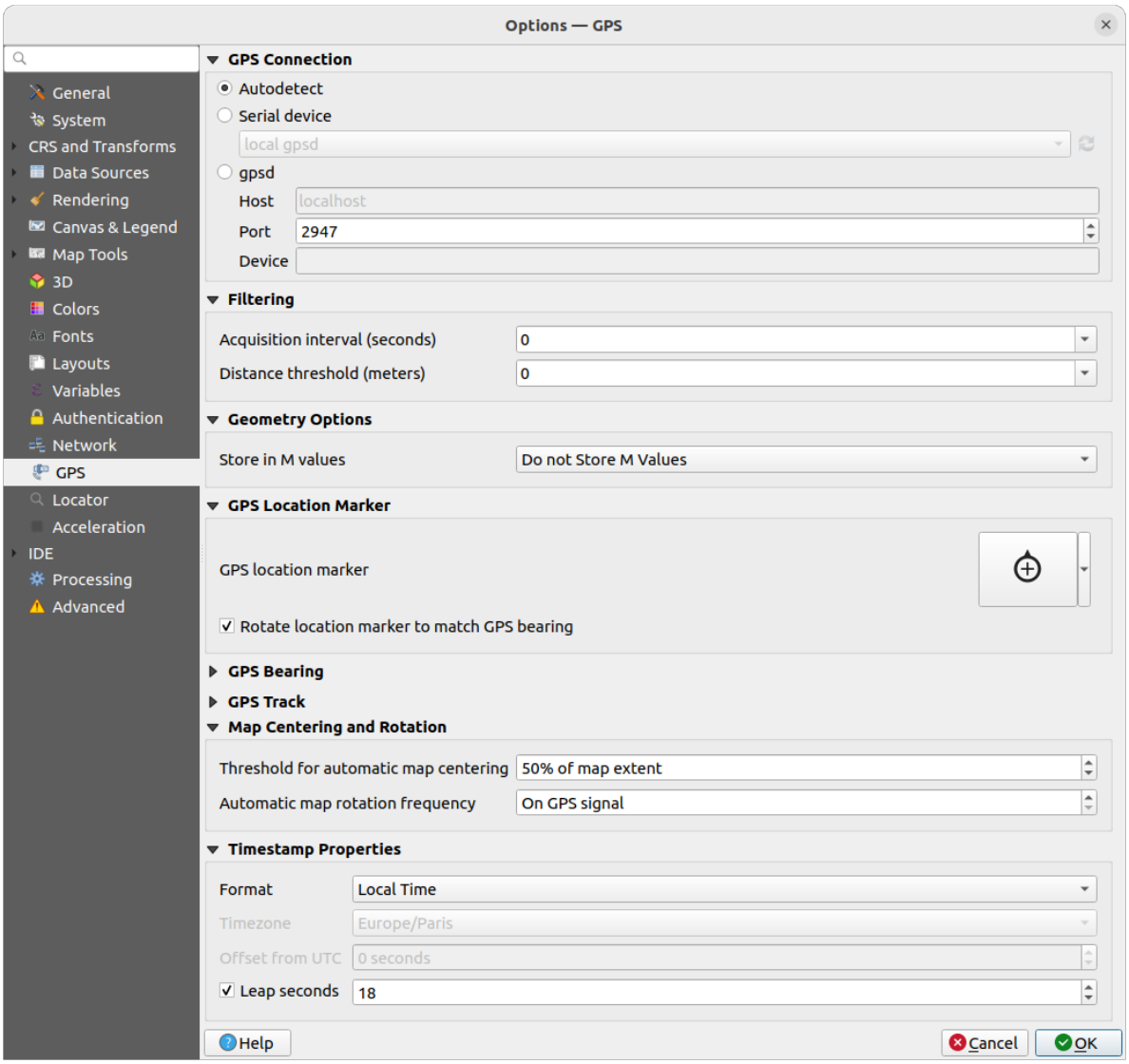


Figura 5.24: Ajustes del GPS

Puede especificar:

- *Conexión GPS*: proporciona diferentes medios para conectar el dispositivo a QGIS
  - *Autodetectar*
  - *Dispositivo serial* (es necesario recargar si se conecta un nuevo dispositivo GPS)
  - *gpsd* (seleccionando el *Host* y *Puerto* al que está conectado su GPS, y proporcionando un *Nombre del dispositivo*)

  En caso de problemas de conexión, puedes intentar cambiar de uno a otro.
- *Filtrado*: Puede establecer un parámetro *Intervalo de adquisición (segundos)* y/o un parámetro *Umbral de distancia (metros)* para mantener el cursor aún activo cuando el receptor se encuentre en condiciones estáticas.
- En *Opciones de geometría*, la opción *Almacenar en valores M* permite crear geometrías con valores M a partir de las herramientas de registro GPS incorporadas. Esto se aplica tanto a los objetos espaciales digitalizados a partir de los registros GPS como a los obtenidos mediante funcionalidad *Acceder a Geopackage/SpatiaLite* … Las opciones incluyen el almacenamiento de marcas de tiempo (como ms desde la época), velocidad sobre el terreno, altitudes, rumbos y componentes de precisión como valores m.

- *Marcador de Localización GPS*
  - Configurar un símbolo para el *Marcador de posición GPS*, indicando la posición GPS actual
  - *Rotar el marcador de posición para que coincida con el rumbo GPS*: si el símbolo del marcador debe girarse para que coincida con la dirección GPS.
- *Rumbo GPS*:
  - configurar una *Estilo de línea portante* utilizando las propiedades de los símbolos de línea de QGIS
  - establece si *Calcular Rumbos a partir de la dirección de viaje*: Si está marcada, se ignorará el rumbo informado por el dispositivo GPS y en su lugar se calculará el rumbo mediante el ángulo entre las dos ubicaciones GPS anteriores.
- *Traza GPS*: establece el símbolo a utilizar para el *Estilo de línea de pista*.
- *Centrado y rotación del mapa*: define cuándo se actualiza el lienzo del mapa según el desplazamiento del usuario sobre el campo:
  - *Umbral para el centrado automático del mapa*: define el desplazamiento mínimo de la posición GPS desde el centro del lienzo del mapa para activar un *recentrado de mapa* automático, si está habilitado.
  - *Frecuencia de rotación automática del mapa*: define la frecuencia con la que puede producirse la rotación del mapa para que coincida con el rumbo GPS; puede tener una duración personalizada o *Con señal GPS*.
- *Propiedades de la marca de tiempo* para configurar cómo se muestran y almacenan los valores de tiempo en los datos. Los parámetros incluyen *Formato* que puede ser:
  - *Hora local*
  - *UTC*
  - *UTC con desfase*, para tener en cuenta los desfases por horario de verano u otros problemas complejos de huso horario
  - una *Zona horaria* específica

  Además, se puede aplicar la corrección *Segundos de salto*, añadiendo los segundos a la marca de tiempo del GPS.

GPSBabel

GPSBabel convierte waypoints, tracks y rutas entre receptores GPS populares como Garmin o Magellan y programas de mapas como Google Earth o Basecamp. Se admiten literalmente cientos de receptores y programas GPS. QGIS se basa en GPSBabel para interactuar con estos dispositivos y *manipular sus datos*.

Para más información sobre cómo hacerlo, consulte *Carga hacia o desde un dispositivo*.

5.1.18 Configuración del localizador

La pestaña *Localizador* le permite configurar la *barra de Localizador*, un control de búsqueda rápida disponible en la barra de estado para ayudarle a llevar a cabo búsquedas en su aplicación. Proporciona algunos filtros predeterminados (con prefijo) a usar:

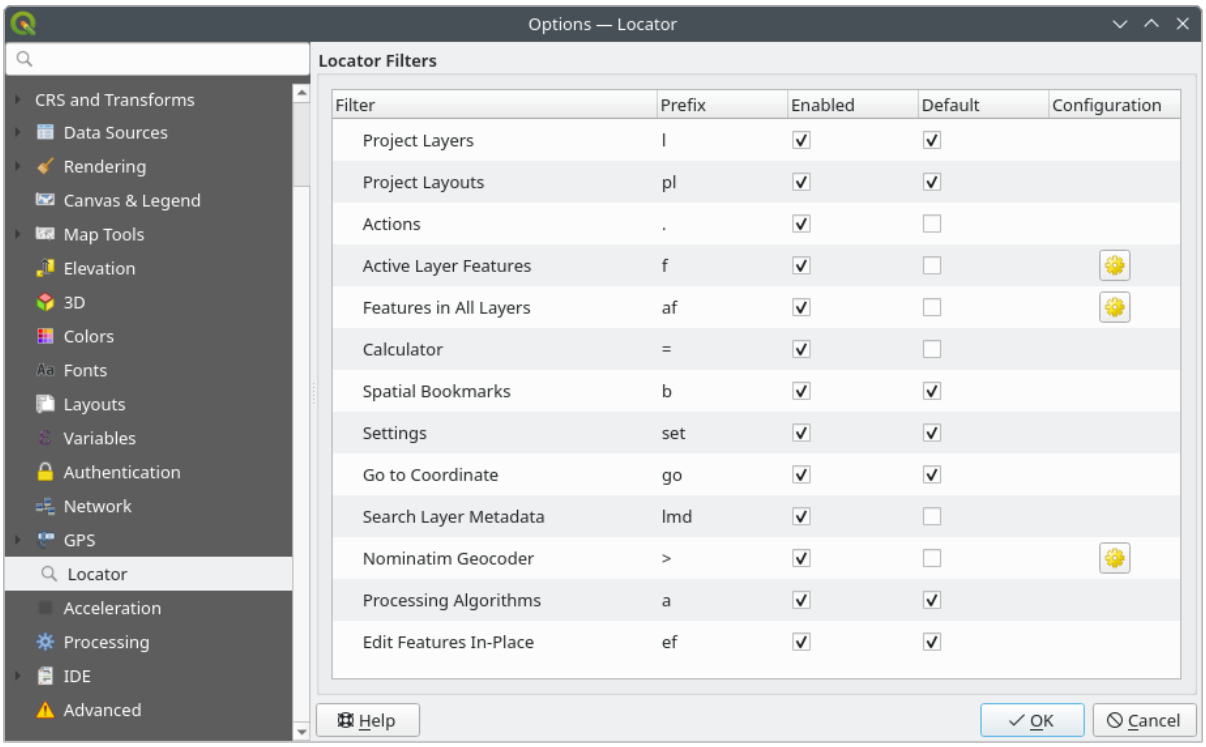


Figura 5.25: Configuración del localizador

- *Capas del proyecto* (`l`): busca y selecciona una capa en el panel *Capas*.
- *Diseños de proyecto* (`pl`): busca y abre un diseño de impresión.
- Acciones (`.`): encuentra y ejecuta una acción de QGIS; las acciones pueden ser cualquier herramienta o menú en QGIS, abriendo un panel…
- *Objetos Espaciales de la capa activa* (`f`): busca atributos coincidentes en cualquier campo de la capa activa actual y se acerca al objeto espacial seleccionado. Pulse **|configuración|** para configurar el número máximo de resultados.
- *Objetos espaciales en todas las capas* (`af`): busca atributos coincidentes en el *nombre de la pantalla* de cada *capas buscables* y se acerca al objeto espacial seleccionado. Pulse **|configuración|** para configurar el número máximo de resultados y el número máximo de resultados por capa.
- *Calculadora* (=): permite evaluar cualquier expresión de QGIS y, si es válida, da la opción de copiar el resultado al portapapeles.
- *Marcadores espaciales* (`b`): busca y amplía la extensión del marcador.
- *Ajustes* (`set`): explora y abre los diálogos de propiedades del proyecto y de la aplicación.
- *ir a Coordenada* (`go`): desplaza el lienzo del mapa a una ubicación definida por un par de coordenadas x e y separadas por comas o espacios o una URL formateada (por ejemplo, OpenStreetMap, Leaflet, OpenLayer, Google Maps, …). La coordenada se espera en WGS 84 (`epsg:4326`) y/o SRC del lienzo del mapa.
- *Nominatim Geocoder* (>): geocodes using the Nominatim geocoding service of the OpenStreetMap Foundation. Press  to limit results to one or more countries using list of country codes. This helps narrow down the search to specific countries, avoiding ambiguous results from other locations. For example, entering `DE` or `DE,FR` restricts results to Germany or Germany and France, respectively.
- Procesar algoritmos (`a`): busca y abre un cuadro de diálogo Procesar algoritmos.
- *Editar objetos espaciales seleccionados* (`ef`): proporciona un acceso rápido y ejecuta un modify-in-place compatible <processing_inplace_edit>` Algoritmo Procesos en la capa activa.

En el Cuadro de Diálogo, puede:

- personalizar el filtro *Prefijo*, es decir, la palabra clave que se utilizará para activar el filtro
- establecer si el filtro es *Habilitado*: el filtro se puede utilizar en las búsquedas y hay un acceso directo disponible en el menú de la barra de localización
- Establecer si el filtro es *Predeterminado*: una búsqueda sin usar filtros devuelve los resultados solo de las categorias de los filtros predeterminados
- Algunos filtros proporcionan un modo de configurar el número de resultados en una búsqueda.

EL conjunto de filtros de localización predeterminados puede ampliarse mediante complementos, p.ej. para búsqudas OSM por nombre, búsqueda directa en base de datos, busquedas en catálogo de capas,…

5.1.19 Ajustes de aceleración

Ajustes de aceleración OpenCL

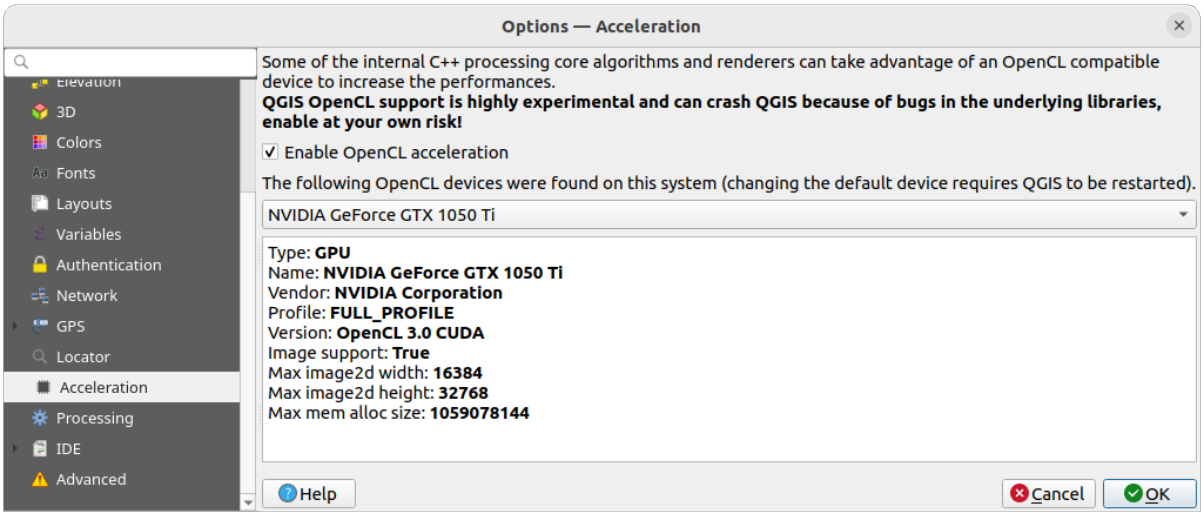


Figura 5.26: Ajustes de aceleración

Dependiendo de su hardware y software, es posible que deba instalar bibliotecas adicionales para habilitar la aceleración OpenCL.

5.1.20 Configuración IDE

En *Token de acceso a GitHub*, puedes generar un token personal que te permita compartir fragmentos de código desde el editor de código Python. Más detalles sobre la autenticación GitHub

Configuración del editor de código

En la pestaña *Editor de código*, puede controlar la apariencia y comportamiento de los controles del editor de código (consola interactiva de Python, control de expresión y editor de función, …).

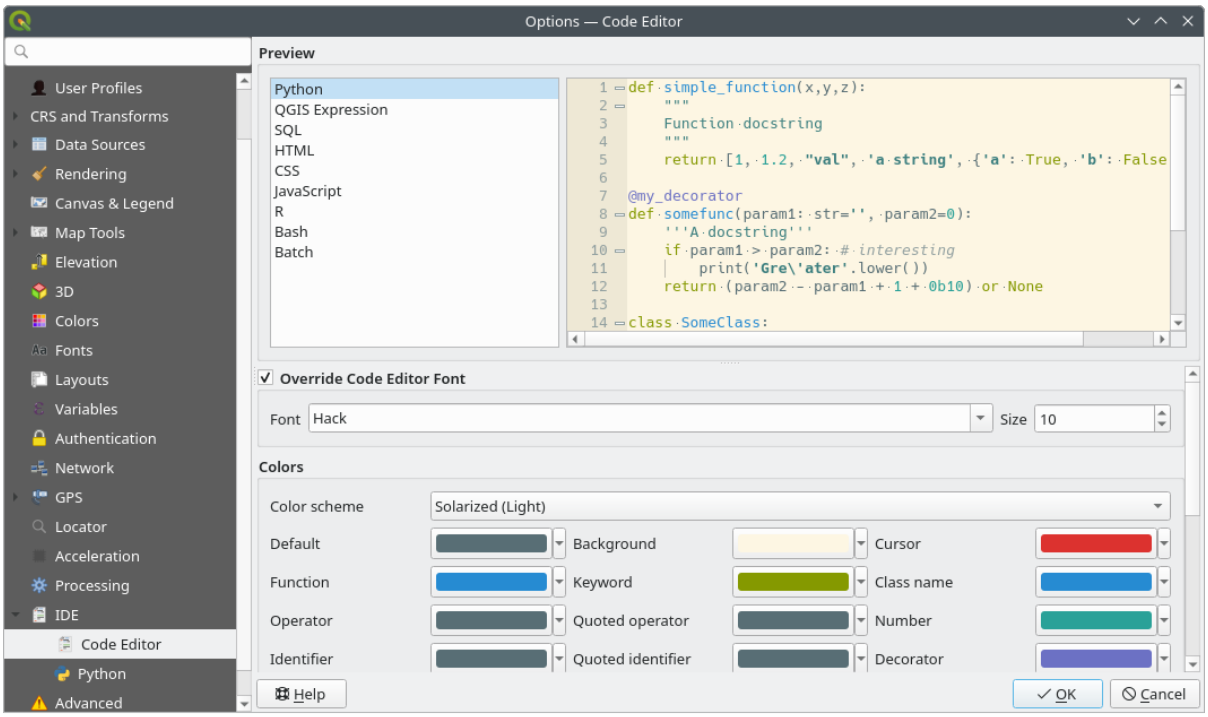


Figura 5.27: Configuración del editor de código

En la parte superior del Cuadro de Diálogo, un control proporciona una previsualización en vivo de la configuración actual, en varios lenguajes de código (Python, expresión QGIS, HTML, SQL, JavaScript). Un modo conveniente de ajustar la configuración.

- Marque *Anular fuente del editor de código* para modificar la familia *Fuente* y *Tamaño* predeterminado.
- En el grupo *Colores* , puede:
  - seleccionar un *Esquema de Color*: las configuraciones predeterminadas son `Default`, `Solarized Dark` y `Solarized Light`. Un esquema `Custom` se desencadena tan pronto como modifica un color y puede ser reseteado mediante la selección de un esquema predeterminado.
  - cambie el *color* de cada elemento en la escritura de código, así como los colores a usar para comentarios, citas, funciones, fondo, …

Configuración de Python

La configuración *Python* le ayuda a gestionar y controlar el comportamiento de los editores Python (*consola interactiva*, *editor de código*, *macros de proyecto*, *expresiones personalizadas*, …). También se puede acceder a él utilizando el botón Opciones… desde:

- la barra de herramientas de *consola de Python*
- el menú contextual del control *consola de Python*
- y desde el menú contextual del editor de código.

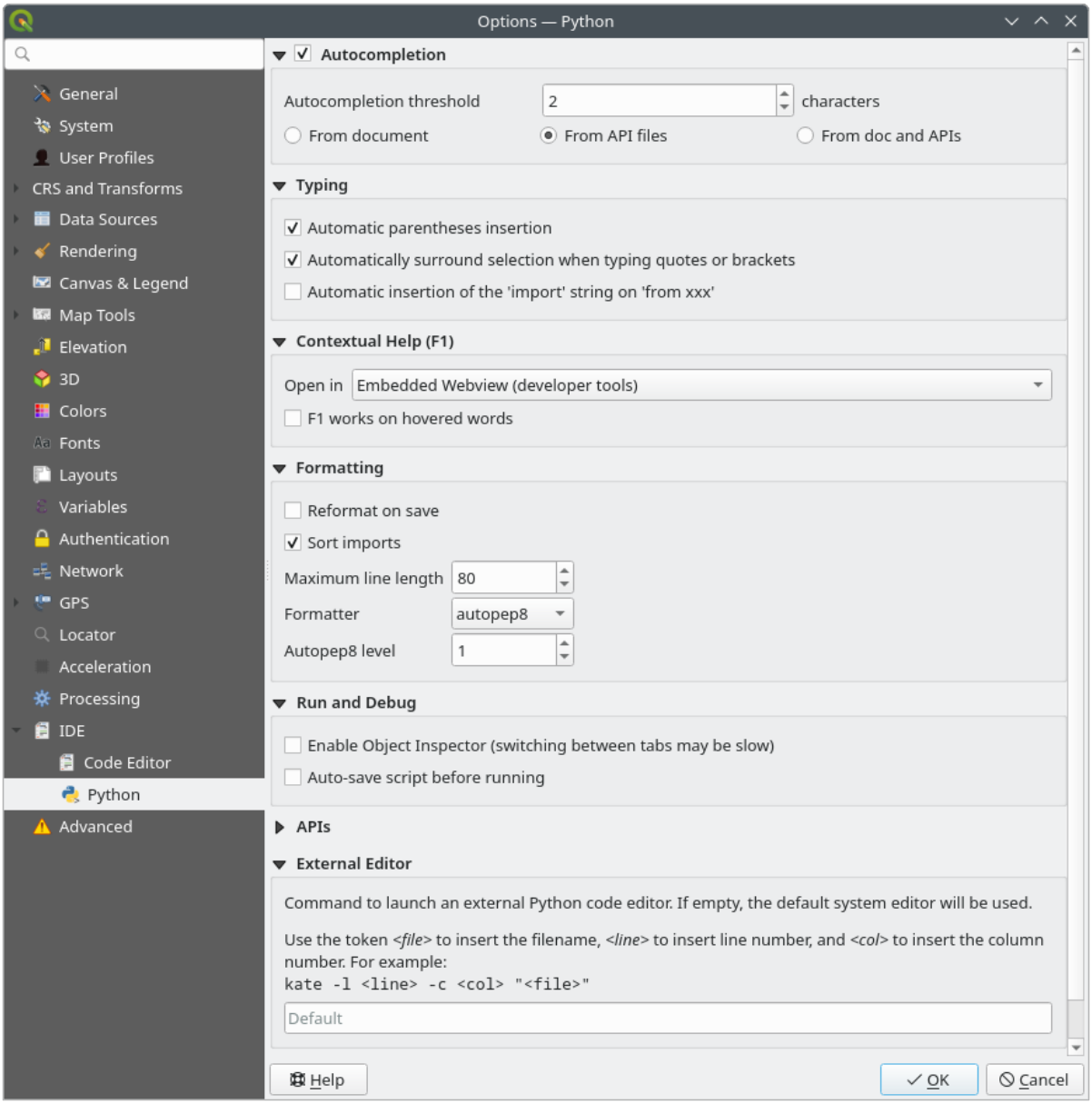


Figura 5.28: Configuración de Python

Puede especificar:

- *Autocompletion*: Enables code completion. You can get autocompletion from the current document, the *installed API* files or both.
  - *Autocompletion threshold*: Sets the threshold for displaying the autocompletion list (in number of characters)
- en *Escribiendo*
  - *Inserción automática de paréntesis*: Cuando no hay texto seleccionado, si se introduce un carácter de apertura (paréntesis, comillas, corchetes, …), inserta el carácter de cierre correspondiente justo después del cursor. Tenga en cuenta que este comportamiento se desactiva si el cursor actual está dentro de una cadena o comentario.
  - *Rodea automáticamente la selección al escribir comillas o corchetes*: Cuando se introduce un carácter de apertura, el texto seleccionado se encierra en el par de apertura/cierre. La selección permanece igual,

por lo que es posible entrecomillar una palabra seleccionada y encerrarla entre paréntesis simplemente escribiendo `"` y luego `(`.

Caso especial para la selección multilínea con comillas y comillas dobles: la selección se encierra entre comillas simples/dobles triples.

- ☐ *Inserción automática de la cadena "import" en "from xxx"*: Habilita la inserción de "import" al especificar importaciones

- *Contextual Help (F1)* provides options to control QGIS behavior when requesting help on a PyQGIS or Qt object in a Python code sample. When pressing `F1`, QGIS will open the API page (or a search results page) matching:

  1. the selected text if any OR
  2. the word at the mouse position (if ☑ *F1 works on hovered words* is enabled) OR
  3. the word at the text cursor position

  This works for both Qt and PyQGIS API documentation. The page is opened either in QGIS, as an *Embedded webview (developer tools)*, or in the *Default system web browser*.

- en *Formateo*, puede añadir herramientas automatizadas para reformatear el código que está escribiendo:
  - ☐ *Reformatear al guardar*: el formato se aplica justo antes de guardar el script
  - ☑ *Ordenar importaciones*: ordena las sentencias *import* utilizando la librería isort
  - *Longitud máxima de línea*: controla cómo el formateador envolverá las líneas, y controla la regla del editor.
  - *Formateador* - las herramientas soportadas son *autopep8* y *negro*, con opción dedicada:
    - *Nivel autopep8* - más detalles en autopep8 advanced usage
    - ☐ *Normalizar comillas*: sustituye todas las comillas simples por comillas dobles si es posible

- en *Ejecutar y Depurar*
  - ☐ *Habilitar Inspector de Objetos (el cambio entre pestañas puede ser lento)*
  - ☐ *Auto-guardar script antes de ejecutar*: Guarda el script automáticamente antes de ejecutar. Esta acción almacenará un archivo temporal (en el directorio temporal del sistema) que se eliminará automáticamente después de ejecutarse.

Under *APIs*, you can specify which APIs your code should rely on:

- ☑ *Using preloaded APIs file*: You can choose if you would like to use the preloaded API files. By default, supported APIs are:
  - Python
  - PyQGIS
  - PyQt
  - QScintilla2
  - osgeo-gdal-ogr
  - osgeo-geos

  You can also choose to add specific API files.

- ☐ *Usar archivo de APIs preparado*: Si está marcado, el archivo `*.pap` escogido será usado para completado de código. Para generar un archivo preparado de API tiene que cargar al menos un archivo `*.api` y luego compilarlo clickando en el botón *Compilar APIs…*.

El grupo *Editor externo* le permite proporcionar instrucciones de línea de comandos para iniciar un editor de código Python externo, dado un nombre de archivo, un número de línea y un número de columna. Si está vacío, se utilizará el editor predeterminado del sistema.

5.1.21 Ajustes de Procesos

La pestaña *Procesos* le proporciona la configuración general de las herramientas y proveedores de datos que se utilizan en el marco de Procesamiento de QGIS. Más información en *Configurando el marco de procesamiento*.

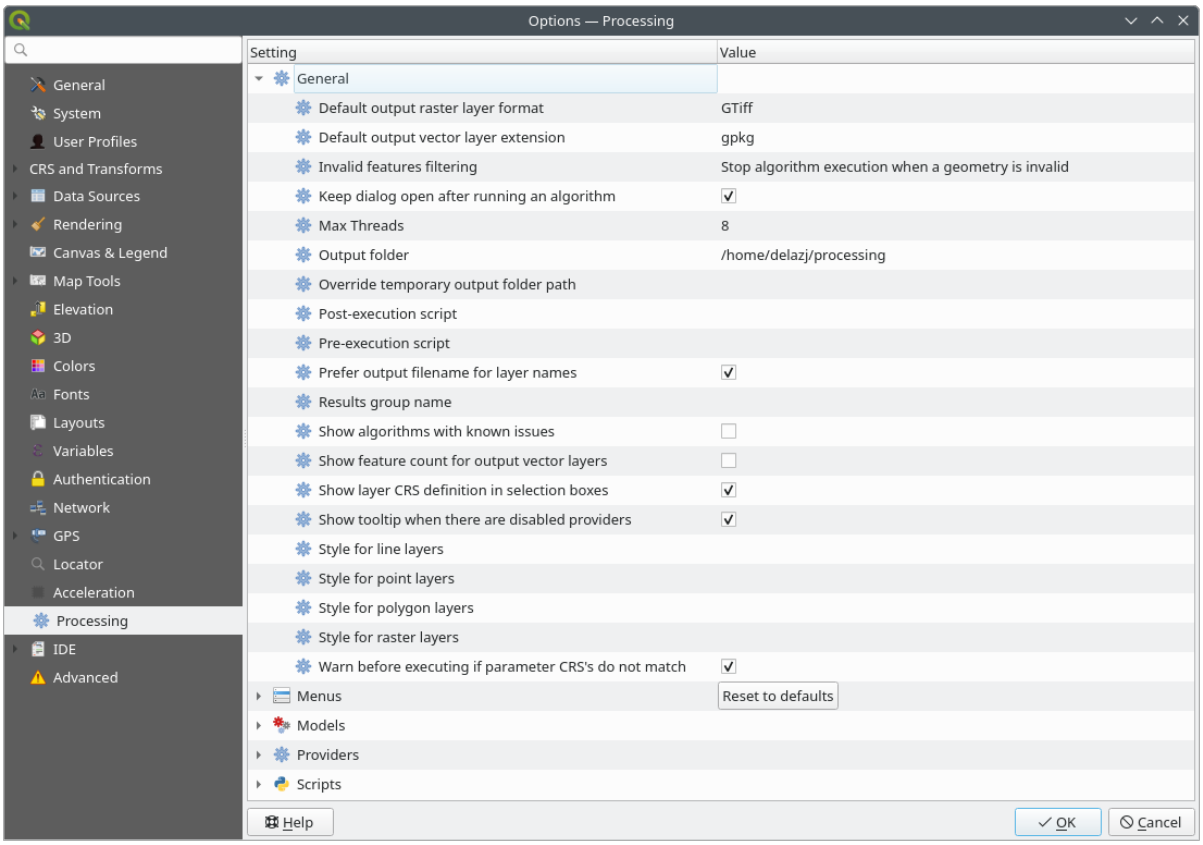


Figura 5.29: Ajustes de Procesos

5.1.22 Ajustes avanzados

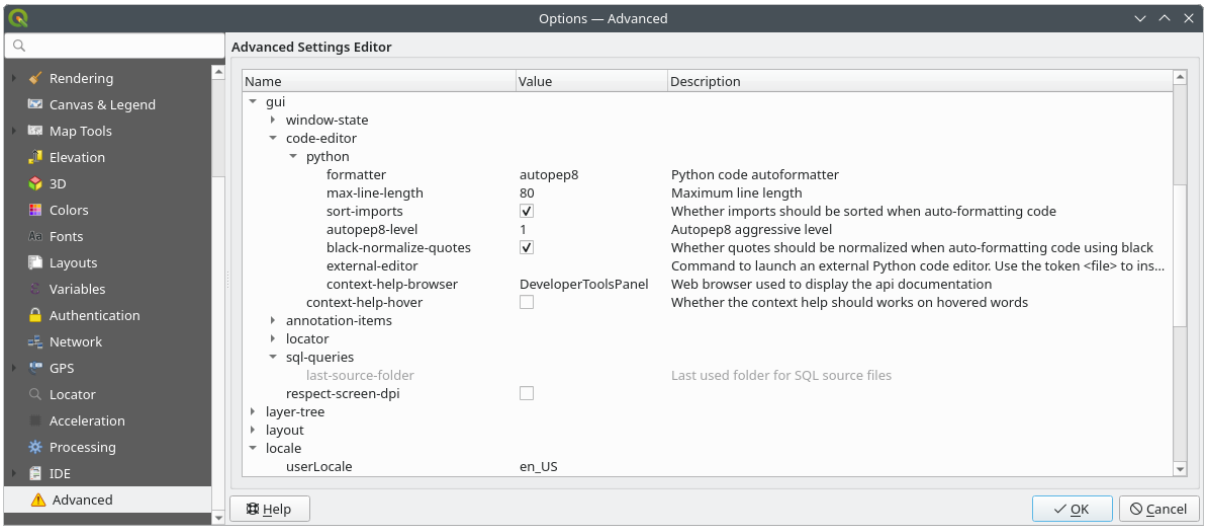


Figura 5.30: Ajustes avanzados

Todos los ajustes relacionados con QGIS (UI, herramientas, proveedores de datos, configuraciones de Procesamiento, valores por defecto y rutas, opciones de complementos, expresiones, verificaciones de geometría…) son guardadas en un archivo `QGIS/QGIS3.ini` en el directorio del *user profile* activo. Las configuraciones se pueden compartir copiando este archivo a otras instalaciones.

Desde QGIS, la pestaña *Avanzado* ofrece una manera de administrar estas configuraciones a través de *Editor de configuraciones avanzadas*. Después de que prometa tener cuidado, el widget se rellena con un árbol de todas las configuraciones existentes, y puede editar su valor. Haga click derecho sobre una configuración o grupo y puede eliminarla (para agregar una configuración o grupo, debe editar el archivo `QGIS3.ini`). Los cambios se guardan automáticamente en el archivo `QGIS3.ini`.

Advertencia

Evite usar la pestaña de Ajustes Avanzados a ciegas

Tenga cuidado al modificar elementos en este diálogo dado que los cambios se aplican automáticamente. Haciendo cambios sin conocimiento puede dañar su instalación de QGIS de varios maneras.

5.2 Trabajando con perfiles de usuario

5.2.1 El concepto

El menú *Ajustes* ► *Perfiles de usuario* proporciona funciones para establecer y acceder a los perfiles de usuario. Un perfil de usuario es una configuración de aplicación unificada que permite almacenar en una sola carpeta:

- todas los *global settings*, incluyendo local, proyecciones, ajustes de autenticación, paletas de color, atajos…
- Configuraciones GUI y :ref:`customization `
- archivos de cuadrícula y otros archivos auxiliares de proyecto instalados para la transformación de datum
- *plugins* instalados y sus configuraciones
- plantillas de proyectos e historial de proyectos guardados con su previsualización de imágenes

- *processing settings*, logs, scripts, modelos.

Por fedecto, una instalación QGIS contiene un único perfil de usuario `por defecto`. Pero puede crear tantos perfiles de usuario como quiera:

1. Haga clic en la opción *Configuración* ► *Perfiles de usuario* ► *Nuevo perfil…*.
2. Se le pedirá que proporcione un nombre de perfil, creando una carpeta con el mismo nombre en `~/<UserProfiles>/` donde:
   - `~` representa el directorio **HOME**, que en Windows suele ser algo así como `C:\Users\<username>`.
   - y `<UserProfiles>` representa el directorio de perfiles principales, p.ej.:
     - `.local/share/QGIS/QGIS3/profiles/`
     - `%AppData%\Roaming\QGIS\QGIS3\profiles\`
     - `Library/Application Support/QGIS/QGIS3/profiles/`

   La carpeta de perfil de usuario se puede abrir desde QGIS utilizando *Abrir carpeta de perfil activo*.
3. Una nueva instancia de QGIS se inicia, utilizando una configuración limpia. Luego puede establecer sus configuraciones personalizadas.

Dependiendo de su hardware y software, es posible que deba instalar bibliotecas adicionales para habilitar la aceleración OpenCL.

Como cada perfil de usuario contiene configuraciones, complementos e historial aislados, pueden ser excelentes para diferentes flujos de trabajo, demostraciones, usuarios de la misma máquina o configuraciones de prueba, etc. Y puede cambiar de uno a otro seleccionándolos en :menuselection:Menú *Configuración -> Perfiles de usuario*. También puede ejecutar QGIS con un perfil de usuario específico desde *command line*.

Truco

Ejecute QGIS con un nuevo perfil de usuario para verificar la persistencia de errores

Los errores que puede encontrar con las funciones en QGIS pueden estar relacionados con restos en el perfil de usuario actual. Ejecutar QGIS bajo otro perfil de usuario puede ayudarte a resolverlos o a comprobar el origen del problema. El consejo general es lanzar QGIS bajo un nuevo perfil de usuario, por lo tanto una configuración más limpia, y ejecutar los comandos de nuevo.

Si el error le impide crear un nuevo perfil de usuario desde el menú *Configuración* ► *Perfiles de usuario*, puede:

- Renombre en el explorador de archivos, la carpeta del perfil de usuario «roto» en la carpeta `QGIS3/profiles` y reinicie QGIS. Se creará y ejecutará un nuevo perfil de usuario `default`.
- Inicie QGIS desde la línea de comandos, utilizando el nuevo argumento *nombre de perfil*:

```
qgis-ltr --profile newprofilename
```

5.2.2 Configuración del perfil de usuario

Por omisión, QGIS abre una nueva sesión con el perfil de la última sesión cerrada. Esto, entre otros ajustes, se puede personalizar en *Configuración* ► **|opciones|** *Opciones* ► **|usuario|** *Perfiles de usuario*:

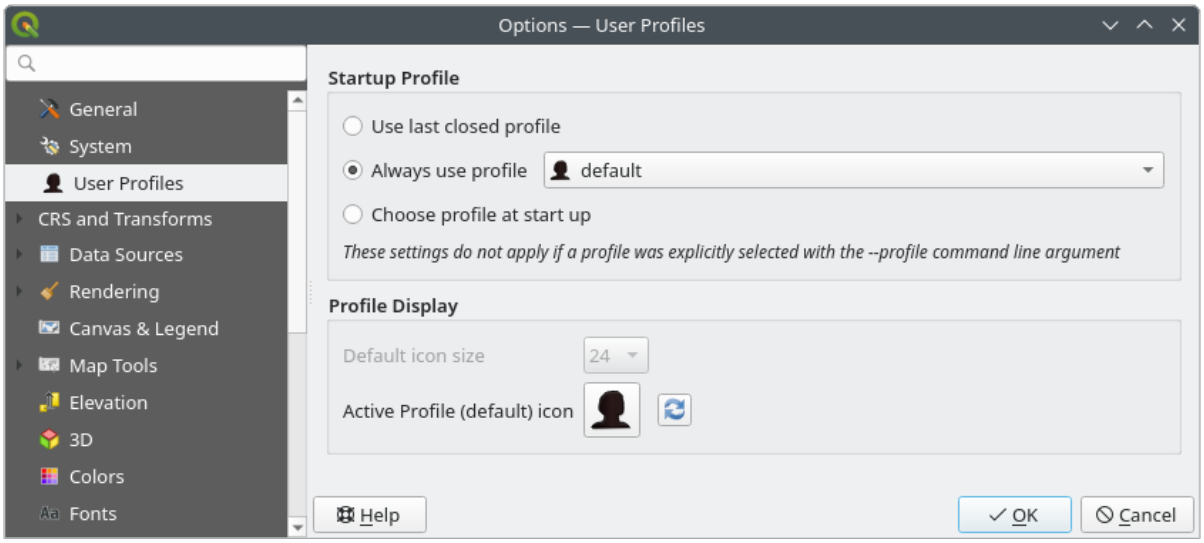


Figura 5.31: Configuración de los perfiles de usuario

- *Perfil de inicio*: indica el perfil de usuario a utilizar al iniciar una sesión QGIS. Puede ser:
  - *Utilizar el último perfil cerrado*.
  - un perfil de usuario específico para seleccionar del menú desplegable *Utilizar siempre perfil*
  - *Elegir perfil al iniciar*: Abre el cuadro de diálogo *Selector de perfil de usuario* con una lista de los perfiles de usuario disponibles. Haga doble clic en una entrada o seleccione un perfil y pulse *OK* para utilizar ese perfil de usuario en la sesión de inicio. También es posible *Añadir nuevo perfil* a la lista: se crea un perfil desde cero y se ejecuta automáticamente con la sesión de apertura.
- En *Visualización del perfil*, puede establecer:
  - El tamaño de los iconos que se utilizarán al seleccionar un perfil en el cuadro de diálogo *Selector de perfiles de usuario*.
  - Un icono específico que se mostrará junto al perfil actual en el menú *Configuración* ► *Perfiles de usuario* o en el cuadro de diálogo *Selector de perfiles de usuario*. Pulse Restablecer icono de perfil para eliminar las modificaciones personalizadas.

5.3 Propiedades del proyecto

En la ventana de propiedades del proyecto en *Proyecto* -> *Propiedades del proyecto*, puede establecer opciones específicas del proyecto. Las opciones específicas del proyecto sobrescriben su equivalente en el diálogo *Opciones* descrito anteriormente.

5.3.1 Propiedades Generales

En la pestaña *General*, la *Configuración general* le permite:

- ver la localización del archivo de proyecto
- establecer la carpeta para el inicio del proyecto (disponible en el elemento *Inicio del proyecto* del panel *Navegador*). La ruta puede ser relativa a la carpeta del archivo del proyecto (introdúzcala) o absoluta. La página de inicio del proyecto puede utilizarse para almacenar datos y otros contenidos útiles para el proyecto. Es conveniente cuando los archivos del conjunto de datos y del proyecto no se almacenan en el mismo lugar. Si no se rellena, la etiqueta *Inicio de proyecto* se establece por defecto en la carpeta del archivo del proyecto.

- dar un título al proyecto además de la ruta del archivo del proyecto
- escoger el color a usar para objetos espaciales cuando son seleccionados
- escoger el color de fondo: el color a usar para la tela de fondo del mapa
- establezca si la ruta a las capas en el proyecto debe guardarse como absoluta (completa) o como relativa a la ubicación del archivo del proyecto. Puede preferir la ruta relativa cuando las capas y los archivos del proyecto se pueden mover o compartir o si se accede al proyecto desde computadoras en diferentes plataformas.
- elegir evitar artefactos cuando el proyecto se represente como mosaicos de mapa. Tenga en cuenta que marcar esta opción puede conducir a una bajada del rendimiento.
- *Recordar ventanas de tablas de atributos y muelles entre sesiones*: Si se marca para un proyecto, cualquier tabla de atributos abierta se guardará en el proyecto y se restaurará inmediatamente al cargar ese proyecto. Esto puede mejorar los flujos de trabajo cuando se construye un proyecto con un conjunto particular de configuraciones de tablas de atributos para sus necesidades, y volver a configurar estas tablas de atributos es una molestia.

Calcular áreas y distancias es una necesidad común en los SIG. Sin embargo, estos valores están realmente vinculados a la configuración de proyección subyacente. El marco *Measurements* le permite controlar estos parámetros. De hecho, puedes elegir:

- el *Ellipsoide*, en el que se basan por completo los cálculos de distancias, áreas y ángulos; puede ser:
  - **None/Planimetric**: los valores devueltos son en este caso mediciones cartesianas. Esta opción puede ser configurada como predeterminada para proyectos nuevos desde el *Settings* ► *Options* ► *CRS Handling* menu
  - uno **Personalizado**: necesitará establecer valores de los ejes semi-mayor y semi-menor.
  - o uno existente de una lista predefinida (Clarke 1866, Clarke 1880 IGN, New International 1967, WGS 84…).
- the *Units for distance measurements* for length and perimeter, and the *Units for area measurements*. These settings default to their corresponding *global options* and can be manually changed for the current project. They are used in:
  - Barra de actualización de campo de tabla de atributos
  - Cálculos de calculadora de campo
  - *Herramienta de identificación* valores derivados de longitud, perímetro y área
  - *diálogo de medida*
  - *decoración de barra de escala*

  Nota

  Para garantizar que las unidades de medida (para distancia y área) siempre coincidan con las unidades del sistema de referencia espacial del proyecto actual, seleccione la opción «Unidades del mapa» en estas opciones.

- the *Scale calculation method*: when working on large map extents, especially with geographic CRS, the scale bar may not reflect actual distance measurement depending on the latitude at which the calculation is done. This setting helps you define the area you want the calculation to be taken from:
  - *Average top, middle and bottom scales*
  - *Calculate along top of map*
  - *Calculate along middle of map* (the default)
  - *Calculate along bottom of map*
  - *Always calculate at equator*, regardless of the actual visible map extent. It can be used to provide a consistent, static scale for maps in geographic reference systems, regardless of the latitudes actually visible

in the map (permitting consistent appearance of these maps when rendering relies on scale based visibility or calculations). Otherwise a project in e.g. `EPSG:4326` which uses scale based visibility of layers and symbols will see layers and features «randomly» disappear as the map is panned, even though you have not zoomed in or out of the map.

> **Nota**
>
> This method is only applicable when calculating scales with a degree based reference system, and while it ensures that the scale remains constant and does not change as the map is panned, it will calculate misleading scales when the map extent is not close to the equator.

Changing the scale calculation method has the following impacts:

- New layout scale bars will default to the project's scale calculation method
- The scale calculations for the status bar widget and map renders will be changed (including flow on impacts like the value of `@map_scale` variable, scale based visibility of layers and symbols)
- This also affects QGIS server map rendering
- Processing algorithms which render maps will respect the project's scale calculation method

> **Nota**
>
> Symbology sizes in map units are NOT affected by this setting.

La *Visualización de coordenadas y rumbos* le permite personalizar la visualización de:

- las coordenadas mostradas en el cuadro *Coordenadas* de la barra de estado de QGIS y en la sección *Derivado* de los resultados de la herramienta  Identificar objetos espaciales
- el valor de rumbo mostrado en la barra de estado para la dirección de desplazamiento del lienzo del mapa y por la herramienta  Medir rumbo.

Los parámetros disponibles son:

- *Visualizar coordenadas usando* o bien:
  - `Unidades de mapa`, basado en el SRC de proyecto
  - `Mapa Geográfico (grados)`: basado en el SRC del proyecto si es de tipo geográfico, en caso contrario utiliza su SRC geográfico asociado. Esto es útil, por ejemplo, para cuerpos celestes no terrestres.
  - o `Unidades de proyección personalizadas`: permite basarse en cualquier SRC que desee para la visualización de coordenadas.
- En la opción *SRC de coordenadas*, puede ver o definir el SRC a utilizar en función de su modo de visualización.
- *Formato de coordenadas*: puede configurarlo como `Grados decimales`, `Grados, minutos` o `Grados, minutos, segundos`, y si debe mostrarse:
  - ☐ *Mostrar sufijo direccional*
  - ☐ *Mostrar ceros a la izquierda en minutos y segundos.*
  - ☐ *Mostrar ceros a la izquierda en los grados.*
  - ☐ *Mostrar ceros finales*
- *Precisión de las coordenadas*: el número de decimales puede ser automático (derivado del tipo de SRC) o establecerse manualmente.

- *Orden de las coordenadas*: puede optar por mostrar las coordenadas en el orden nativo del SRC (`Predeterminado`) o cambiarlo a `Este, Norte (Longitud, Latitud)` o `Norte, Este (Latitud, Longitud)`.
- *Formato de orientación* los valores posibles son `0 a 180°, con sufijo E/W`, `-180 a +180°` o `0 a 360°`. Se puede definir el número de *Lugares decimales* y si se desea *Mostrar ceros al final*`.

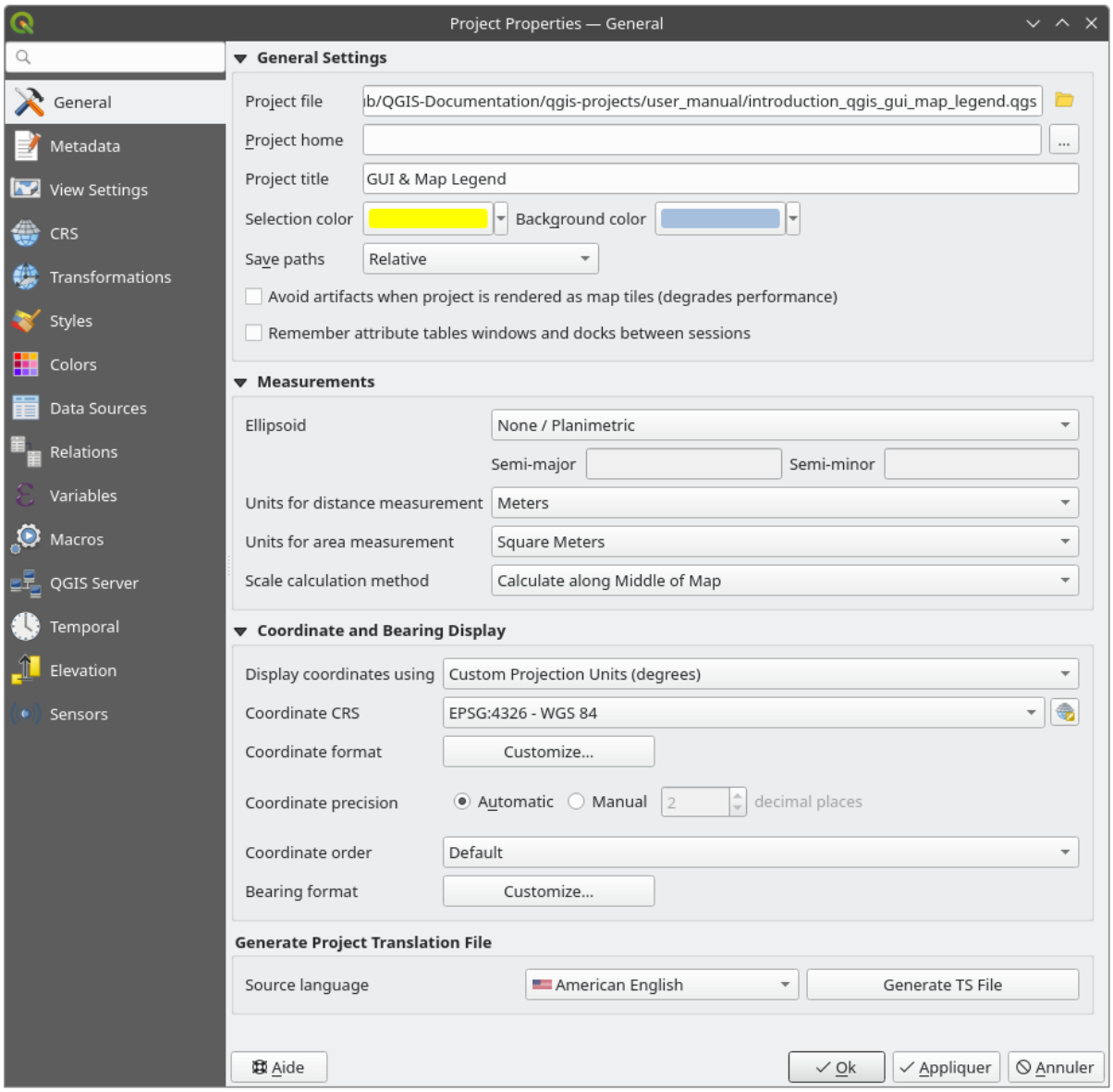


Figura 5.32: Pestaña general del diálogo de Propiedades del Proyecto

5.3.2 Propiedades de metadatos

The *Metadata* tab provides you with options to create and edit a metadata report on your project. See *Metadatos* for more information.

5.3.3 Ver configuraciones

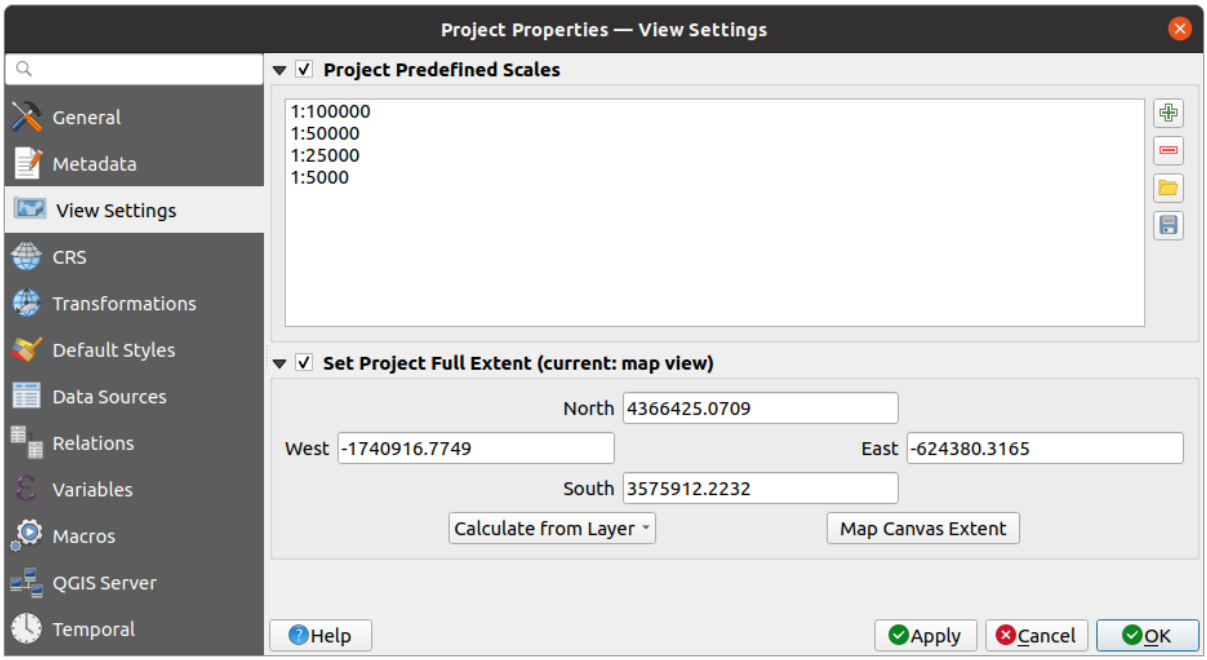


Figura 5.33: Ficha Ver configuración del cuadro de diálogo Propiedades del proyecto

La pestaña *View Settings* proporciona medios para controlar el lienzo del mapa del proyecto. Puede:

- set *Escalas predefinidas del proyecto*: la lista de escalas que se mostrarán en los widgets desplegables relacionados con la escala, como la barra de estado *Escala*, el selector de escalas de visibilidad o la configuración de la vista secundaria del mapa 2D,… en sustitución de la global *escalas predefinidas*.
- *Set Project full Extent*: esta extensión se utilizará en lugar de la extensión de todas las capas al hacer zoom en la extensión completa del mapa ( ). Es útil cuando un proyecto contiene capas web/capas nacionales/capas globales pero el área real de interés del proyecto es un área geográfica más pequeña. Las coordenadas de extensión completa del proyecto se pueden establecer con el widget *extent selector*.

5.3.4 Propiedades CRS

> **Nota**
>
> Para obtener más información sobre cómo QGIS maneja la proyección del proyecto, lea la sección dedicada en *Trabajar con Proyecciones*.

La pestaña *CRS* le ayuda a establecer el sistema de referencia de coordenadas para usar en este proyecto. Puede ser:

- *No CRS (o proyección desconocida/no-Terrestre)*: las capas son dibujadas basándose en sus coordenadas puras

- o un sistema de referencia de coordenadas existente que puede ser *geográfico*, *proyectado* o *definido por el usuario*. Las capas agregadas al proyecto se traducen sobre la marcha a este CRS para superponerlas independientemente de su CRS original.

5.3.5 Propiedades de Transformaciones

La pestaña *Transformaciones* le ayuda a controlar la configuración para la reproyección de capas mediante la configuración de las preferencias de transformación de datum a aplicar en el proyecto actual. Como es habitual, esto sobreescribe cualquier configuración global correspondiente. Ver *Transformaciones de Datum* para mas detalles.

5.3.6 Propiedades de Estilos

En la pestaña *Estilos*, puede configurar los símbolos y colores inherentes al proyecto, permitiendo compartir el proyecto de forma segura entre diferentes máquinas.

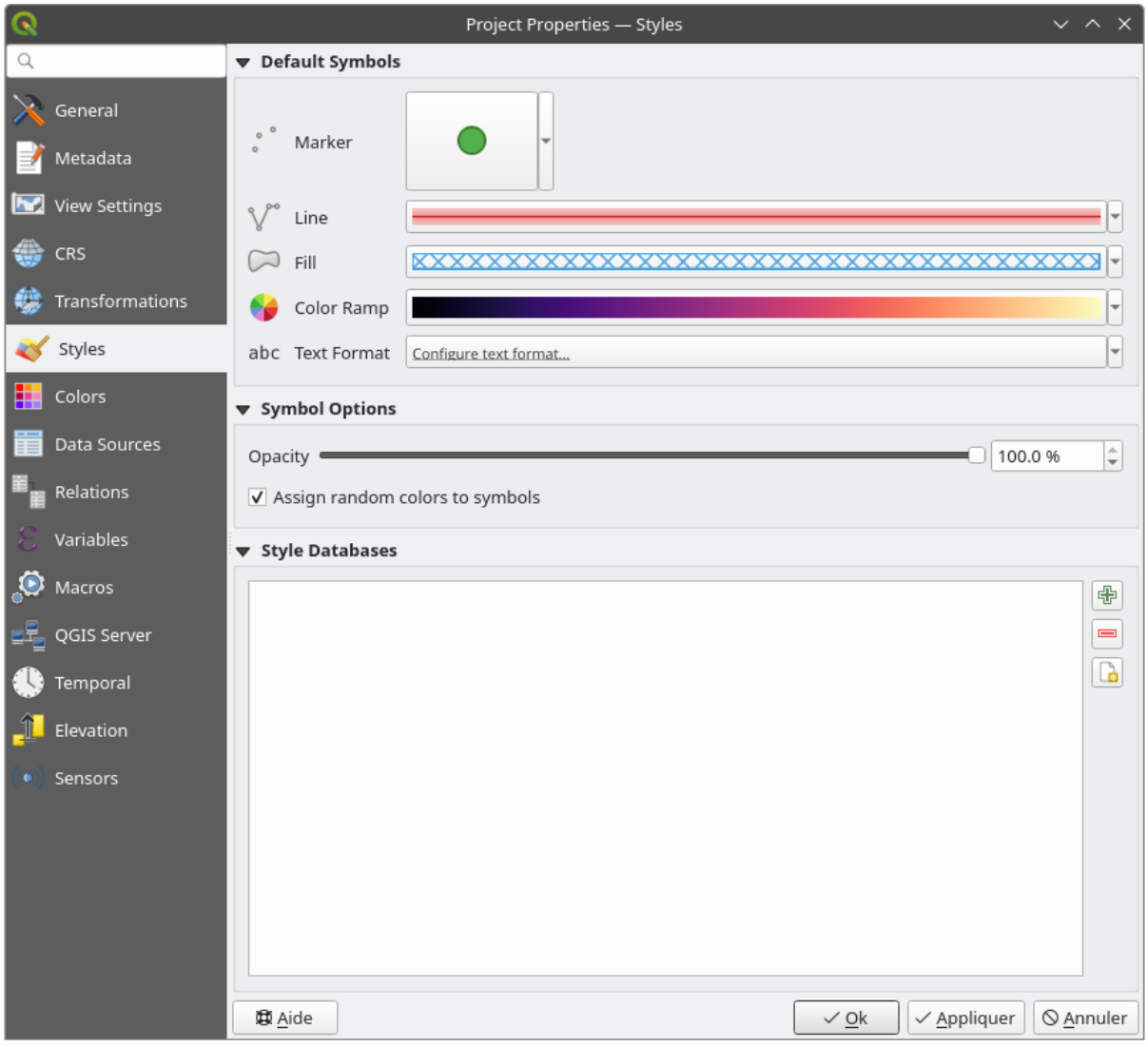


Figura 5.34: Pestaña Estilos

Símbolos Predeterminados

El grupo *Símbolos predeterminados* permite controlar cómo se dibujarán las nuevas capas en el proyecto cuando no tengan definido un estilo `.qml` existente. Puede configurar *Marcador*, *Línea*, *Relleno* para que se apliquen en función del tipo de geometría de la capa, así como *Rampa de colores* y *Formato de texto* por defecto (por ejemplo, al activar el etiquetado). Cualquiera de estos elementos puede restablecerse utilizando la entrada *Clear* del widget desplegable correspondiente.

Opciones de símbolos

En el grupo *Opciones de símbolo*, puede:

- Aplicar una *Opacidad* por defecto a nuevas capas
- *Asignar colores aleatorios a los símbolos*, modificando los colores de relleno de los símbolos, evitando así la misma representación para todas las capas.

Base de datos de estilos

Una base de datos de estilos en QGIS es un repositorio estructurado diseñado para almacenar símbolos, formatos de texto y otros elementos de estilo. Sirve como una ubicación centralizada donde puede organizar y gestionar sus recursos de simbología de manera eficiente. Puede crear una base de datos de estilos dedicada para un cliente específico, que albergue símbolos adaptados a las necesidades de ese cliente. Esto garantiza un enfoque limpio y organizado, sin saturar la base de datos de estilos Predeterminada. En entornos multiusuario, es posible almacenar estilos específicos de un proyecto en una ubicación compartida. Al vincular un proyecto a estos estilos, todos los usuarios del proyecto tienen acceso a la simbología común, lo que elimina la necesidad de importar o actualizar manualmente la base de datos de estilos local individual. El almacenamiento de símbolos en un archivo de proyecto ofrece una solución práctica, ya que evita que los usuarios saturen su base de datos de estilos global con simbología específica del proyecto. Este enfoque garantiza que otros usuarios que trabajen en el mismo proyecto tengan acceso inmediato a toda la simbología necesaria al cargar el proyecto.

En la sección *Base de datos de estilos* puede elegir entre *Añadir* o *Eliminar* base de datos de estilos o puede *Crear nueva base de datos de estilos*. Cuando añada o elimine una base de datos de estilos en esta sección, los cambios se reflejarán automáticamente en *Propiedades de la simbología*.

5.3.7 Propiedades de los colores

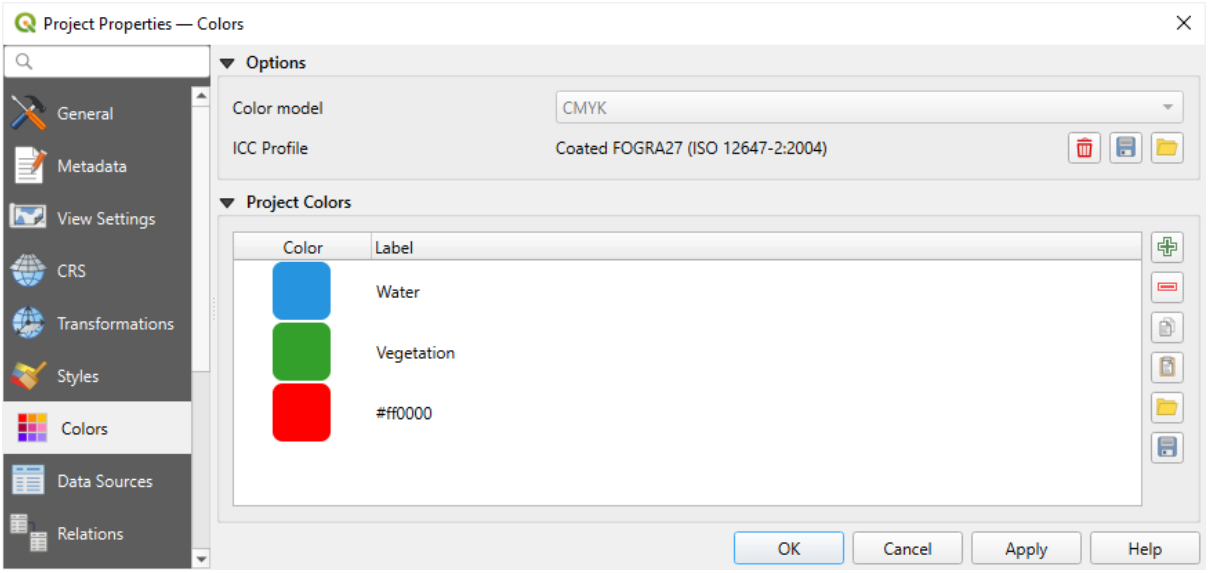


Figura 5.35: Pestaña Colores

En la pestaña *Colores*, puede seleccionar el *Modelo de color*, es decir, si al seleccionar un color debe definirse por defecto como `RGB` o `CMYK`. Cualquier color definido en un modelo de color diferente al especificado aquí se convertirá a este modelo de color al exportar una maqueta.

Additionally, an option is available for selecting an *ICC Profile* which allows you to *Load* a specific ICC profile that defines the color space for your project. After you load the ICC profile, you can choose to *Save ICC Profile* to a file on disk or to *Remove selected ICC Profile*. Once applied, QGIS will enforce the selected color model to ensure consistent color representation.

Nota

Note that the *ICC Profile* option will only be visible if QGIS is built with Qt 6.8.0 or higher.

También hay una sección adicional donde puede definir colores específicos para el proyecto en ejecución. Como los *global colors*, puede:

- *Añadir* o *Borrar* color
- *Copiar* o *Pegar* color
- *Importar* o *Exportar* el conjunto de colores de/a archivo `.gpl`.

Haga doble clic en un color de la lista para modificarlo o reemplazarlo en el diálogo *Color Selector*. También puede cambiarle el nombre haciendo doble click en la columna *Etiqueta*.

Estos colores son identificados como *Colores de proyecto* y listados como parte de *color widgets*.

Truco

Utilice los colores del proyecto para asignar y actualizar rápidamente widgets de color

Se puede hacer referencia a los colores del proyecto para usar su etiqueta y los widgets de color en los que se usan están vinculados a ellos. Esto significa que en lugar de establecer repetidamente el mismo color para muchas propiedades y, para evitar una actualización engorrosa, puede:

1. Define el color como color del proyecto
2. Click en el *data defined override widget* al lado de la propiedad color puede establecer
3. Pase el cursor sobre el menú *Color* y seleccione el color del proyecto. La propiedad es entonces asignada a la expresión `project_color('color_label')` y el widget de color refleja ese color.
4. Repita los pasos 2 y 3 tanto como necesite
5. Actualice el color del proyecto una vez y el cambio se reflejará POR TODAS LAS PARTES que estén en uso.

5.3.8 Propiedades de Fuentes de Datos

En la pestaña  *Fuentes de datos*, puede:

- *Modo de transacción*, define cómo se envían las ediciones al proveedor de datos:
  - *Búfer de edición local*: las ediciones se almacenan localmente y se envían al proveedor al cambiar el modo de edición de capas o al hacer clic en *Guardar ediciones*.
  - *Grupos de transacciones automáticas*: en las fuentes de datos compatibles (bases de datos postgres y geopackage), el estado de edición de todas las tablas que se originan en la misma base de datos se sincroniza y se ejecuta en una transacción del lado del servidor. Además, en lugar de almacenar los cambios de edición localmente, se envían directamente a una transacción en la base de datos que se confirma al cambiar el modo de edición de la capa o al hacer clic en *Guardar ediciones*.
  - *Grupos de transacciones con buffer*: todas las capas editables, independientemente del proveedor, se activan de forma sincrónica y todas las ediciones se guardan en un búfer de edición local. El guardado de cambios se ejecuta dentro de una única transacción en todas las capas (por proveedor).

  Tenga en cuenta que sólo puede cambiar esta opción si no se está editando ninguna capa en el proyecto.
-  *Evaluar valores predeterminados en el lado del proveedor*: al agregar nuevas funciones en una tabla PostgreSQL, los campos con restricción de valor predeterminado se evalúan y se completan en la apertura del formulario, y no en el momento de la confirmación. Esto significa que en lugar de una expresión como `nextval('serial')`, el campo en el formulario *Añadir entidad* mostrará el valor esperado (por ejemplo,``25``).
-  *Recordar estado de capa editable entre sesiones*: asegura que todas las capas que son editables en un proyecto serán recordadas como tales al guardar el proyecto, así como asegura que esas capas se hacen editables inmediatamente cada vez que se restaura el proyecto.
- Configure las *Capacidades de Capas*, p.ej.:
  - Establece (o desactiva) qué capas son `identificables`, es decir, responderán a la *herramienta identificar*. De forma predeterminada, las capas se pueden consultar.
  - Establezca si una capa debe aparecer como `solo lectura`, lo que significa que el usuario no puede editarla, independientemente de las capacidades del proveedor de datos. Aunque esta es una protección débil, sigue siendo una configuración rápida y práctica para evitar que los usuarios finales modifiquen datos cuando trabajan con capas basadas en archivos.
  - Definir qué capas son `buscables`, es decir, pueden consultarse mediante el *widget localizador*. De forma predeterminada, las capas se pueden buscar.
  - Define que capas son definidas como `requeridas`. Las capas marcadas en esta lista son protegidas de borrado involuntario del proyecto.

  - Define qué capas son `privadas`, es decir, ocultas del panel *Capas*. Esta opción está pensada para capas accesorias (mapa base, unión, búsquedas de relaciones de valores, probablemente capas espaciales, etc.) que aún se necesitan en un proyecto pero que no se desea que contaminen el árbol de leyenda y otras herramientas de selección de capas. Si se establecen como visibles, se siguen mostrando en el lienzo del mapa y se renderizan en la leyenda del diseño de impresión. Utilice la opción *Filtro leyenda* ► *Mostrar capas privadas* en la barra de herramientas superior del panel *Capas* para activarlas temporalmente para cualquier interacción.

  La tabla *Capacidades de capas* proporciona algunas herramientas convenientes para:

  - Seleccionar multiples celdas y presionar *Alternar selección* para que cambien su estado de la casilla de verificación;
  - *Mostrar solo capas espaciales*, filtrar capas no espaciales de la lista de capas;
  - *Filtrar capas…* y agilmente encuentre una capa en particular para configurar.

- En el grupo *Configuración avanzada*, puede seleccionar *Confiar en el proyecto cuando la fuente de datos no tiene metadatos*: Para acelerar la carga del proyecto omitiendo la comprobación de datos. Útil en el contexto de QGIS Server o en proyectos con grandes vistas de base de datos/vistas materializadas. La extensión de las capas se leerá del archivo de proyecto de QGIS (en lugar de las fuentes de datos) y cuando se utilice el proveedor PostgreSQL no se comprobará la unicidad de la clave primaria para las vistas y vistas materializadas.

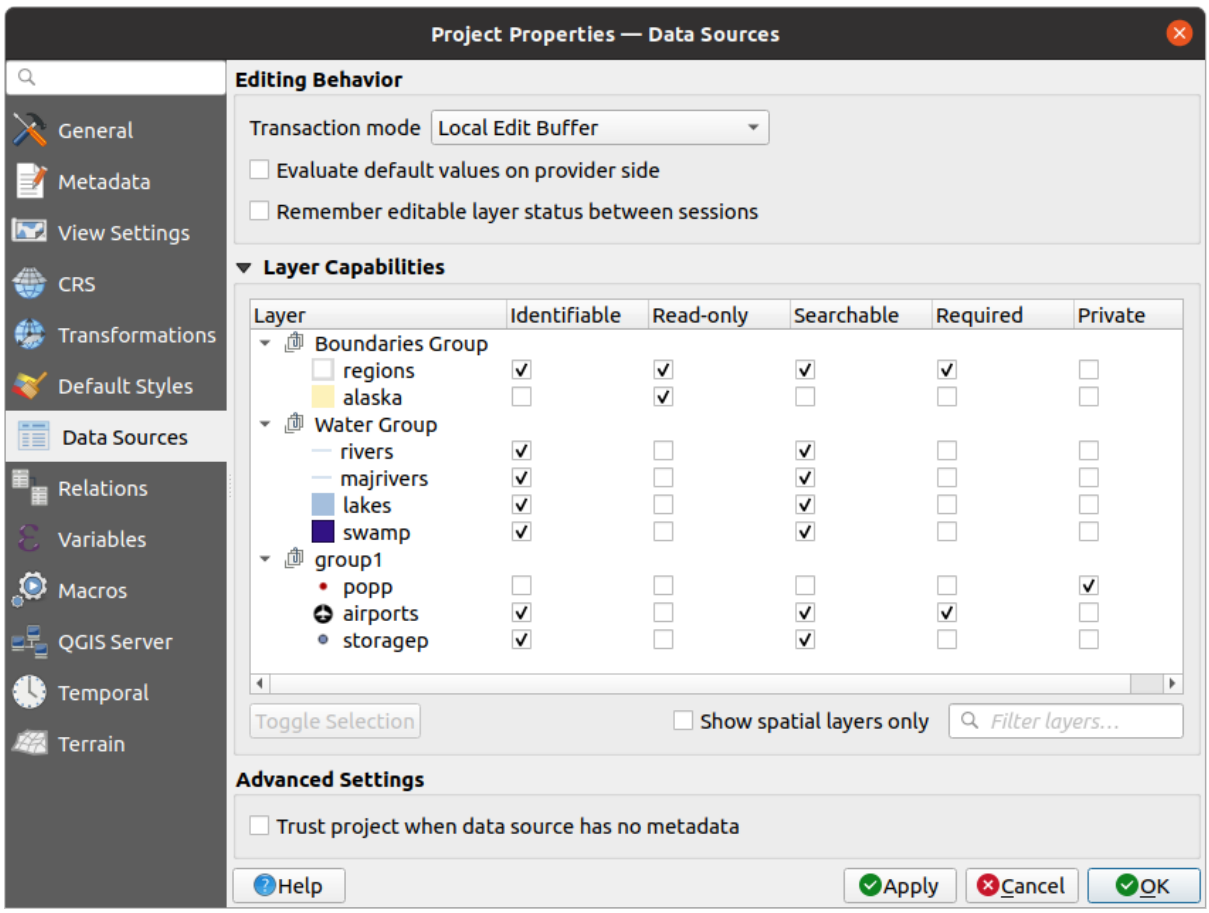


Figura 5.36: Pestaña de Fuentes de Datos

5.3.9 Propiedades de relación

La pestaña *Relaciones* se utiliza para definir relaciones entre capas. Las relaciones pueden ser de tipo uno a uno, muchos a muchos o polimórficas. Se definen en el diálogo de propiedades del proyecto. Una vez que existen relaciones para una capa, un nuevo elemento de la interfaz de usuario en la vista de formulario (por ejemplo, al identificar una característica y abrir su formulario) listará las entidades relacionadas. Esto proporciona una forma eficaz de expresar, por ejemplo, el historial de inspección de un tramo de tubería o de carretera. Puede obtener más información sobre el soporte de relaciones en la sección *Establecer relaciones entre varias capas*.

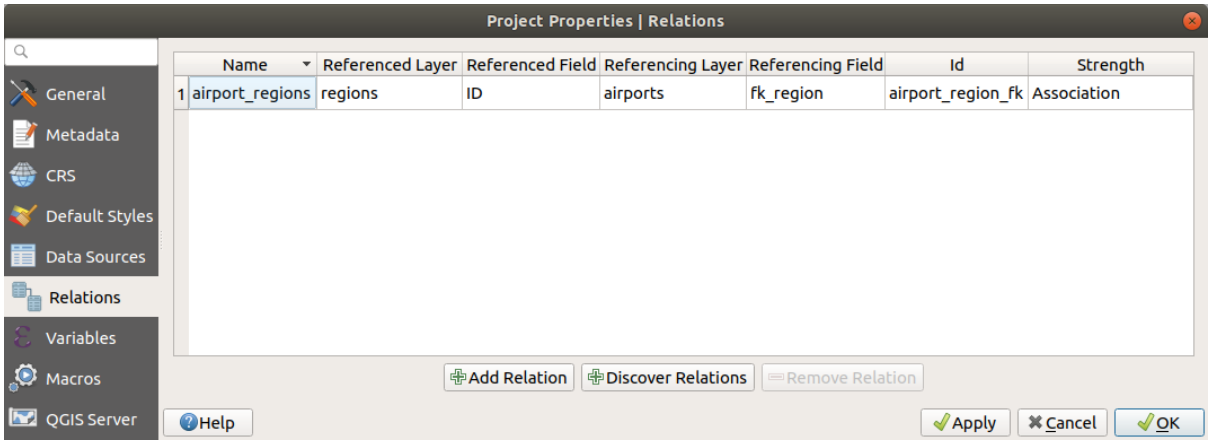


Figura 5.37: Pestaña Relaciones

5.3.10 Propiedades de variables

La pestaña *Variables* lista todas las variables disponibles a nivel de proyecto (que incluye todas las variables globales). Además, también permite al usuario gestionar las variables a nivel de proyecto. Haga clic en el botón para añadir una nueva variable personalizada a nivel de proyecto. Asimismo, seleccione una variable personalizada a nivel de proyecto de la lista y haga clic en el botón para eliminarla. Más información sobre el uso de variables en la sección Herramientas Generales *Almacenando valores en variables*.

5.3.11 Propiedades de Macros

La pestaña *Macros* se utiliza para editar las macros de Python de los proyectos. Actualmente, sólo hay tres macros disponibles: `openProject()`, `saveProject()` y `closeProject()`.

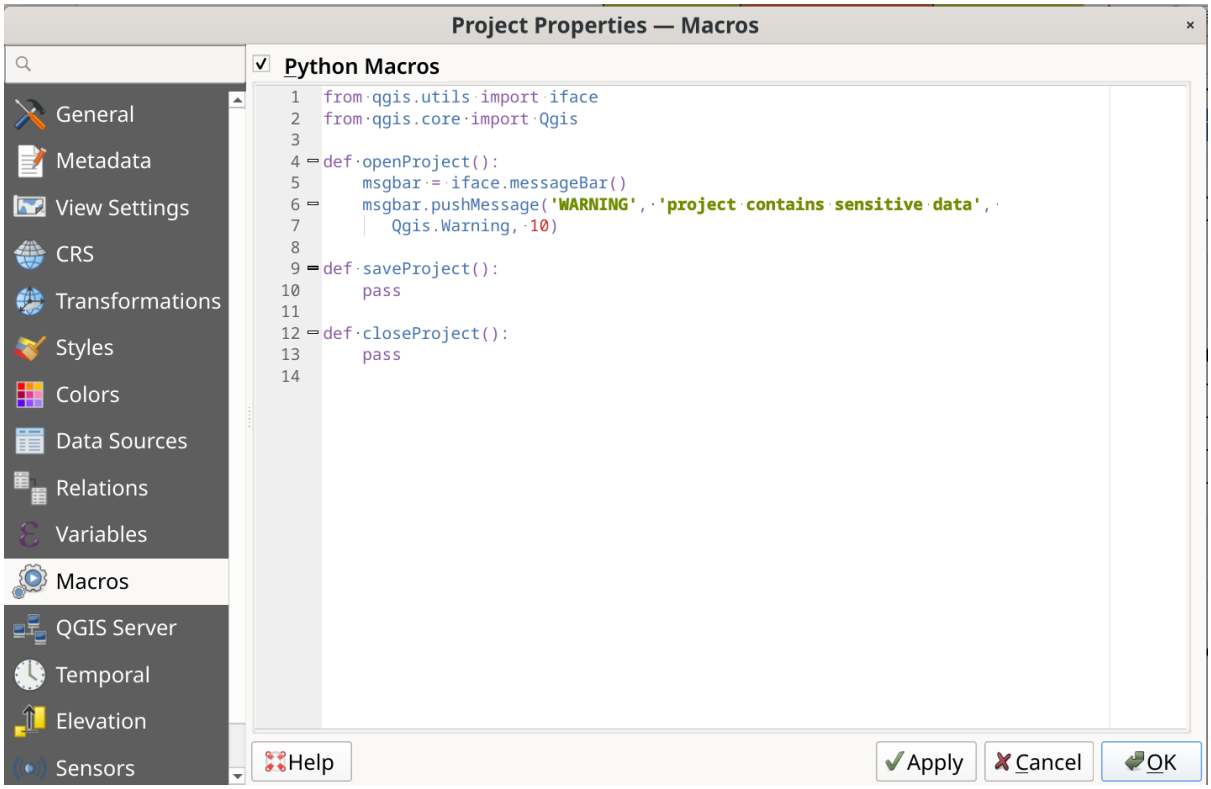


Figura 5.38: Ajustes macro

5.3.12 Propiedades de servidor QGIS

La pestaña *Servidor QGIS* le permite configurar su proyecto para publicarlo en línea. Aquí puede definir la información sobre las capacidades WMS y WFS del Servidor QGIS, la extensión y las restricciones SRC. Más información disponible en la sección Crearwmsdeproyecto y siguientes.

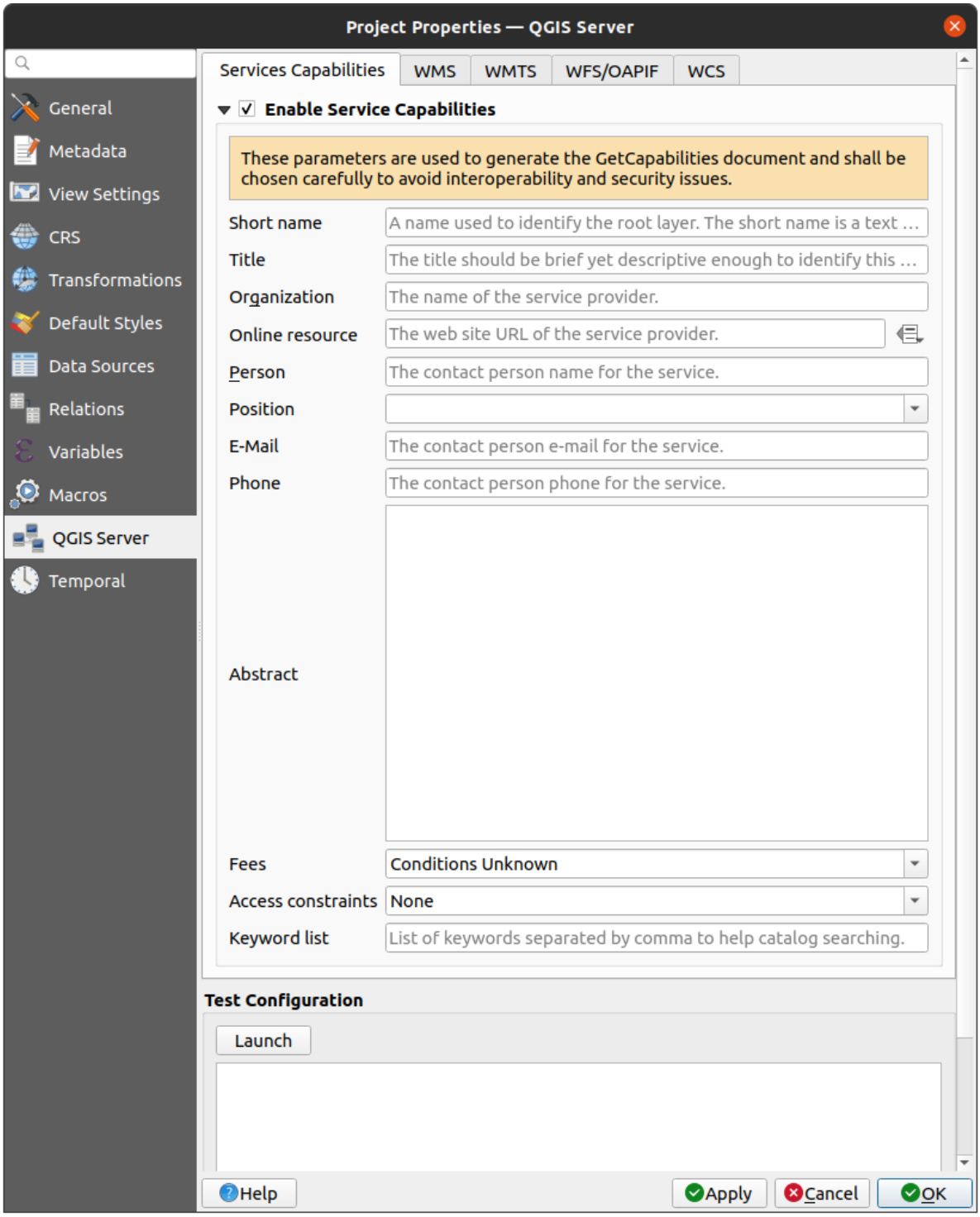


Figura 5.39: Configuración del servidor QGIS

5.3.13 Propiedades Temporales

La pestaña *Temporal* se utiliza para establecer el rango temporal de su proyecto, bien utilizando entradas manuales de *Fecha de inicio* y *Fecha de finalización* o calculándolo a partir de las capas temporales del proyecto actual. El rango temporal del proyecto puede utilizarse entonces en el panel del controlador *Temporal* para gestionar la navegación temporal del lienzo del mapa <maptimecontrol>`.

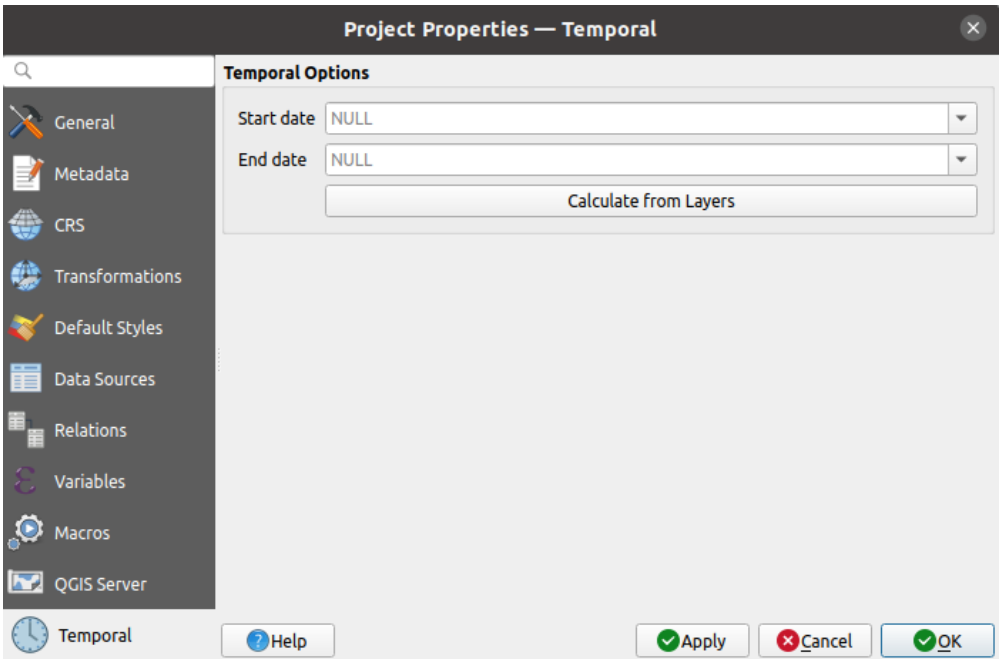


Figura 5.40: Pestaña Temporal del proyecto

5.3.14 Propiedades de elevación

La pestaña *Elevación* le ayuda a configurar los ajustes por defecto para el terreno y la elevación. Cuando cualquier nuevo *mapa 3D* en el proyecto, el mapa utilizará por defecto la misma configuración de terreno que la definida para el proyecto. Los ajustes de elevación del proyecto también serán respetados por la herramienta *perfil de elevación*.

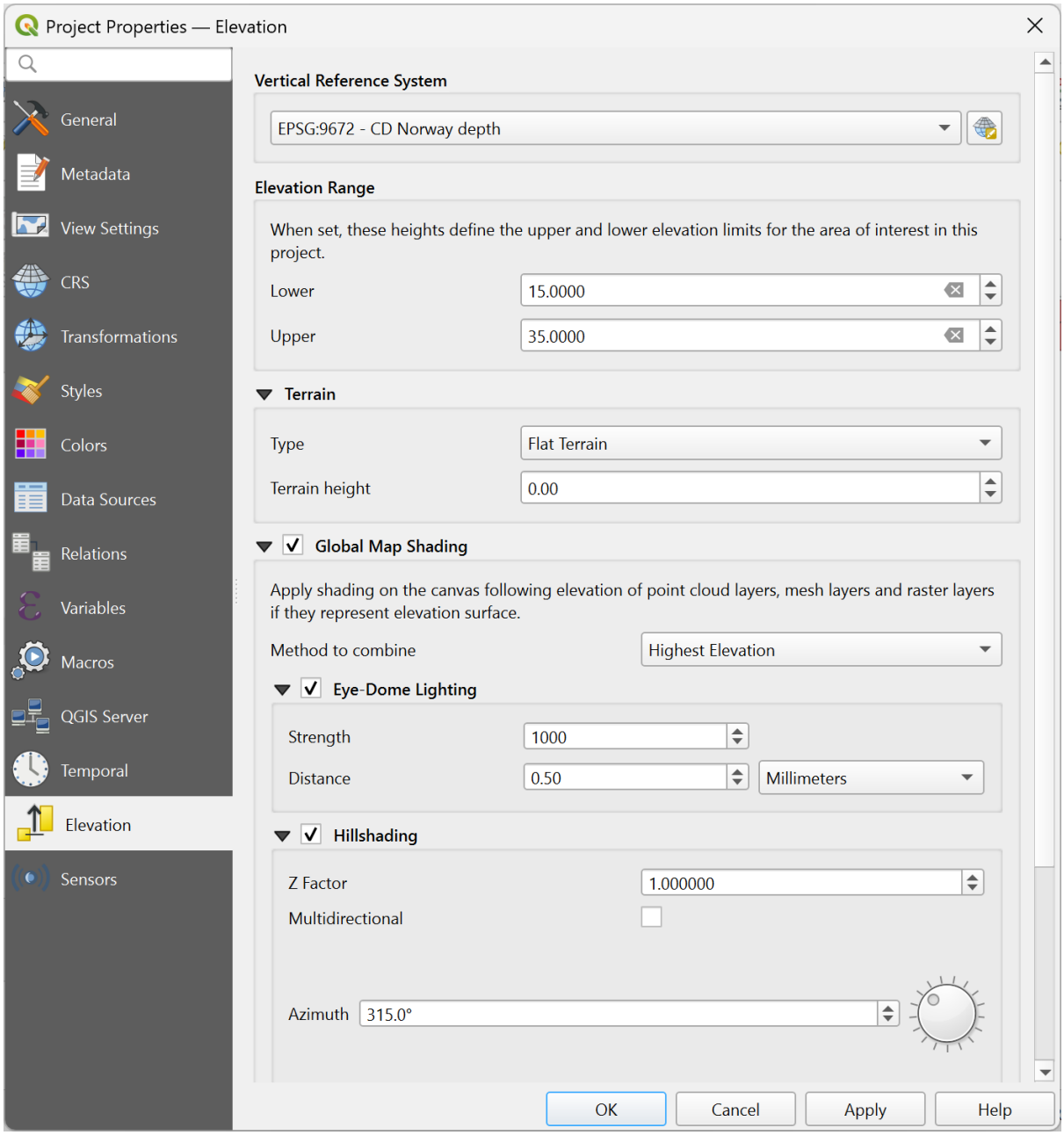


Figura 5.41: Pestaña de elevación del proyecto

- *Sistema de referencia vertical*: Si el SRC de su proyecto es compuesto uno (incluyendo una dimensión Z), entonces el SRC vertical utilizado para el proyecto será el componente vertical del SRC del proyecto. Si el SRC de su proyecto es horizontal, entonces puede seleccionar un SRC vertical específico haciendo clic en el botón Seleccionar SRC.
- *Elevation Range*: helps you define the *Lower* and *Upper* limits for the *elevation controller* in the project.
- *Terreno* y las opciones de elevación están disponibles para:
  - *Terreno plano* con *Altura del terreno* configurado
  - *DEM (Capa Raster)*: con parámetros para definir la *Capa raster*, un factor *Escala vertical* para aplicar a los valores de banda y un *Desplazamiento* vertical.
  - *Malla*: con parámetros para definir la capa *Malla*, un factor *Escala vertical* para aplicar al valor Z de los vértices y un valor vertical *Desplazamiento*.

Estos ajustes pueden sobrescribirse desde el cuadro de diálogo de configuración del mapa 3D ..

- Con la configuración *Sombreado global del mapa*, se aplica un efecto de sombreado global al mapa, basado en la elevación de todas las capas activas que tengan activadas las propiedades de elevación, incluyendo:
  - *capas ráster*
  - *capas malla*
  - *capas de nubes de puntos*

  Método para combinar: se combina el valor de elevación en cualquier posición concreta para todas las capas relevantes, y la elevación considerada se elige en función de uno de estos métodos:
  - valor *Mayor elevación*
  - *Basado en el orden de las capas*: se considera la elevación de la capa más alta del árbol de capas.

  Las opciones de sombreado admitidas son:
  - *Iluminación de la cúpula del ojo*: aplica efectos de sombreado al lienzo del mapa para un mejor renderizado de la profundidad. Se pueden controlar los siguientes parámetros:
    - *Fuerza*: aumenta el contraste, lo que permite una mejor percepción de la profundidad
    - *Distancia*: representa la distancia de los píxeles utilizados respecto al píxel central y tiene el efecto de hacer los bordes más gruesos.
  - *Relieve sombreado*, da forma a algunos relieves del mapa mediante sombreado (niveles de gris):
    - *Factor Z*: Factor de escala para el valor de elevación.
    - *Multidireccional*: Especifique si se va a utilizar el sombreado multidireccional.
    - *Azimut*: El azimut de la fuente de luz
    - *Altitud*: El ángulo de elevación de la fuente de luz

> **Nota**
>
> Un acceso directo a las propiedades de *Sombreado global del mapa* está disponible a través del panel *Estilos de capa*.

5.3.15 Propiedades de los sensores

La pestaña *Sensores* se utiliza para configurar los sensores y alternar su estado de conexión. Cuando están activos, los sensores recopilan datos de forma pasiva en segundo plano y ponen sus datos más recientes a disposición de expresiones y scripts python.

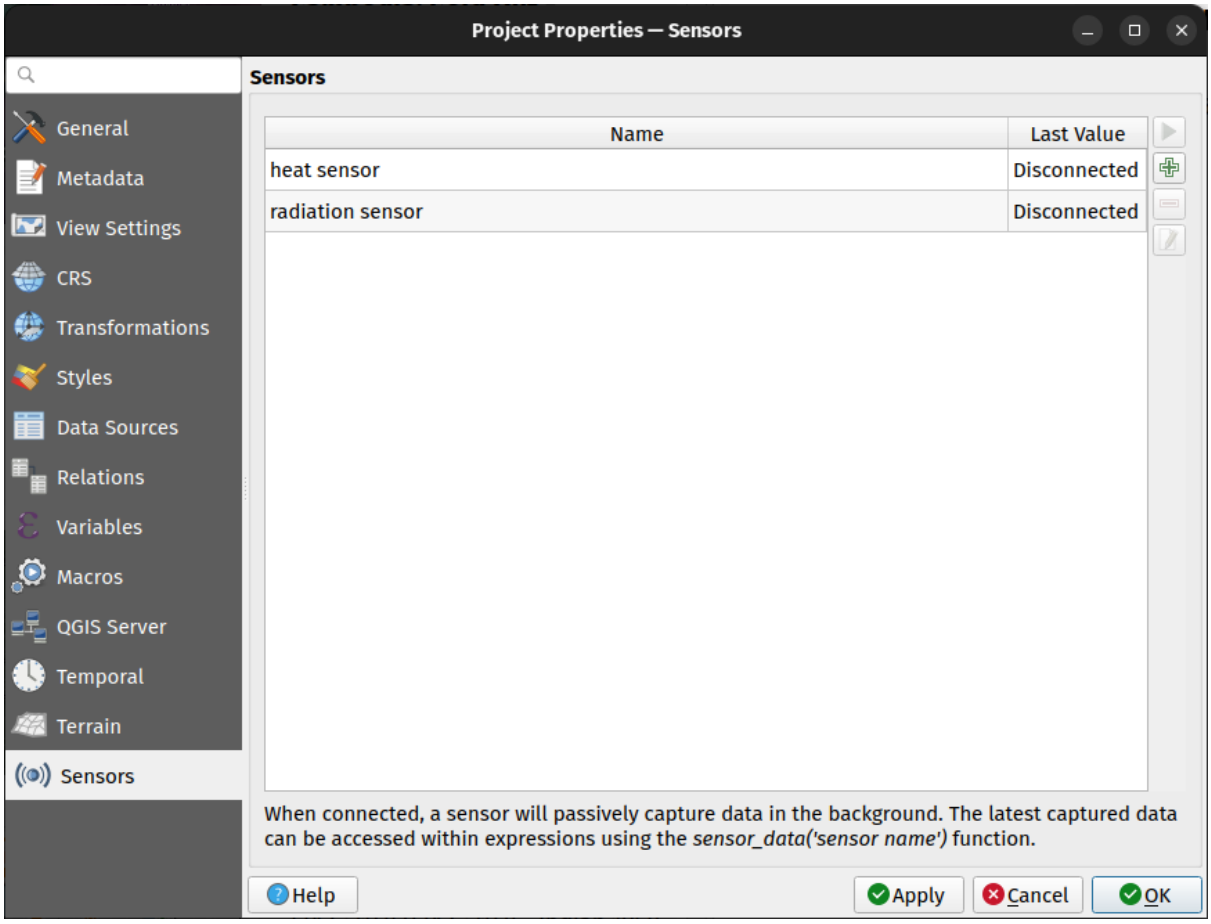


Figura 5.42: Pestaña Sensores del proyecto

Para añadir un nuevo sensor, pulse el botón . Se abrirá un subpanel de configuración que le permitirá configurar:

- la *Nombre del sensor*: se utiliza para recuperar los valores del sensor en expresiones y scripts python;
- la *Tipo de sensor*: TCP, UDP, puerto serie, etc.; y
- detalles adicionales específicos del tipo (por ejemplo, nombre de host y puerto).

  Para los sensores de puerto serie, se dispone de comboboxes con los puertos serie disponibles del sistema y la velocidad en baudios de la conexión para su selección. Se puede utilizar un *Data frame delimiter* para analizar los datos devueltos. En caso de que se encuentren múltiples tramas de datos en el último trozo de datos enviados por el puerto serie, se devuelven todas las tramas de datos. A continuación, puede utilizar una expresión para dividir el último valor de datos del sensor devuelto en marcos individuales (por ejemplo, utilizando la función *string_to_array*).

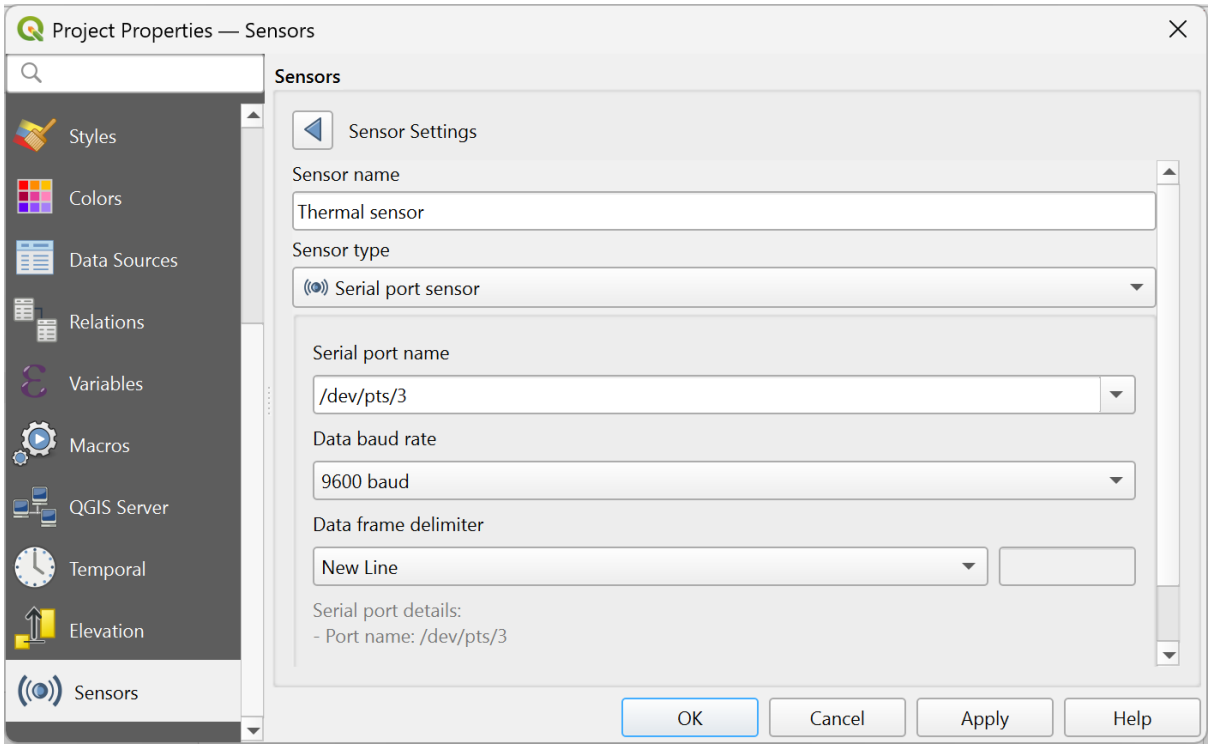


Figura 5.43: Subpanel de configuración del sensor

Una vez configurado un sensor, puede utilizar el botón Iniciar para conectar el sensor. Una vez activo, los últimos datos recogidos se mostrarán en la columna *Último valor* de la tabla de sensores.

5.4 Personalización

El diálogo *Personalización* le permite (des)activar casi todos los elementos de la interfaz de usuario de QGIS. Esto puede ser muy útil si quiere proporcionar a sus usuarios finales una versión «ligera» de QGIS, que contenga sólo los iconos, menús o paneles que necesitan.

Nota

Antes de aplicar los cambios, debe reiniciar QGIS.

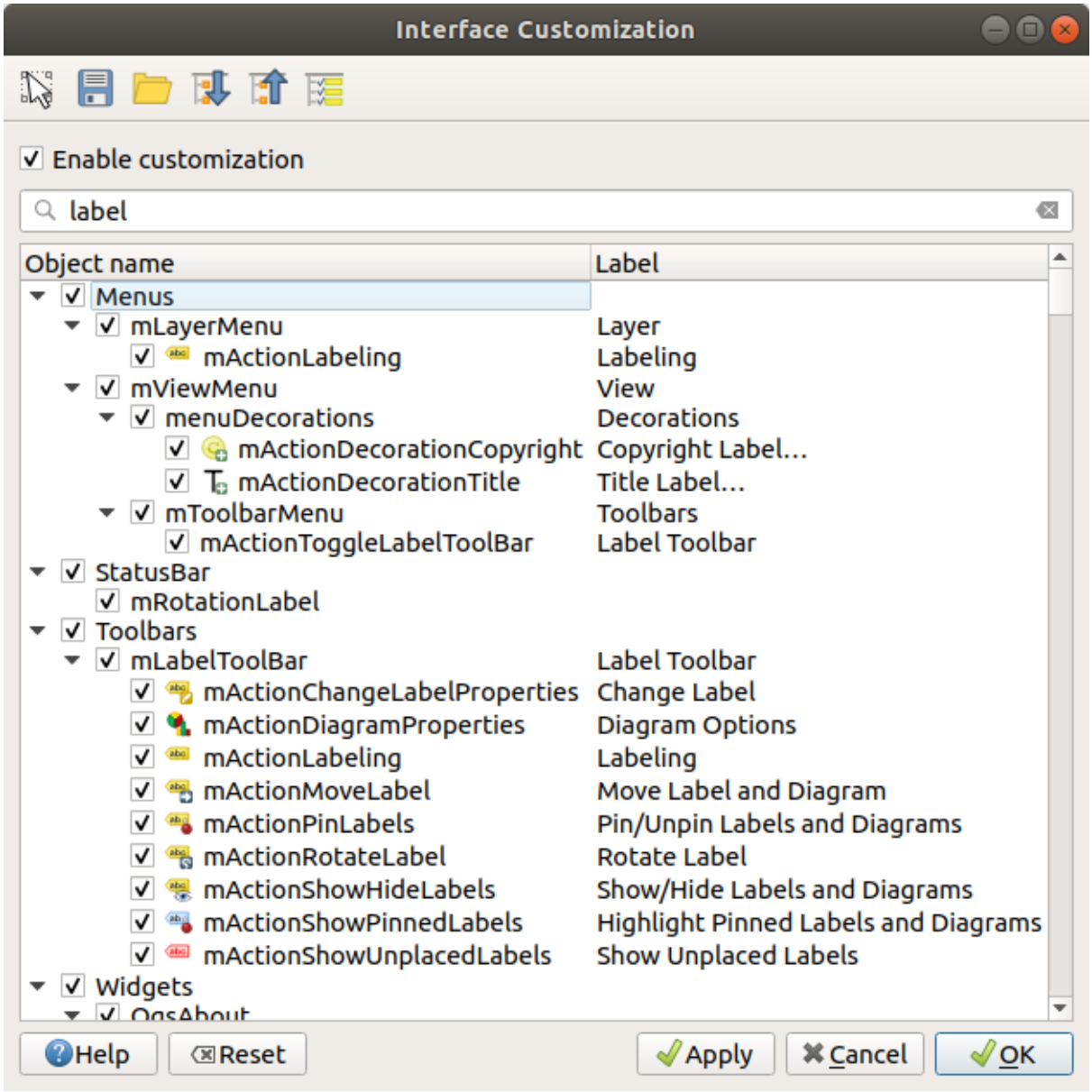


Figura 5.44: El diálogo de Personalización

Marcando la casilla *Habilitar personalización* es el primer paso en el camino a personalizar QGIS. Esto activa la barra de herramientas y el panel de widget desde el cual puede desmarcar y así deshabilitar algunos elementos de GUI.

El elemento configurable puede ser:

- un **Menu** o alguno de sus submenús de la ref:*label_menubar*
- un **Panel** entero (ver *Paneles y Barras de Herramientas*)
- la **Barra de Estado** descrita en *Barra de Estado* o algunos de sus elementos
- una **Barra de Herramientas**: la barra entera o alguno de sus iconos
- o alg'un **widget** de algún diálogo en QGIS: etiqueta, botón, combobox…

Con Cambie a los widgets de captura en la aplicación principal, puede hacer click en un elemento en la interfaz QGIS que desea ocultar y QGIS desmarca automáticamente la entrada correspondiente en el cuadro de diálogo Personalización. También puede usar el cuadro *Buscar* para buscar elementos por su nombre o etiqueta.

Una vez que haya establecido su configuración, click *Aplicar* o *OK* to para validar sus cambos. Esta configuración se conviertirá en la utilizada por defecto por QGIS en el próximo inicio.

Las modificaciones también se pueden guardar en un archivo `.ini` usando el botón  Guardar a archivo. Esta es un modo manejable para compartir una interfaz común de QGIS entre múltiples usaruios. Solo haga click en  Cargar desde archivo desde el ordenador de destino con el objetivo de importar el archivo `.ini`. También puedes ejecutar *command line tools* y guarde varias configuraciones para diferentes casos de uso también.

Truco

Restaurar fácilmente el QGIS predefinido

La configuración inicial de QGIS GUI puede ser reestablecida por uno de los siguientes métodos:

- desmarcando la opción  *Habilitar personalización* en el diálogo Personalización o haga click en el botón  Check All
- presionando el botón *Reset* en el cuadro **Configuración** debajo del menú *Settings* ► *Options*, pestaña *System*
- iniciar QGIS en un símbolo del sistema con la siguiente línea de comando `qgis --nocustomization`
- estableciendo a `falso` el valor de la variable *UI* ► *Personalización* ► *Habilitado* en el menú *Ajustes* ► *Opciones*, pestaña *Avanzado* (ver *warning*).

En la mayoría de los casos, debe reiniciar QGIS para que se aplique el cambio.

5.5 Atajos de teclado

QGIS proporciona atajos de teclado predeterminados para muchas funciones. Puedes encontrarlos en la sección *Barra de Menú*. Además, la opción de menú *Configuración ->`\keyboardShortcuts\ :menuselection:`Atajos de teclado…* le permite cambiar los atajos de teclado predeterminados y agregar otros nuevos a las funciones de QGIS.

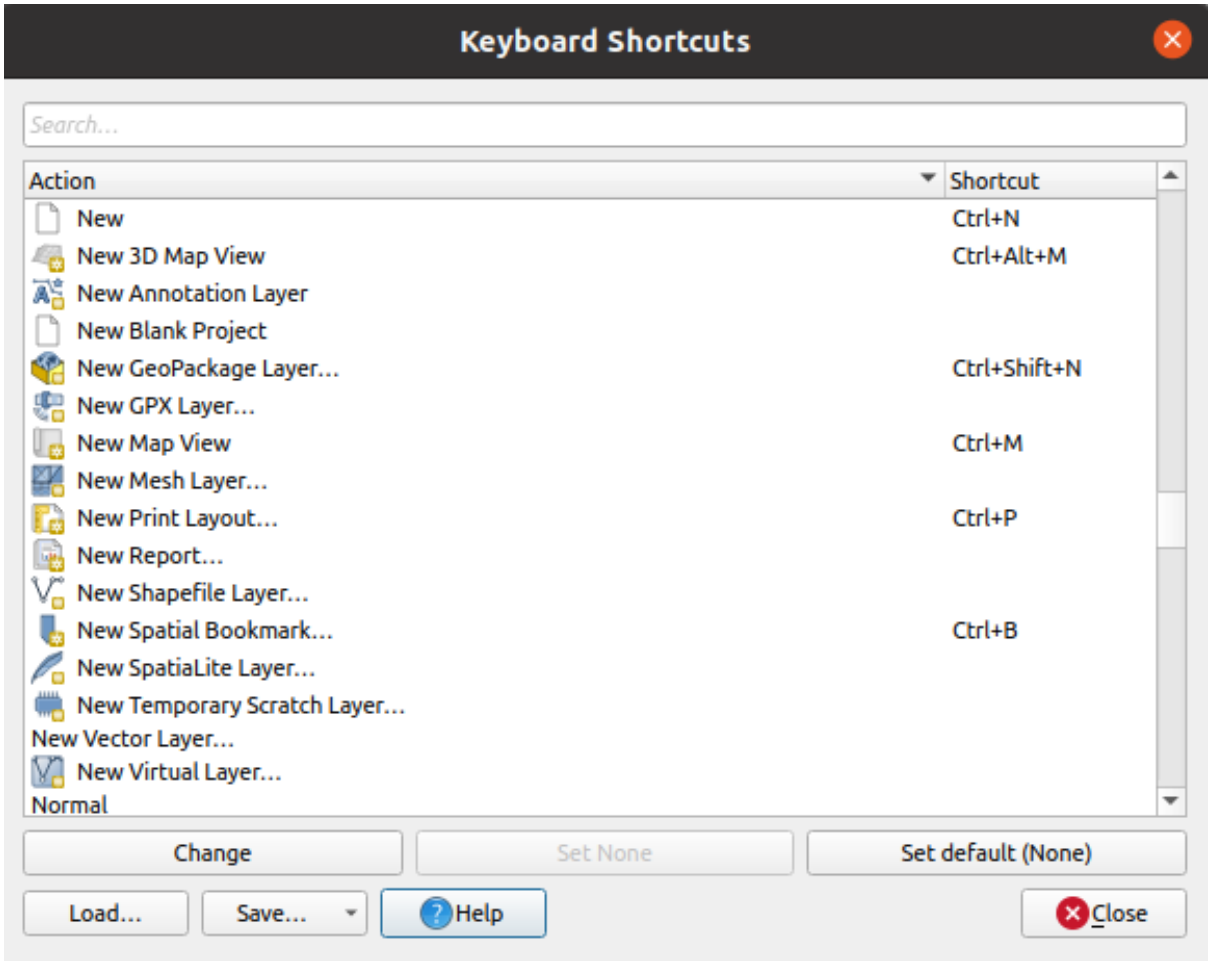


Figura 5.45: Defina las opciones de atajos

La configuración es muy simple. Use el cuadro de búsqueda en la parte superior del cuadro de diálogo para encontrar una acción en particular, selecciónela de la lista y haga click en:

- *Cambio* y presione la nueva combinación que quiera asignar como nuevo atajo
- *Establecer ninguno* para limpiar cualquier atajo asignado
- o *Establecer Predeterminado* para hacer una copia de seguridad del acceso directo a su valor original y predeterminado.

Proceda como se indicó anteriormente para cualquier otra herramienta que desee personalizar. Una vez que haya terminado su configuración, simplemente *Close* el cuadro de diálogo para que se apliquen los cambios. También puede *Save* los cambios como un archivo `.XML` con solo los accesos directos de usuario o con todos los accesos directos o como un archivo `.PDF` archivo con todos los accesos directos y *Load* en otra instalación de QGIS.

5.6 Ejecutando QGIS con ajustes avanzados

5.6.1 La línea de comandos y variables de entorno

Hemos visto que *iniciar QGIS* se realiza como cualquier otra aplicación en su sistema operativo. QGIS ofrece opciones de línea de comandos para casos de uso más avanzados (en algunos casos, puede utilizar una variable de entorno en lugar de la opción de línea de comandos). Para obtener una lista de las opciones, introduzca «qgis –help» en la línea de comandos, lo que devolverá:

```
QGIS is a user friendly Open Source Geographic Information System.
Usage: /usr/bin/qgis.bin [OPTION] [FILE]
  OPTION:
        [-v, --version]     display version information and exit
        [-s, --snapshot filename]   emit snapshot of loaded datasets to given file
        [-w, --width width] width of snapshot to emit
        [-h, --height height]       height of snapshot to emit
        [-l, --lang language]       use language for interface text (changes␣
↪existing override)
        [-p, --project projectfile] load the given QGIS project
        [-e, --extent xmin,ymin,xmax,ymax]  set initial map extent
        [-n, --nologo]      hide splash screen
        [-V, --noversioncheck]      don't check for new version of QGIS at startup
        [-P, --noplugins]   don't restore plugins on startup
        [-B, --skipbadlayers]     don't prompt for missing layers
        [-C, --nocustomization]     don't apply GUI customization
        [-z, --customizationfile path]      use the given ini file as GUI␣
↪customization
        [-g, --globalsettingsfile path]     use the given ini file as Global␣
↪Settings (defaults)
        [-a, --authdbdirectory path] use the given directory for authentication␣
↪database
        [-f, --code path]   run the given python file on load
        [-F, --py-args arguments]   arguments for python. These arguments will be␣
↪available for each python execution via 'sys.argv' including the file specified␣
↪by '--code'. All arguments till '--' are passed to python and ignored by QGIS.
        [-d, --defaultui]   start by resetting user ui settings to default
        [--hide-browser]        hide the browser widget
        [--dxf-export filename.dxf]     emit dxf output of loaded datasets to␣
↪given file
        [--dxf-extent xmin,ymin,xmax,ymax]      set extent to export to dxf
        [--dxf-symbology-mode none|symbollayer|feature] symbology mode for dxf␣
↪output
        [--dxf-scale-denom scale]       scale for dxf output
        [--dxf-encoding encoding]       encoding to use for dxf output
        [--dxf-map-theme maptheme]      map theme to use for dxf output
        [--take-screenshots output_path]        take screen shots for the user␣
↪documentation
        [--screenshots-categories categories]   specify the categories of␣
↪screenshot to be used (see QgsAppScreenShots::Categories).
        [--profile name]        load a named profile from the user's profiles␣
↪folder.
        [-S, --profiles-path path]  path to store user profile folders. Will␣
↪create profiles inside a {path}\profiles folder
        [--version-migration]   force the settings migration from older version if␣
↪found
        [--openclprogramfolder]         path to the folder containing the sources␣
↪for OpenCL programs.
        [--help]                this text
        [--]            treat all following arguments as FILEs

  FILE:
```

(continúe en la próxima página)

(proviene de la página anterior)

```
   Files specified on the command line can include rasters,
   vectors, and QGIS project files (.qgs and .qgz):
    1. Rasters - supported formats include GeoTiff, DEM
       and others supported by GDAL
    2. Vectors - supported formats include ESRI Shapefiles
       and others supported by OGR and PostgreSQL layers using
       the PostGIS extension
```

Truco

Ejemplo usando argumentos de línea de comando

Puede iniciar QGIS especificando uno o más archivos de datos en la línea de comando. Por ejemplo, suponiendo que estás en el directorio `qgis_sample_data`, puede iniciar QGIS con una capa vectorial y un archivo ráster configurado para cargar al inicio utilizando el siguiente comando: `qgis ./raster/landcover.img ./gml/lakes.gml`

`--version`

Esta opción devuelve información de la versión de QGIS

`--snapshot`

Esta opción le permite crear una instantánea en formato PNG desde la vista actual. Esto resulta útil cuando tiene muchos proyectos y desea generar instantáneas a partir de sus datos, o cuando necesita crear instantáneas del mismo proyecto con datos actualizados.

Actuálmente, genera un archivo PNG con 800x600pixels. El tamaño puede ser ajustado usando los argumentos `--width` y `--height`. El nombre de archivo puede ser añadido despues de `--snapshot`. Por Ejemplo:

```
qgis --snapshot my_image.png --width 1000 --height 600 --project my_project.qgs
```

`--width`

Esta opción devuelve la anchura de la captura de imagen a emitir (usado con `--snapshot`).

`--height`

Esta opción devuelve la altura de la captura de imagen a emitir (usado con `--snapshot`).

`--lang`

Según su entorno local, QGIS selecciona la localización correcta. Si desea cambiar su idioma, puede especificar un código de idioma. Por ejemplo, `qgis --lang it` comienza QGIS en localización italiana.

--project

También es posible iniciar QGIS con un archivo de proyecto existente. Simplemente agregue la opción de línea de comando `--project` seguido del nombre de su proyecto y QGIS se abrirá con todas las capas en el archivo dado cargado.

--extent

Para comenzar con una extensión de mapa específica, use esta opción. Debe agregar el cuadro delimitador de su extensión en el siguiente orden, separado por una coma:

```
--extent xmin,ymin,xmax,ymax
```

Esta opción probablemente tenga más sentido cuando se combina con la opción `--proyecto` para abrir un proyecto específico en la medida deseada.

--nologo

Esta opción oculta la pantalla de inicio cuando inicia QGIS.

--noversioncheck

Saltar la búsqueda de una nueva versión de QGIS al arrancar.

--noplugins

Si tiene problemas al iniciar con complementos, puede evitar cargarlos al inicio con esta opción. Todavía estarán disponibles desde el Administrador de complementos después.

--nocustomization

Usando esta opción, cualquier *GUI customization* existente no será aplicada al arrancar. Esto significa que cualquier botón oculto, elemento de menú, barra de herramientas, etc., aparecerá en el inicio de QGIS. Este no es un cambio permanente. La personalización se aplicará nuevamente si QGIS se inicia sin esta opción.

Esta opción es útil para permitir temporalmente el acceso a herramientas que se han eliminado mediante personalización.

--skipbadlayers

Usando esta opción, puede evitar que QGIS muestre el diálogo *Manejar capas no disponibles* al inicio. El fichero de proyecto se carga, manteniendo las capas que faltan como no disponibles. Más detalles sobre el tema en *Manejo de rutas de archivo rotas*.

--customizationfile

Con esta opción, puede definir un archivo de personalización de la interfaz de usuario, que se utilizará al inicio.

--globalsettingsfile

La variable de entorno equivalente es `QGIS_GLOBAL_SETTINGS_FILE`.

Con esta opción, puede especificar la ruta para un archivo de configuración global (`.ini`), también conocido como configuración predeterminada. Las configuraciones en el archivo especificado reemplazan las predeterminadas en línea originales, pero las configuraciones de los perfiles de usuario se establecerán encima de ellas.

QGIS busca el archivo de configuración global predeterminado en el siguiente orden y solo se utilizará el primer archivo encontrado:

- ruta especificada por el parámetro de la línea de comandos
- ruta definida por la variable de entorno
- la carpeta AppDataLocation, donde se pueden almacenar los datos persistentes de la aplicación; es administrado por el usuario o el administrador del sistema y no es tocado por el instalador y no requiere ninguna configuración adicional como pasar parámetros de línea de comandos o variables de entorno de configuración. Dependiendo del sistema operativo, es:
  - `$HOME/.local/share/QGIS/QGIS3/`
  - `C:\Users\<username>\%AppData%\Roaming\QGIS\QGIS3\`
  - `$HOME/Library/Application Support/QGIS/QGIS3/`
- el directorio de instalación, es decir, `your_QGIS_package_path/resources/qgis_global_settings.ini`.

Actualmente, no hay forma de especificar un archivo para escribir la configuración; por lo tanto, puede crear una copia de un archivo de configuración original, cambiarle el nombre y adaptarlo.

Estableciendo la ruta de archivo `qgis_global_setting.ini` a una carpeta compartida de red, permite que un administrador del sistema cambie la configuración global y los valores predeterminados en varias máquinas editando solo un archivo.

--authdbdirectory

Esta opción es similar a `--globalsettingsfile`, pero define la ruta al directorio donde se almacenará y cargará la base de datos de autenticación.

--code

Esta opción se puede utilizar para ejecutar un archivo de Python determinado directamente después de que QGIS se haya iniciado.

Por ejemplo, cuando tienes un nombre de archivo de Python `load_alaska.py` con el siguiente contenido:

```
from qgis.utils import iface
raster_file = "/home/gisadmin/Documents/qgis_sample_data/raster/landcover.img"
layer_name = "Alaska"
iface.addRasterLayer(raster_file, layer_name)
```

Suponiendo que estás en el directorio donde se encuentra el archivo `load_alaska.py`, puedes iniciar QGIS, cargar el archivo raster `landcover.img` y dar a la capa el nombre “Alaska” usando el siguiente comando:

```
qgis --code load_alaska.py
```

--py-args

Esta opción permite pasar argumentos al script proporcionados a través del *–code* parámetro y más en general a cada ejecución de python. Después de `--py-args`, no es posible pasar ningún otro parámetro que no sea –. a QGIS. Todos los argumentos después de `--py-args` hasta `--` se pasan al intérprete de python y son ignorados por QGIS.

```
qgis --code /home/user/test.py --py-args --specialScriptArguments "a text arg"
↪'and another arg' -- layer1 layer2
```

En el código anterior, `test.py` tendrá este contenido en `sys.argv`: [“/home/usuario/prueba.py”, “–specialScriptArguments”, “un arg de texto”, “y otro arg”]. Las capas `layer1` y `layer2` serán manejadas normalmente por QGIS como capas a cargar.

--defaultui

Al cargar, **restablecer permanentemente** la interfaz de usuario (IU) a la configuración predeterminada. Esta opción restaurará la visibilidad, posición y tamaño de los paneles y barras de herramientas. A menos que se cambie nuevamente, la configuración predeterminada de la IU se usará en las siguientes sesiones.

Tenga en cuenta que esta opción no tiene ningún efecto sobre la *personalización de GUI*. Los elementos ocultos por personalización de la GUI (por ejemplo, la barra de estado) permanecerán ocultos incluso utilizando la opción `--defaultui`. Véase también la opción `--nocustomization`.

--hide-browser

Al cargar, oculta el panel *Navegador* de la IU. El panel se puede habilitar haciendo click derecho en un espacio en las barras de herramientas o usando *Ver* -> *Paneles* (:menuselection:` Ajustes -> Paneles` en  Linux KDE).

A menos que se vuelva a habilitar, el panel del navegador permanecerá oculto en las siguientes sesiones.

--dxf-*

Estas opciones se pueden usar para exportar un proyecto QGIS a un archivo DXF. Hay varias opciones disponibles:

- *–dxf-export*: el nombre de archivo DXF al que exportar las capas;
- *–dxf-extent*: la extensión del archivo DXF final;
- *–dxf-symbology-mode*: Aquí se pueden usar varios valores: `none` (sin simbología), `symbollayer``(simbología de capa de símbolo), ``feature` (simbología de entidad);
- *–dxf-scale-denom*: el denominador de escala de la simbología;
- *–dxf-encoding*: el codificado del archivo;
- *–dxf-map-theme*: elegir un *map theme* del árbol de configuración de la capa.

--take-screenshots

Toma capturas de pantalla para la documentación del usuario. Se puede usar junto con `--screenshots-categories` para filtrar qué categorías/secciones de las capturas de pantalla de la documentación se deben crear (ver QgsAppScreenShots::Categories).

--profile

Carga QGIS utilizando un perfil específico de la carpeta de perfiles del usuario. Si el perfil nombrado no existe, será creado. Esta opción tiene prioridad sobre *configuración de inicio de perfil de usuario*.

--profiles-path

Con esta opción, puede elegir una ruta para cargar y guardar los perfiles (configuración del usuario). Crea perfiles dentro de una carpeta `{path}\profiles`, que incluye configuraciones, complementos instalados, modelos de procesamiento y scripts, etc.

Esta opción le permite, por ejemplo, llevar todos sus complementos y configuraciones en una unidad flash o, por ejemplo, compartir la configuración entre diferentes computadoras utilizando un servicio de intercambio de archivos.

La variable de entorno equivalente es `QGIS_CUSTOM_CONFIG_PATH`.

--version-migration

Si se encuentran configuraciones de una versión anterior (*por ejemplo*, la carpeta `.qgis2` de QGIS 2.18), esta opción las importará al perfil QGIS predeterminado.

--openclprogramfolder

Con esta opción, puede especificar una ruta alternativa para sus programas OpenCL. Esto es útil para los desarrolladores mientras prueban nuevas versiones de los programas sin necesidad de reemplazar las existentes.

La variable de entorno equivalente es `QGIS_OPENCL_PROGRAM_FOLDER`.

--

Esta opción permite pasar los siguientes argumentos como archivos a abrir en QGIS. Pueden ser archivos ráster o vectoriales, archivos de definición de capas de QGIS o archivos de proyecto.

```
qgis -- /home/user/project.qgs /home/user/path_to_data/layer1.gpkg
```

El código anterior iniciará QGIS, abrirá el archivo `project` y cargará `layer1` en él.

5.6.2 Implementar QGIS dentro de una organización

Si necesita implementar QGIS dentro de una organización con un archivo de configuración personalizado, primero debe copiar/pegar el contenido del archivo de configuración predeterminado ubicado en `your_QGIS_package_path/resources/qgis_global_settings.ini`. Este archivo ya contiene algunas secciones predeterminadas identificadas por un bloque que comienza con `[]`. Le recomendamos que mantenga estos valores predeterminados y agregue sus propias secciones al final del archivo. Si una sección está duplicada en el archivo, QGIS tomará la última de arriba a abajo.

Puede cambiar `allowVersionCheck=false` para deshabilitar la comprobación de versión de QGIS.

Si no desea mostrar la ventana de migración después de una nueva instalación, necesita la siguiente sección:

```
[migration]
fileVersion=2
settings=true
```

Si desea agregar una variable personalizada en el ámbito global:

```
[variables]
organisation="Your organization"
```

A lot of settings can be set using the `.INI` file such as WMS/WMTS, PostgreSQL connections, proxy settings, maptips, default values for algorithms parameters… For example, if you want to configure organization wide default values for some algorithms it would be handy if you could configure it in the prepared ini-file for all users. If the users then open the toolbox to execute an algorithm, the default values will be there. To set the default value, e.g., for `GRID_SIZE` parameter for the `native:intersection` algorithm, you could write the following into the `.INI` file:

```
[Processing]
DefaultGuiParam\native%3Aintersection\GRID_SIZE=0.01
```

No todas las configuraciones pueden prepararse desde QGIS GUI, pero para descubrir las posibilidades del archivo de configuración `.INI`, le sugerimos que establezca la configuración que desee en QGIS Desktop y luego la busque en su archivo `.INI` ubicado en su perfil utilizando un editor de texto.

Finalmente, debe establecer la variable de entorno `` QGIS_GLOBAL_SETTINGS_FILE “” en la ruta de su archivo personalizado.

Además, también puede implementar archivos como macros de Python, paletas de colores, plantillas de diseño, plantillas de proyectos … en el directorio del sistema QGIS o en el perfil de usuario QGIS.

- Las plantillas de diseño deben implementarse en el directorio `composer_templates`.
- Las plantillas de proyecto deben implementarse en el directorio `project_templates`.
- Las macros de Python personalizadas deben implementarse en el directorio `python`.

CAPÍTULO 6

Trabajar con Proyecciones

Un Sistema de Referencia de Coordenadas, o CRS, es un método para asociar coordenadas numéricas con una posición en la superficie de la Tierra. QGIS tiene soporte para aproximadamente 7,000 CRS estándar, ¡ cada uno con diferentes casos de uso, pros y contras! Elegir un sistema de referencia apropiado para sus proyectos y datos de QGIS puede ser una tarea compleja, pero afortunadamente QGIS le ayuda a guiarse a través de esta elección y hace que trabajar con diferentes CRS sea lo más transparente y preciso posible.

6.1 Vista general de la ayuda de proyección

QGIS tiene soporte para aproximadamente 7,000 CRS conocidos. Estos CRS estándar se basan en los definidos por el Grupo Europeo de Búsqueda de Petróleo (EPSG) y el Institut Geographique National de France (IGNF), y están disponibles en QGIS a través de la biblioteca de proyección subyacente «Proj». Comúnmente, estas proyecciones estándar se identifican mediante el uso de una autoridad: combinación de código, donde la autoridad es un nombre de organización como «EPSG» o «IGNF», y el código es un número único asociado con un CRS específico. Por ejemplo, el CRS de latitud/longitud WGS 84 común se conoce por el identificador `EPSG: 4326`, y el CRS estándar de mapeo web es `EPSG:3857`.

Los CRS personalizados creados por el usuario se almacenan en una base de datos de CRS de usuario. Consulte la sección *Sistema de referencia de coordenadas personalizada* para obtener información sobre cómo administrar sus sistemas de referencia de coordenadas personalizados.

6.2 Sistemas de Coordenadas de Capas

In order to correctly project data into a specific target CRS, either your data must contain information about its coordinate reference system or you will need to manually assign the correct CRS to the layer. For PostgreSQL layers, QGIS uses the spatial reference identifier that was specified when that PostgreSQL layer was created. For data supported by GDAL, QGIS relies on the presence of a recognized means of specifying the CRS. For instance, for the Shapefile format this is a file containing an ESRI Well-Known Text (WKT) representation of the layer's CRS. This projection file has the same base name as the `.shp` file and a `.prj` extension. For example, `alaska.shp` would have a corresponding projection file named `alaska.prj`.

Cada vez que se carga una capa en QGIS, QGIS intenta determinar automáticamente el CRS correcto para esa capa. En algunos casos esto no es posible, p.ej. cuando se ha proporcionado una capa sin retener esta información. Puede configurar el comportamiento de QGIS siempre que no pueda determinar automáticamente el CRS correcto para una capa:

1. Abra *Ajustes* ► *Opciones…* ► *CRS*

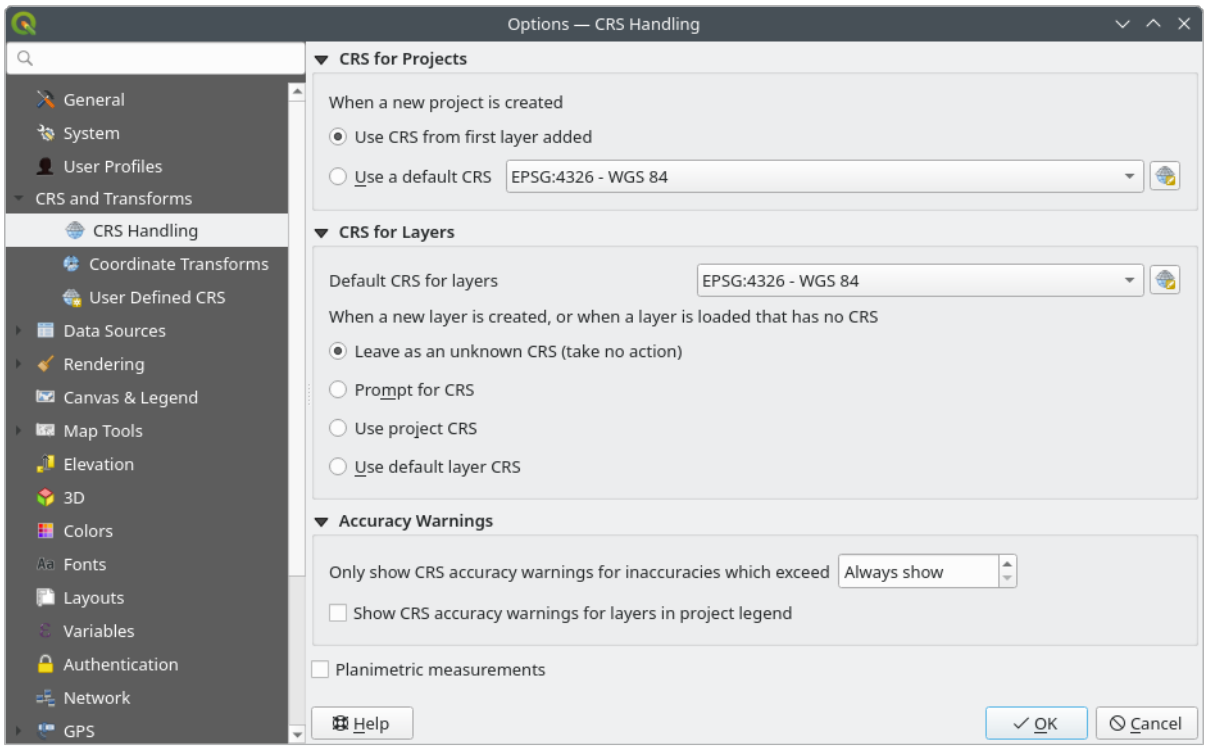


Figura 6.1: La pestaña CRS en el Diálogo de Opciones de QGIS

2. En el grupo *CRS para capas*, configure la acción a realizar *cuando se crea una nueva capa, o cuando se carga una capa que no tiene CRS*. Uno de:

   - *Dejar como CRS desconocido (no tomar medidas)*: no se solicitará que seleccione un CRS cuando se cargue una capa sin CRS, lo que aplazará la elección de CRS a un momento posterior. Conveniente al cargar muchas capas a la vez. Tales capas serán identificables en el panel *Capas* por el icono al lado de ellos. También estarán sin referencia, con las coordenadas de la capa tratadas como valores puramente numéricos, no terrestres, es decir, el mismo comportamiento que tienen todas las capas cuando *a project is set to have no CRS*.
   - *Solicitar CRS*: le pedirá que seleccione manualmente el CRS. Seleccionar la opción correcta es crucial, ya que una elección incorrecta colocará su capa en la posición incorrecta en la superficie de la Tierra. En ocasiones, los metadatos que se acompañan describirán el CRS correcto para una capa; en otros casos, deberá comunicarse con el autor original de los datos para determinar el CRS correcto que se utilizará.
   - *Usar CRS del proyecto*
   - *Use el CRS de capa predeterminado*, como se establece en *CRS predeterminado para el cuadro combinado de capas anterior*

Truco

Para asignar el mismo CRS a varias capas que no tienen crs o tienen una incorrecta en una operación:

1. Seleccione las capas en el panel *Layers*
2. Presiona `Ctrl+Shift+C`. También puede hacer clic derecho sobre una de las capas seleccionadas o ir a *Capa -> Establecer CRS de capa(s)*
3. Encuentre y selecciones el CRS correcto a usar

4. Y presione *OK*. Puede confirmar que se ha configurado correctamente en la pestaña *Fuente* del cuadro de diálogo de propiedades de las capas.

Tenga en cuenta que cambiar el CRS en esta configuración no altera la fuente de datos subyacente de ninguna manera, sino que simplemente cambia la forma en que QGIS interpreta las coordenadas sin formato de la capa en el proyecto QGIS actual.

6.3 Sistema de Coordenadas de Referencia del Proyecto

Cada proyecto en QGIS también tiene un Sistema de referencia de coordenadas asociado. El CRS del proyecto determina cómo se proyectan los datos desde coordenadas fundamentales en bruto al plano del mapa representado dentro de su lienzo del mapa en QGIS.

QGIS admite la transformación CRS «sobre la marcha» para datos ráster y vectoriales. Esto significa que, independientemente del CRS subyacente de capas de mapas particulares en su proyecto, siempre se transformarán automáticamente en el CRS común definido para su proyecto. Detrás de escena, QGIS reproyecta de forma transparente todas las capas contenidas en su proyecto en el CRS del proyecto, de modo que todas se representarán en la posición correcta entre sí.

Es importante hacer una elección adecuada de CRS para sus proyectos QGIS. Elegir un CRS inapropiado puede hacer que sus mapas se vean distorsionados y reflejen mal los tamaños relativos reales y las posiciones de las características. Por lo general, mientras se trabaja en áreas geográficas más pequeñas, habrá una serie de CRS estándar utilizados dentro de un país o área administrativa en particular. Es importante investigar qué CRS son opciones apropiadas o estándar para el área que está mapeando, y asegurarse de que su proyecto QGIS siga estos estándares.

Por defecto, QGIS comienza cada nuevo proyecto usando una proyección global predeterminada. Este CRS predeterminado es `EPSG:4326` (también conocido como «WGS 84»), y es un sistema de referencia global basado en latitud/longitud. Este CRS predeterminado puede ser cambiado vía configuración *CRS para Nuevos Proyexctos* en la pestaña *CRS* en *Configuración* ► 🔧 *Opciones...* (ver Figura 6.1). Hay una opción de ajustar automáticamente el CRS del proyecto para que coincida con el CRS de la primera capa cargada en el nuevo proyecto, o de manera alternativa puede seleccionar un CRS predeterminado a usar para todos los nuevos proyectos creados. Esta elección será guardada para usar en posteriores sesiones de QGIS.

El CRS del proyecto también se puede configurar a través de la pestaña: guilabel: *CRS* del cuadro de diálogo :menuselection:`Proyecto -> Propiedades … `. También se mostrará en la esquina inferior derecha de la barra de estado de QGIS.

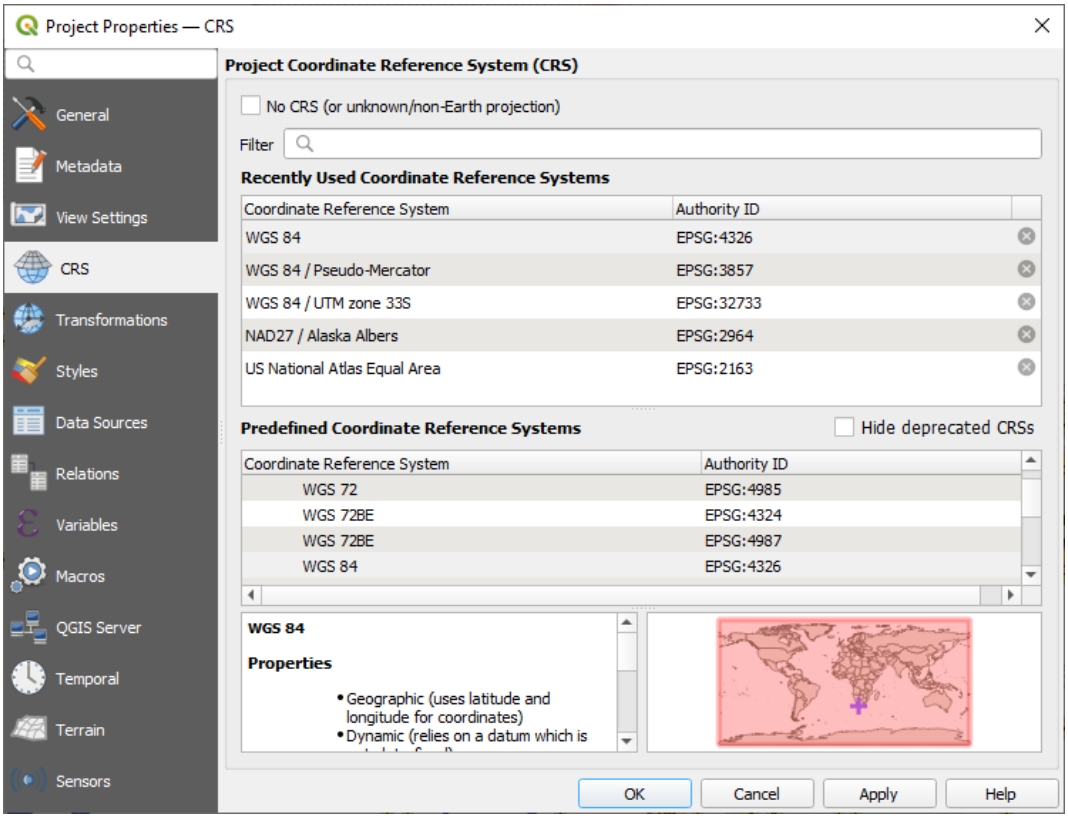


Figura 6.2: Diálogo de Propiedades del proyecto

Las opciones disponibles son:

- *No CRS (o proyección desconocida/no.Terrestre)*: Al marcar esta configuración, se deshabilitará TODO el manejo de proyección dentro del proyecto QGIS, lo que hará que todas las capas y coordenadas del mapa se traten como coordenadas cartesianas 2D simples, sin relación con las posiciones en la superficie de la Tierra. Se puede usar para adivinar un CRS de capa (basado en sus coordenadas sin procesar o cuando se usa QGIS para usos no terrestres como mapas de juegos de rol, mapas de edificios o cosas microscópicas. En este caso:
  - No se realiza ninguna reproyección mientras se procesan las capas: las entidades solo se dibujan utilizando sus coordenadas sin formato.
  - El elipsoide está bloqueado y forzado a `Ninguno / Planimétrico`.
  - Las unidades de distancia y área, y la visualización de coordenadas están bloqueadas y forzadas a «unidades desconocidas»; Todas las mediciones se realizan en unidades de mapa desconocidas y no es posible la conversión.
- o un sistema de referencia de coordenadas existente que puede ser *geográfico*, *proyectado* o *definido por el usuario*. Se muestra una vista previa de la extensión del SRC en la Tierra para ayudarle a seleccionar la adecuada. Las capas agregadas al proyecto se traducen sobre la marcha a este SRC para superponerlas independientemente de su SRC original. El uso de unidades y configuración de elipsoides está disponible y tiene sentido y puede realizar cálculos en consecuencia.

If you change your project's CRS and want the units (for distance and area) to match the units defined by the CRS, make sure you have selected *Map units* in the relevant settings, in the *General* tab of the *Project* ► *Properties…* dialog.

Truco

Estableciendo el CRS del proyecto de una capa

Puede asignar un CRS al proyecto usando el CRS de una capa:

1. En el panel *Capas*, haga click-derecho en la capa de la que quiera tomar el CRS
2. Seleccione *Establecer CRS del proyecto desde capa*.

El CRS del proyecto se redefine utilizando el CRS de la capa. La extensión del lienzo del mapa, la visualización de las coordenadas se actualizan en consecuencia y todas las capas del proyecto se traducen sobre la marcha al nuevo CRS del proyecto.

6.4 Selector del Sistema de Coordenadas de Referencia

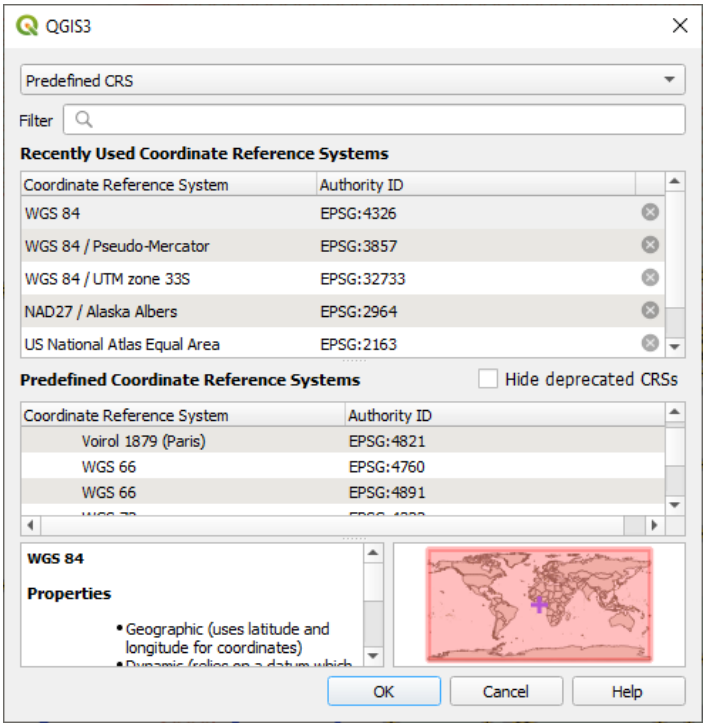


Figura 6.3: Selector SRC

Este cuadro de diálogo le ayuda a asignar un Sistema de referencia de coordenadas a un proyecto o una capa, siempre que haya un conjunto de bases de datos de proyección. Los elementos en el diálogo son:

- ** Filter**: si conoce el código EPSG, el identificador o el nombre de un Sistema de referencia de coordenadas, puede usar la función de búsqueda para encontrarlo. Ingrese el código EPSG, el identificador o el nombre.
- **Recently used coordinate reference systems**: Si tiene ciertos CRS que utiliza con frecuencia en su trabajo SIG diario, estos se mostrarán en esta lista. Haga click en uno de estos elementos para seleccionar el CRS asociado.
- **Coordinate reference systems of the world**: Esta es una lista de todos los CRS compatibles con QGIS, incluidos los sistemas de referencia de coordenadas geográficas, proyectadas y personalizadas. Para definir un CRS, selecciónelo de la lista expandiendo el nodo apropiado y seleccionando el CRS. El CRS activo está preseleccionado.
- **PROJ text**: Esta es la cadena CRS utilizada por el motor de proyección PROJ. Este texto es de solo lectura y se proporciona con fines informativos.

Si desea eliminar un SRC utilizado recientemente de la lista, puede hacerlo de varias maneras:

- Presione :sup:Botón *Eliminar del SRC usado recientemente* en el lado derecho.

- Presione `Supr` después de seleccionar SRC que desea eliminar.
- Haga clic derecho en SRC usado y elija *Eliminar SRC seleccionado de SRC usado recientemente* para eliminar SRC seleccionado o *Borrar todos los SRC usados recientemente* para eliminar todos los SRC usados.

El selector CRS también muestra una vista previa aproximada del área geográfica para la cual un CRS seleccionado es válido para su uso. Muchos CRS están diseñados solo para su uso en áreas geográficas pequeñas, y no debe usarlos fuera del área para la que fueron diseñados. El mapa de vista previa sombrea un área aproximada de uso cada vez que se selecciona un CRS de la lista. Además, este mapa de vista previa también muestra un indicador de la extensión actual del mapa del lienzo principal.

6.5 Sistema de referencia de coordenadas personalizada

Si QGIS no provee el sistema de referencia por coordenadas que necesita, puede definir un CRS predeterminado. Para definir un CRS, elegir *Custom CRS…* del menú *Ajustes*. CRSs personalizados se almacenan en su base de datos de usuario lqgl. A más del CRS personalizado, esta base de datos también contiene sus separadores espaciales y otros datos personalizados.

Definir un SRC personalizado en QGIS requiere una buena comprensión de la biblioteca de proyección PROJ. Para comenzar, consulte a «Cartographic Projection Procedures for the UNIX Environment - A User's Manual» por Gerald I. Evenden, U.S. Geological Survey Open-File Report 90-284, 1990 (disponible en https://pubs.usgs.gov/of/1990/of90-284/ofr90-284.pdf).

Este manual describe el uso de `proj` y las utilidades de línea de comandos relacionadas. Los parámetros cartográficos utilizados con `proj` se describen en el manual del usuario y son los mismos que los utilizados por QGIS.

El diálogo *Definición del Sistema de Referencia de Coordenadas personalizada* sólo necesita dos parámetros para definir un usuario de SRC:

1. Un nombre descriptivo
2. Los parámetros cartográficos en formato PROJ o WKT

Para crear un nuevo CRS:

1. Click en el botón Agregar nuevo CRS
2. Introduzca un nombre descriptivo
3. Seleccione el formato: puede ser *Proj String* o *WKT*
4. Agregar los *Parámetros* del CRS .

> **Nota**
>
> **Preferir almacenar la definición del CRS en formato WKT**
>
> Aunque ambos formatos `Proj String` y `WKT` son soportados, es altamente recomendado para almacenar las definiciones de proyecciones en formato WKT. Por consiguiente, si la definición disponible está en el formato proj, seleccione ese formato, ingrese los parámetros y luego cambie al formato WKT. QGIS convertirá la definición al formato WKT que luego podrá guardar.

5. Click *Validate* para probar si la definición de CRS es una definición de proyección aceptable.

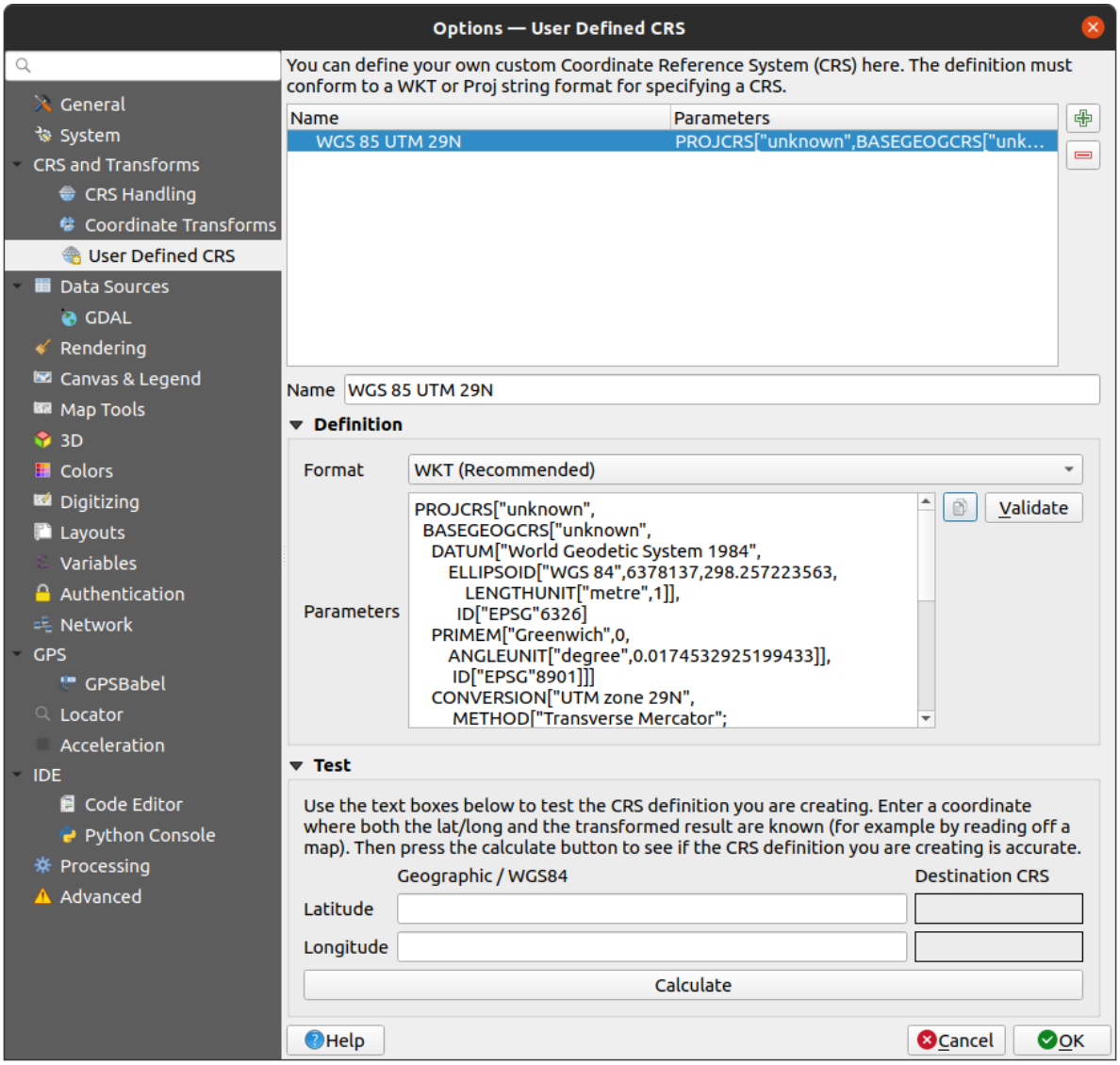


Figura 6.4: Diálogo SRC Personalizado

Puede probar sus parámetros de CRS para ver si dan buenos resultados. Para hacer esto, ingrese los valores conocidos de latitud y longitud de WGS 84 en *Norte* y *Este*, respectivamente. Haga clic en *Calcular* y compare los resultados con los valores conocidos en su sistema de referencia de coordenadas.

6.5.1 Integrar una transformación NTv2 en QGIS

Para integrar un archivo de transformación NTv2 en QGIS, necesita un paso más:

1. Coloque el archivo NTv2 (.gsb) en la carpeta CRS/Proj que usa QGIS (p. Ej. `C:\OSGeo4W64\share\proj` para usuarios de windows)
2. Agregue **nadgrids** (``+nadgrids=nameofthefile.gsb"") a la definición del Proj en el campo *Parametros* del *Definición de sistema de referencia de coordenadas personalizado* (:menuselection:`Ajustes - -> Proyecciones personalizadas … `).

Figura 6.5: Estableciendo una transformación NTv2

6.6 Transformaciones de Datum

En QGIS, la transformación CRS “sobre la marcha” está habilitada de forma predeterminada, lo que significa que cada vez que utiliza capas con diferentes sistemas de coordenadas, QGIS las reproyecta de forma transparente al proyecto CRS. ¡Para algunos CRS, hay una serie de posibles transformaciones disponibles para reproyectar al CRS del proyecto!

Por defecto, QGIS intentará usar la transformación más precisa disponible. Sin embargo, en algunos casos esto puede no ser posible, p. siempre que se requieran archivos de soporte adicionales para usar una transformación. Siempre que esté disponible una transformación más precisa, pero que no se pueda utilizar actualmente, QGIS mostrará un mensaje informativo de advertencia que le informará sobre la transformación más precisa y cómo habilitarla en su sistema. Por lo general, esto requiere la descarga de un paquete externo de archivos de soporte de transformación, y extraerlos a la carpeta `proj` en su carpeta QGIS *user profile*.

Si lo desea, QGIS también puede solicitarle que se realicen múltiples transformaciones posibles entre dos CRS, y le permite hacer una selección informada de cuál es la transformación más apropiada para usar con sus datos.

Esta personalización se hace en la pestaña del menú *Configuración* ► *Opciones* ► *Transformaciones* en el grupo *Transformaciones de datum predeterminadas*:

- usar *Preguntar por transformación de datum si hay varias disponibles*: cuando existe mas de una transformación de datum apropiada para una combinación de SRC fuente/destino, un cuadro de diálogo se abrirá automáticamente solicitando a los usuarios que elijan cuál de estas transformaciones de datum usarán para el proyecto. Si la casilla de verificación *Hacer Predeterminado* está marcada al seleccionar una transformación de este cuadro de diálogo, entonces la elección se recuerda y se aplica automáticamente a cualquier proyecto QGIS recién creado.
- o definir una lista de transformaciones de datos apropiadas para usar como valores predeterminados al cargar una capa en un proyecto o volver a proyectar una capa.

  Utilice el para abrir el cuadro de diálogo :guilabel:`Seleccionar transformaciones de datos”. Luego:

  1. Elija el *CRS Origen* de la capa, utilizando el menú desplegable o Seleccionar el widget CRS.
  2. Proporcione el *CRS Destino* de la misma manera.
  3. Se mostrará una lista de transformaciones disponibles de fuente a destino en la tabla. Clickando en una fila se muestran detalles de los ajustes aplicados, la precisión correspondiente y el área de uso de la transformación.

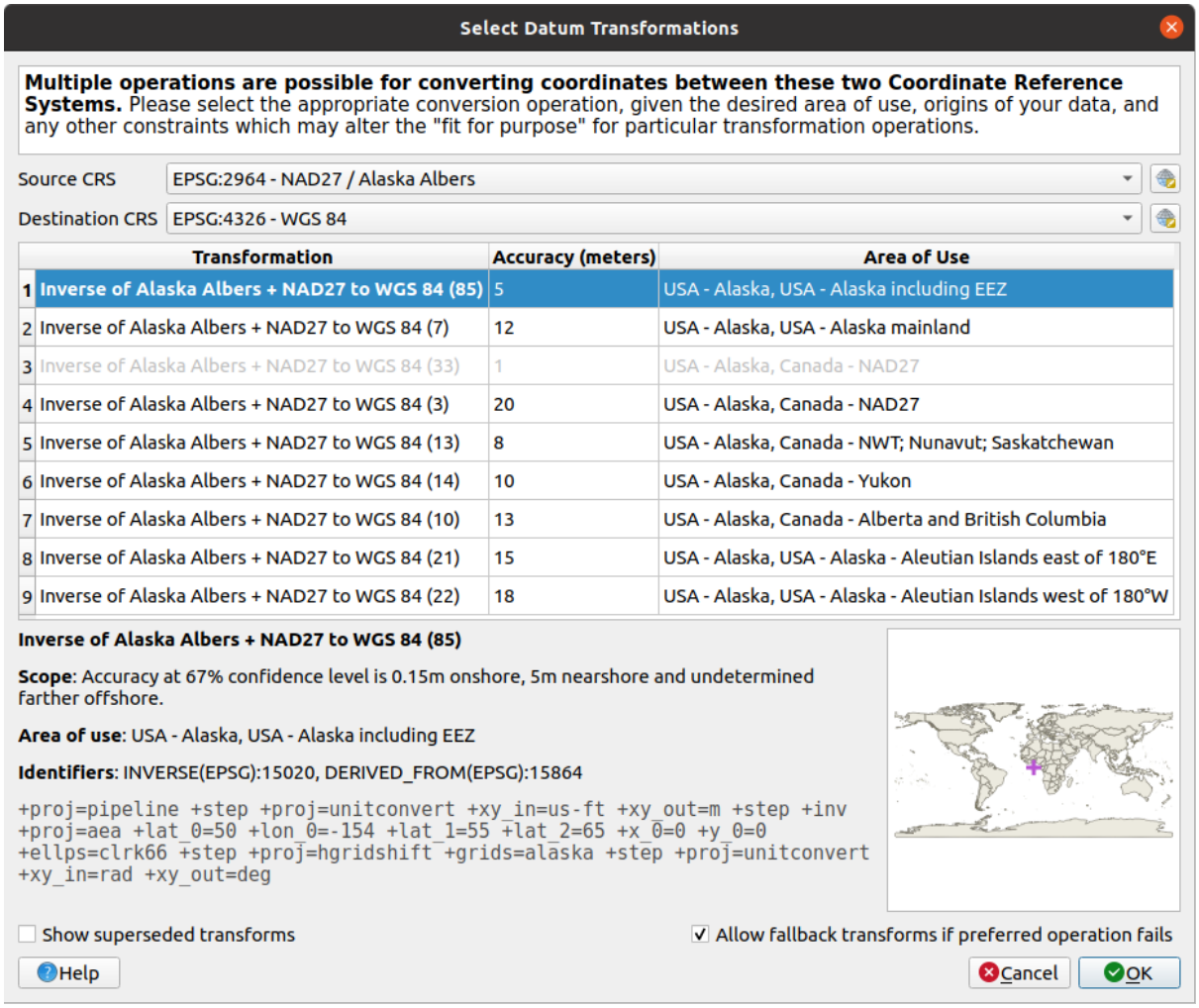


Figura 6.6: Seleccionar una transformación de referencia predeterminada preferida

En algunos casos, es posible que una transformación no esté disponible para su uso en su sistema. En este caso, la transformación seguirá apareciendo (atenuada) en esta lista, pero no podrá elegirse hasta que instale el paquete necesario de compatibilidad con la transformación. Por lo general, se proporciona un botón para descargar e instalar la cuadrícula correspondiente, que luego se almacena en la carpeta `proj` en el directorio activo *user profile*.

4. Encuentre su transformación preferida y seleccionela
5. Establecer si *Permitir transformaciones de respaldo si falla la operación preferida*
6. Haz clic en *Aceptar*.

   Se añade una fila a la tabla en *Transformaciones Predeterminadas de Datum* con información sobre el *SRC Fuente*, el *SRC Destino*, la *Operación* aplicada para la transformación y si *Permitir Transformaciones de Respaldo* está activo.

A partir de ahora, QGIS utiliza automáticamente las transformaciones de referencia seleccionadas para una mayor transformación entre estos dos CRS hasta que lo elimine ( ) de la lista o cambie la entrada ( ) en la lista.

Las Transformaciones de Datum establecidas en la pestaña *Configuración* ► *Opciones* ► *Transformaciones* seran inherentes para todos los nuevos proyectos creados de QGIS en el sistema. Adicionalmente, un proyecto particular debe tener sus propios conjuntos específicos de transformaciones especificados mediante la pestaña *SRC* del cuadro de diálogo *propiedades de Proyecto* (*Proyecto* ► *Propiedades…*). Estos ajustes se aplican solo al proyecto actual.

CAPÍTULO 7

Visualización de mapas

La creación de mapas es uno de los «fines de negocio» de QGIS. Las vistas de mapas son lienzos para visualizar las capas espaciales cargadas en el proyecto, con las distintas configuraciones de renderizado aplicadas a las mismas. Pueden ser de tipo *2D* o *3D* o trazada a lo largo de una *línea de perfil*, mostrar diferente escala o extensión, o mostrar un conjunto diferente de las capas cargadas gracias a *temas de mapa*.

7.1 Vista de mapa 2D

La vista de mapa 2D (también llamada **Lienzo de mapa**) es el lugar central donde se muestran los mapas. QGIS se abre por defecto con una sola vista de mapa (llamada `mapa principal`), que muestra las capas en 2D y está estrechamente vinculada al panel *Layers*. En la ventana se muestra la representación (simbología, etiquetado, visibilidades…) que se aplicó a las capas cargadas.

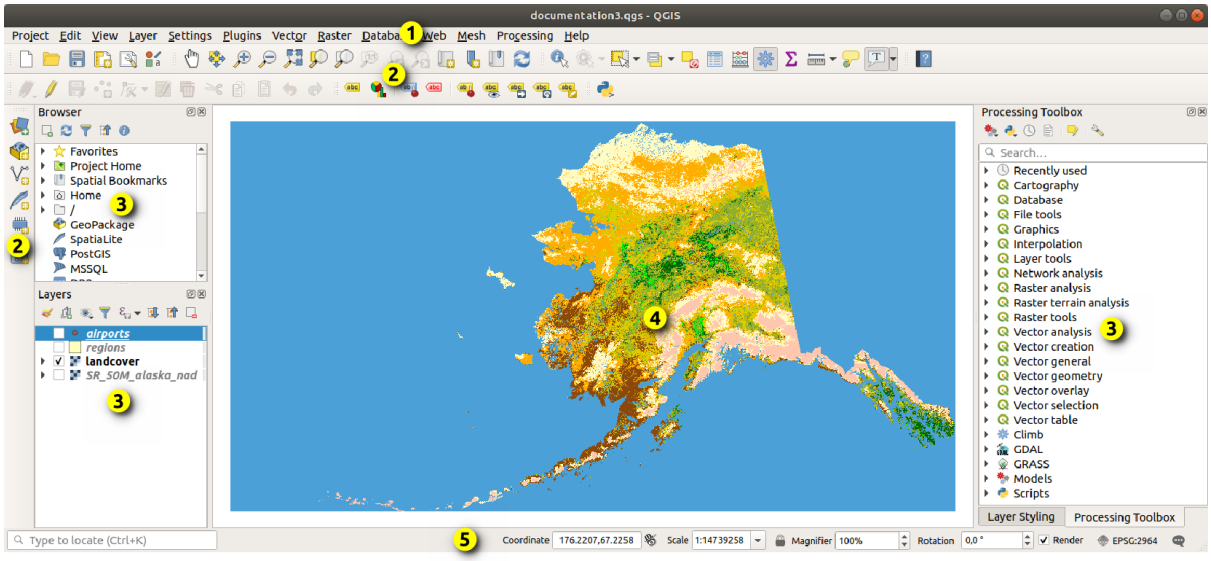


Figura 7.1: IGU QGIS

7.1.1 Explorando la vista de mapa

Cuando añade una capa (ver p.ej. *Abriendo Datos*), QGIS busca automáticamente su CRS. Si se establece un CRS diferente de forma predeterminada para el proyecto (consulte *Sistema de Coordenadas de Referencia del Proyecto*) entonces la extensión de la capa es trasladada «on-the-fly» a ese CRS, y la vista de mapa es ampliada a aquella extensión si empezó con un proyecto QGIS en blanco. Si ya hay capas en el proyecto, no se realiza el cambio de tamaño del lienzo del mapa, por lo que solo se verán las entidades que se encuentren dentro de la extensión del lienzo del mapa actual.

Haga clic en la vista del mapa y podrá interactuar con él, desplazándose o haciendo zoom a diferentes zonas del mapa. Existen herramientas específicas en la *Barra de herramientas de navegación* y en el menú *Vista*, con accesos directos desde el teclado o los botones del ratón.

Tabla 7.1: Herramientas de navegación en el lienzo del mapa

Herramienta	Uso
Tomar panorámica del mapa	• Un solo clic izquierdo: el mapa se centra en el punto en el que se hizo clic, a la misma escala • Mantenga presionado el botón izquierdo del mouse y arrastre el lienzo del mapa.
Acercar	• Un solo clic con el botón izquierdo: el mapa se centra en el punto seleccionado y la escala se duplica. • Arrastre un rectángulo sobre el lienzo del mapa con el botón izquierdo del ratón para ampliar una zona. • Mantenga pulsada la tecla `Alt` para cambiar a la herramienta Alejar.
Alejar	• Un solo clic con el botón izquierdo del ratón: el mapa se centra en el punto seleccionado y la escala se reduce a la mitad. • Arrastre un rectángulo sobre el lienzo del mapa con el botón izquierdo del ratón para alejar el zoom de una zona. • Mantenga pulsada la tecla `Alt` para cambiar a la herramienta Acercar.
Pan Mapa a Selección	Desplace el mapa a las características seleccionadas de todas las capas seleccionadas en el panel *Layers*.
Acercar a la selección	Zoom a las características seleccionadas de todas las capas seleccionadas en el panel *Layers*. *También disponible en el menú contextual de la capa*.
Zoom A Capa(s)	Zoom a la extensión de todas las capas seleccionadas en el panel *Layers*. *También disponible en el menú contextual de la capa*.
Zoom Completo	Zoom a la extensión de todas las capas en el proyecto o al *project full extent 1*.
Último zoom	Zoom en el mapa a la extensión previa en el historial.
Zoom Siguiente	Zoom en el mapa a la siguiente extensión en el historial.
Zoom a Resolución Nativa	Zoom en el mapa a un nivel donde un pixel de la capa ráster activa ocupa un pixel de la pantalla. *También disponible en el menú contextual de la capa*.

continúe en la próxima página

Tabla 7.1 – proviene de la página anterior

Herramienta	Uso
Rueda del ratón	• Desplazar el mapa: Mantén pulsada la rueda del ratón y arrástrala. • Zoom: Gira la rueda del ratón para acercar o alejar la imagen. Si se pulsa la tecla `Ctrl` mientras se gira la rueda del ratón, el zoom será más fino. • Pulse el botón atrás o adelante para navegar por el historial de zoom del lienzo del mapa.
Teclado	• Desplaza el mapa: Mantén pulsada la tecla `Espacio` y mueve el ratón. Pulsa las flechas para desplazarte hacia arriba, abajo, izquierda y derecha. • Acercar: Pulse `Subir página` o `Ctrl++`. • Alejar: Pulse `Bajar página` o `Ctrl+-`. • Acercar a una zona: Cuando ciertas herramientas de mapa están activas (Identificar, Medir…), mantenga pulsado `Mayús` y arrastre un rectángulo en el mapa para hacer zoom a esa zona. No es compatible con las herramientas de selección o edición activas.

Haga clic derecho sobre el mapa y debería poder  *Copy coordinates* del punto pinchado en el SRC del mapa, en WGS84 o en un SRC personalizado. La información copiada puede luego pegarse en una expresión, en líneas de código, un editor de texto o una hoja de cálculo…

7.1.2 Control de la representación de mapas

De forma predeterminada, QGIS representa todas las capas visibles cada vez que se actualiza el lienzo del mapa. Los eventos que desencadenan una actualización del lienzo del mapa incluyen:

- cambiar la visibilidad de una capa
- modificar la simbología de una capa visible
- añadiendo una capa
- enfocando o ampliando
- redimensionando la ventana QGIS

QGIS le permite controlar el proceso de representación de varias maneras.

- a nivel *global*
- por capa, utilizando, por ejemplo, la función *renderización dependiente de la escala*
- o con herramientas específicas de la interfaz gráfica de usuario.

Para detener el dibujo del mapa, pulsa la tecla `Esc`. Esto detendrá la actualización del lienzo del mapa y dejará el mapa parcialmente dibujado. Sin embargo, puede tardar un poco después de pulsar `Esc` para que el dibujo del mapa se detenga.

Para suspender la renderización, haga clic en la casilla  *Renderizar* en la esquina inferior derecha de la barra de estado. Cuando  *Renderizar* está desmarcada, QGIS no redibuja el lienzo en respuesta a ninguno de los disparadores habituales mencionados anteriormente. Ejemplos de cuando puede querer suspender el renderizado incluyen:

- añadiendo muchas capas y simbolizándolas antes del dibujo
- añadiendo una o mas capas grandes y establecer la dependencia de escala antes de dibujar
- Agregar una o más capas grandes y hacer zoom a una vista específica antes de dibujar
- Cualquier combinación de las anteriores

Marcar la casilla *Renderizar* habilita el renderizado y origina un refresco inmediato del lienzo del mapa.

7.1.3 Control basado en el tiempo en el lienzo del mapa

QGIS puede manejar el control temporal en las capas cargadas, es decir, modificar la representación del lienzo del mapa en función de una variación de tiempo. Para lograr esto, necesitas:

1. Capas que tienen establecidas propiedades temporales dinámicas. QGIS admite control temporal para diferentes proveedores de datos, con configuraciones personalizadas. Se trata principalmente de establecer el rango de tiempo en el que se mostrará la capa:
   - *raster layers*: controla si mostrar o no la capa.
   - *Capas WMTS*: controla si los datos deben representarse basándose en un intervalo temporal estático o siguiendo un intervalo temporal dinámico.
   - *vector layers*: las características se filtran en función de los valores de tiempo asociados a sus atributos
   - *mesh layers*: muestra dinámicamente los valores de los grupos de conjuntos de datos activos

   Cuando las opciones temporales dinámicas están habilitadas para una capa, un |indicadorTemporal| El icono se muestra junto a la capa *Layers* para recordarle que la capa está controlada temporalmente. Haga clic en el icono para actualizar la configuración temporal.
2. Habilite la navegación temporal del lienzo del mapa usando el:ref:*Temporal controller panel <temporal_controller>*. El panel se activa:
   - usando el |temporal Temporal controller panel icon located in the :guilabel:`Barra de herramientas de navegación del mapa
   - desde :menuselection:`Ver –> Paneles –> Menú del panel del controlador temporal
   - o desde el menú :menuselection:`Ver –> Filtrado de datos –> Panel del controlador temporal

El panel del controlador temporal

The :guilabel:`El panel del controlador temporal tiene los siguientes modos:

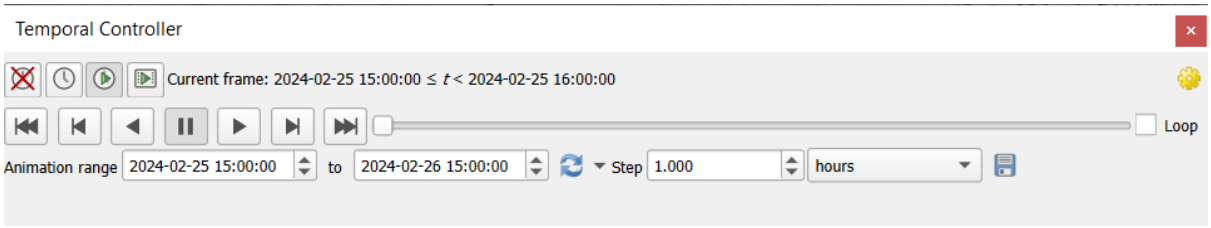


Figura 7.2: Panel de control temporal en modo de navegación

- Desactivar la navegación temporal: todas las configuraciones temporales están deshabilitadas y las capas visibles se representan como de costumbre
- Fixed range temporal navigation: se establece un rango de tiempo y solo se muestran en el mapa las capas (o entidades) cuyo rango temporal se superpone con este rango.
- Animated temporal navigation: se establece un intervalo de tiempo, se divide en pasos y solo se muestran en el mapa las capas (o entidades) cuyo intervalo temporal se superpone con cada cuadro
- Película animada: se establece un número total de fotogramas para la animación, con la configuración del mapa avanzando en cada paso, mostrando la progresión fotograma a fotograma sin ningún filtro de datos basado en el tiempo.

- Settings para el control general de la animación
  - *Frames rate*: número de pasos que se muestran por segundo
  - *Rango acumulativo*: todos los fotogramas de la animación tendrán la misma fecha-hora de inicio pero diferentes fechas y horas de finalización. Esto es útil si desea acumular datos en su visualización temporal en lugar de mostrar una «ventana de tiempo en movimiento» a través de sus datos.

Animando una navegación temporal

Una animación se basa en un conjunto variable de capas visibles en momentos determinados dentro de un intervalo de tiempo. Para crear una animación temporal:

1. Toggle on the Animated temporal navigation, mostrando el widget del reproductor de animación
2. Introduzca el *Rango de tiempo* a considerar. Usando el botón , esto puede ser definido como:
   - *Set to full range* de todas las capas habilitadas en el tiempo
   - *Establecido en el rango de proyecto preestablecido como se define en el :ref:`project properties <project_temporal>*
   - :guilabel:`Establecido en el rango de una sola capa tomado de una capa habilitada para el tiempo
3. Llenar in el tiempo *Step* para separar el rango de tiempo. Se admiten diferentes unidades, desde `segundos` hasta `siglos`. También está disponible como paso una opción de `marcas de tiempo de origen`: cuando se selecciona, hace que la navegación temporal avance entre todos los rangos de tiempo disponibles de las capas en el proyecto. Es útil cuando un proyecto contiene capas con horas disponibles no contiguas, como un servicio WMS-T que proporciona imágenes que están disponibles en fechas irregulares. Esta opción le permitirá pasar solo entre rangos de tiempo donde se muestra la siguiente imagen disponible.
4. Click al botón para obtener una vista previa de la animación. QGIS generará escenas utilizando la representación de capas en los tiempos establecidos. La visualización de las capas depende de si se superponen a un marco de tiempo individual.

   La animación también puede previsualizarse moviendo el deslizador de tiempo. Si se marca la casilla de verificación *Bucle* se ejecutará repetidamente la animación, mientras que si se pulsa se detendrá una animación en ejecución. Hay disponible un conjunto completo de botones de reproducción de vídeo.

   El desplazamiento horizontal con la rueda del mouse (cuando sea compatible) con el cursor en el lienzo del mapa también le permitirá navegar, o «barrer», el control deslizante de navegación temporal hacia adelante y hacia atrás.
5. Click en el botón Export animation si desea generar una serie de imágenes que representan la escena. Se pueden combinar más tarde en un software de edición de video:

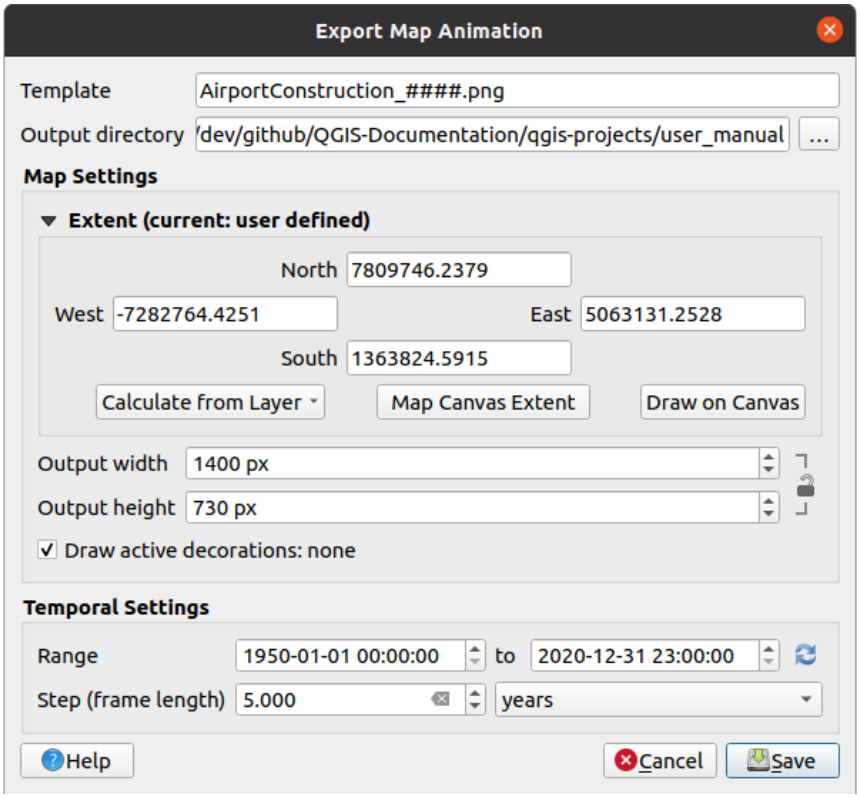


Figura 7.3: Exportación de escenas de animación de lienzo de mapa a imágenes

- El nombre del archivo:guilabel:*Template*: los #### se reemplazan con el número de secuencia del cuadro
- El *Directorio de salida*
- En *Configuración del mapa*, puedes:
  - redefinir el *spatial extent* para usar
  - control the *Resolución* de la imagen (*Ancho de salida* y *Altura de salida*)
  - *Dibujar decoraciones activas*: si está activa:ref:*decorations* <*decorations*> debe mantenerse en la salida
- En *Configuración temporal*, puede redefinir:
  - El tiempo *Range* para la animación
  - El *Paso (longitud del marco)* en la unidad de su elección

7.1.4 El panel del controlador de elevación

Controlador de elevación permite manejar datos de rango z de elevación en mapas 2D. Esta opción admite actualmente capas de nubes de puntos y MDE ráster. Utilice el panel para visualizar e interactuar con los datos de elevación en el lienzo del mapa.

Active el panel *Controlador de elevación* en el lienzo del mapa haciendo clic en *Filtrado de datos* en el menú *Ver*. La **|elevación|** *Controlador de elevación* aparece como un deslizador de rango en la parte izquierda del lienzo del mapa. En la parte superior del deslizador, hay un menú **|configuración|** Configuración en el que puede:

- Establezca el valor *Tamaño de rango fijo* para bloquear el rango de elevación a un tamaño específico. Esta configuración se guarda en el proyecto y se mantiene en todas las sesiones, garantizando la coherencia de su filtro de elevación. Utilice la opción *Invertir dirección* para invertir el deslizador del filtro de elevación, de forma que se mueva desde los valores altos en la parte inferior hasta los valores bajos en la parte superior.

- *Establecer rango de elevación* para abrir la pestaña de elevación del proyecto, donde puede establecer los límites inferior y superior dedicados para el rango de elevación.
- *Desactivar filtro de elevación* para eliminar el filtro de elevación del lienzo del mapa

7.1.5 Marcar extensiones en el mapa

Los marcadores espaciales le permiten «marcar» una ubicación geográfica y volver a ella más tarde. De forma predeterminada, los marcadores se guardan en el perfil del usuario (como *Marcadores de usuario*), lo que significa que están disponibles en cualquier proyecto que el usuario abra. También se pueden guardar para un solo proyecto (llamado *Marcadores del proyecto*) y almacenarse dentro del archivo del proyecto, lo que puede ser útil si el proyecto se va a compartir con otros usuarios.

Crear un marcador

Para crear un marcador:

1. Ampliar y desplazarse al área de interés.
2. Seleccione la opción de menú:menuselection:*Ver* -> *Nuevo Marcador Espacial…*, presione `Ctrl + B` o haga click derecho en la entrada *Marcadores Espaciales* en el panel *Navegador* y seleccione *Nuevo Marcador Espacial*. Se abre el cuadro de diálogo *Editor de marcadores*.

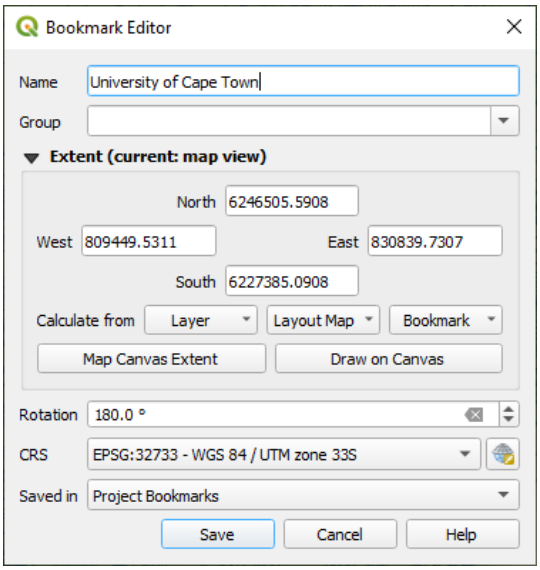


Figura 7.4: El diálogo de editor de Marcadores

3. Introduzca un nombre descriptivo para el marcador
4. Introduzca o seleccione un nombre de grupo en el cuál almacenar los marcadores relacionados
5. Seleccione la extensión del área que desea guardar, utilizando el *extent selector* widget
6. Cambiar el mapa *Rotación*
7. Indique el *CRS* a usar para la extensión
8. Seleccione si el marcador será *Guardado en Marcadores del usuario* o *Marcadores del proyecto* (de forma predeterminada, esta lista desplegable está establecida en *Marcadores del usuario*)
9. Presione *Guardar* para añadir el marcador a la lista

Tenga en cuenta que puede tener múltiples marcadores con el mismo nombre.

Trabajar con marcadores

To use and manage bookmarks, you can either use the *Spatial Bookmarks* panel or *Browser*:

- Select *View* ► *Show Spatial Bookmark Manager* or press `Ctrl+7` to open the *Spatial Bookmark Manager* panel.
- Select *View* ► *Show Spatial Bookmarks* or press `Ctrl+Shift+B` to show the *Spatial Bookmarks* entry in the *Browser* panel.

Puede realizar las siguientes tareas:

Tabla 7.2: Gestionar las acciones de los marcadores

Tarea	Administrador de Marcador Espacial	Navegador
Zoom a Marcador	Doble-click en él, o seleccione el marcador y presione el botón *Zoom a marcador*.	Doble-click en él, arrástrelo y suéltelo en el lienzo del mapa, o haga click con el botón derecho en el marcador y seleccione *Zoom a Marcador*.
Borrar a Marcador	Seleccione el marcador y haga click en el botón *Eliminar marcador*. Confirma tu elección.	Haga click derecho en el marcador y seleccione *Eliminar marcador espacial*. Confirma tu elección.
Exportar marcadores a XML	Haga click en el botón **|compartir|** *Importar/Exportar Marcadores* y seleccione *Exportar*. Todos los marcadores (usuario o proyecto) se guardan en un archivo xml.	Seleccione una o más carpetas (usuario o proyecto) o subcarpetas (grupos), luego haga click derecho y seleccione *Exportar marcadores espaciales* …. El subconjunto de marcadores seleccionado se guarda.
Importar marcadores de XML	Haga click en el botón **|compartir|** *Importar/Exportar marcadores* y seleccione *Importar*. Todos los marcadores en el archivo XML se importan como marcadores de usuario.	Haga click con el botón derecho en la entrada *Marcadores Espaciales* o en una de sus carpetas (usuario o proyecto) o subcarpetas (grupos) para determinar dónde importar los marcadores, luego seleccione *Importar marcadores espaciales*. Si se realiza en la entrada *Marcadores espaciales*, los marcadores se agregan a :guilabel:` Marcadores de usuario`.
Editar Marcador	Puede cambiar un marcador cambiando los valores en la tabla. Puede editar el nombre, el grupo, la extensión y si está almacenado en el proyecto o no.	Haga click con el botón derecho en el marcador deseado y seleccione *Editar marcador espacial* …. Se abrirá *Editor de Marcadores*, permitiéndole redefinir cada aspecto del marcador como si lo estuviera creando por primera vez. También puede arrastrar y soltar el marcador entre carpetas (usuario y proyecto) y subcarpetas (grupos).

Puede gestionar las acciones de los marcadores haciendo clic con el botón derecho en el marcador deseado en el *Administrador de marcadores espaciales*. También puede hacer zoom hasta los marcadores escribiendo el nombre del marcador en el *localizador*.

7.1.6 Decorar el mapa

Las decoraciones incluyen Cuadrícula, Etiqueta de título, Etiqueta de copyright, Imagen, Flecha norte, Barra de escala y Extensiones de diseño. Se utilizan para “decorar” el mapa agregando elementos cartográficos.

Cuadrícula

Cuadrícula le permite agregar una malla de coordenadas y anotaciones de coordenadas al lienzo del mapa.

1. Seleccione la opción menú *Ver* ► *Decoraciones* ► *Cuadrícula…* para abrir el diálogo.

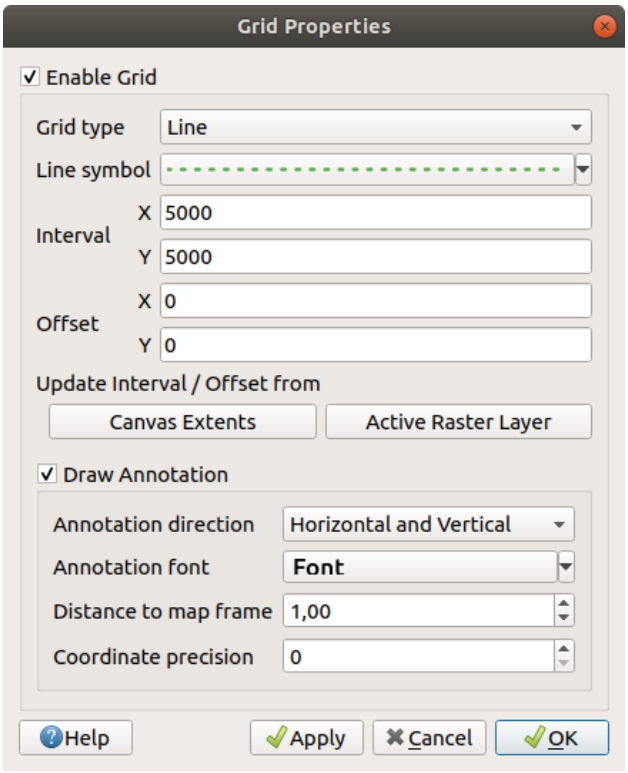


Figura 7.5: El Diálogo Cuadrícula

2. Marque *Activar cuadrícula* y establecer definiciones de cuadrícula de acuerdo con las capas cargadas en el lienzo del mapa:
   - El *tipo de cuadrícula*: puede ser *Linea* o *Marcador*
   - El asociado *Line symbol* o *marker symbol* usado para representar las marcas de la cuadrícula
   - El *Intervalo X* y :guilabel:` Intervalo Y` entre las marcas de la cuadrícula, en unidades de mapa
   - Una distancia *Offset X* y *Offset Y* de las marcas de cuadrícula desde la esquina inferior izquierda del lienzo del mapa, en unidades de mapa
   - Los parámetros intervalo y offset pueden ser configurados en función de:
     - *Extensión del Lienzo*: genera una cuadrícula con un intervalo que es aproximadamente 1/5 del ancho del lienzo
     - Resolución de *Capa Ráster Activa*
3. Marque *Dibujar anotaciones* para mostrar las coordenadas de las marcas de la cuadrícula y establecer:
   - La *Dirección de Anotación*, p.ej. como las etiquetas se colocaría en relación con su línea de cuadrícula. Puede ser:

  - *Horizontal* o *Vertical* para todas las etiquetas
  - *Horizontal y Vertical*, es decir, cada etiqueta es paralela a la marca de cuadrícula a la que se refiere
  - *Dirección del límite*, es decir, cada etiqueta sigue el límite del lienzo y es perpendicular a la marca de cuadrícula a la que se refiere
- La *Fuente de Anotación* (formateado de texto, buffer, sombra…) usando el *control de selección de fuente*
- La *Distancia al marco del mapa*, margen entre anotaciones y límites del lienzo del mapa. Conveniente cuando *exportar el lienzo del mapa* p.ej. a un formato de imagen o PDF, y evitar anotaciones en los límites del «papel».
- La *Precisión de Coordinada*

4. Click en *Aplicar* para verificar que se ve como se esperaba u *OK* si está satisfecho.

Etiqueta de título

La :guilabel:"Etiqueta de título" le permite decorar su mapa con un **Título**.

Para añadir una decoración con Etiqueta de Título:

1. Seleccione la opción de menú *Ver* ► *Decoraciones* ► *Etiqueta de Título…* para abrir el diálogo.

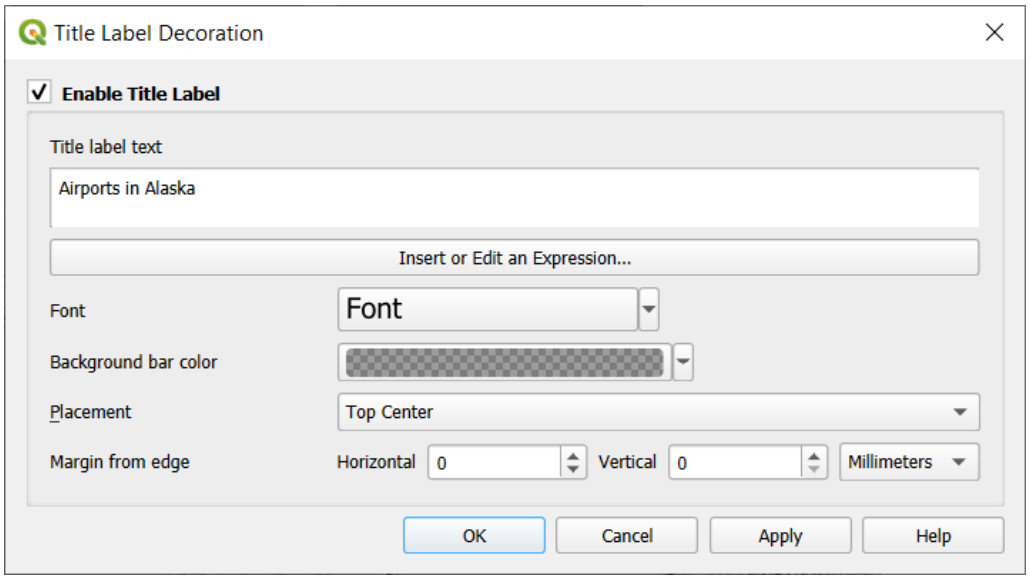


Figura 7.6: El Diálogo de Título de Decoración

2. Asegúrese de que *Habilitar Etiqueta de Título* está marcada
3. Ingrese el texto del título que desea colocar en el mapa. Puede hacerlo dinámico utilizando el botón *Insertar o Editar una Expresión…*.
4. Elija el:guilabel:*Fuente* para la etiqueta usando el *font selector widget* con acceso completo a las opciones de QGIS *text formatting*. Establezca rápidamente el color y la opacidad de la fuente haciendo click en la flecha negra a la derecha del cuadro combinado de fuentes.
5. Seleccione el *color* a aplicar al título *Color de la barra de fondo*.
6. Elija *Colocación* de la etiqueta en el lienzo: las opciones son *Arriba a la izquierda*, *Centro superior`(predeterminado), :guilabel:* Arriba a la derecha`, *Abajo a la izquierda*, :guilabel:` Centro inferior` y *Abajo a la derecha*.

7. Refine la ubicación del elemento estableciendo un horizontal y/o vertical *Margen from Edge*. Estos valores pueden estar en **Milímetros** ó **Píxeles** ó configurarse como un ** Porcentaje ** del ancho o alto del lienzo del mapa.
8. Click en *Aplicar* para verificar que se ve como se esperaba u *OK* si está satisfecho.

Etiqueta de Derechos de Copia

Etiqueta de Derechos de Copia se puede usar para ornamentar su mapa con una etiqueta **Derechos de Copia**.

Para añadir este ornamento:

1. Seleccione la opción de menú *Ver* ► *Decoraciones* ► *Etiqueta Derechos de Copia…* para abrir el diálogo.

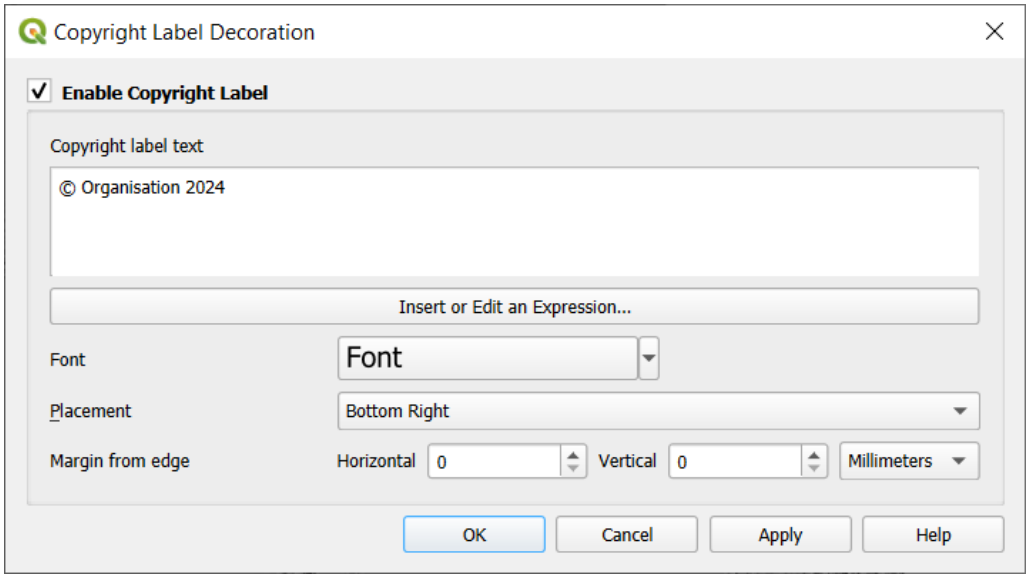


Figura 7.7: El Diálogo de Ornamentación de Derechos de Copia

2. Asegúrese de que *Habilitar Etiqueta de Derechos de Copia* esté marcada
3. Ingrese el texto de derechos de autor que desea colocar en el mapa. Puedes dinamizarlo usando el botón *Insert or Edit an Expression…*.
4. Elija el:guilabel:*Fuente* para la etiqueta usando el *font selector widget* con acceso completo a las opciones de QGIS *text formatting*. Establezca rápidamente el color y la opacidad de la fuente haciendo click en la flecha negra a la derecha del cuadro combinado de fuentes.
5. Elija *Colocación* de la etiqueta en el lienzo: las opciones son *Arriba a la izquierda*, *Arriba al centro*, *Arriba a la derecha*, *Abajo a la izquierda*, *Centro Abajo*, y *Derecha Abajo* (predeterminado para la ornamentación de Derechos de Copia)
6. Refine la ubicación del elemento estableciendo un horizontal y/o vertical *Margen from Edge*. Estos valores pueden estar en **Milímetros** ó **Píxeles** ó configurarse como un ** Porcentaje ** del ancho o alto del lienzo del mapa.
7. Click en *Aplicar* para verificar que se ve como se esperaba u *OK* si está satisfecho.

Imagen Ornamental

Image le permite agregar una imagen (logotipo, leyenda, ..) en el lienzo del mapa.

Para añadir una imagen:

1. Seleccione la opción del menú *Ver* ► *Decoraciones* ► *Imagen…* para abrir el diálogo.

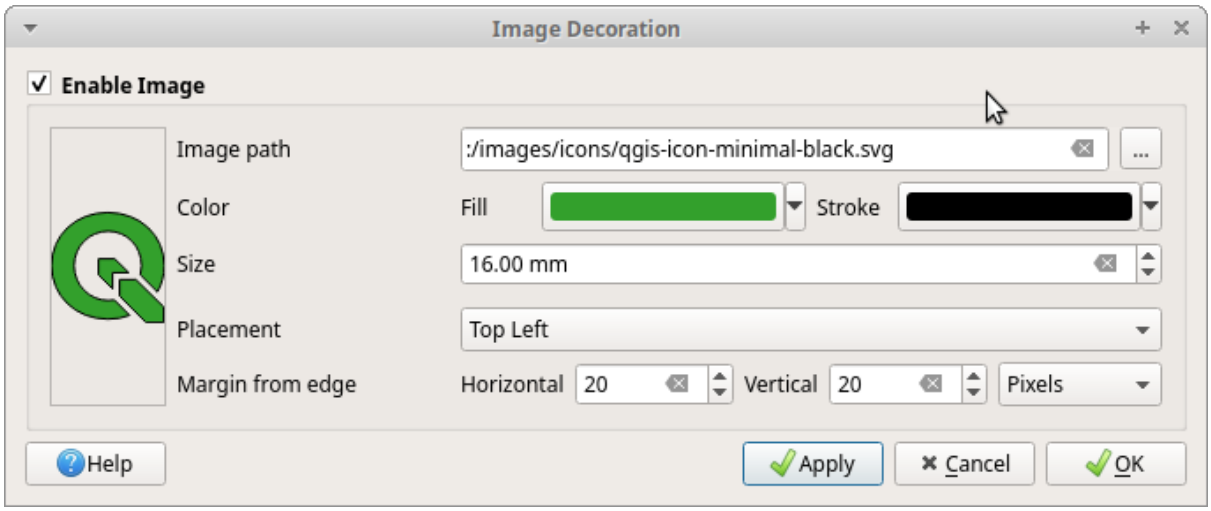


Figura 7.8: El Diálogo de Imagen Ornamental

2. Asegúrese de que *Habilitar Imagen* esté marcado
3. Seleccione una imagen de mapa de bits (e.g. png ó jpg) ó imagen SVG usando el botón … Navegar
4. Si ha elegido un parámetro SVG habilitado, también puede establecer un color *Relleno* o *Trazo* (contorno). Para las imágenes de mapa de bits, la configuración de color está deshabilitada.
5. Establezca un *Size* de la imagen en mm. La anchura de la imagen seleccionada es usada para redimensionarla al *Size* dado.
6. Elija donde quiere ubicar la imagen en el lienzo del mapa con el cuadro combinado *Ubicación*. La posición por defecto es *Top Left*.
7. Establezca *Horizontal* y :guilabel:` Margen vertical desde el borde (lienzo )`. Estos valores se pueden establecer en **Milímetros**, **Píxeles** o como **Porcentaje** del ancho o alto del lienzo del mapa.
8. Click en *Aplicar* para verificar que se acerca a lo esperado y pulse *OK* si está satisfecho.

Flecha del Norte

Flecha Norte le permite añadir una flecha de norte en el lienzo del mapa.

Para añadir una flecha norte:

1. Seleccione la opción de menú *Ver* ► *Decoraciones* ► *Flecah Norte…* para abrir el diálogo.

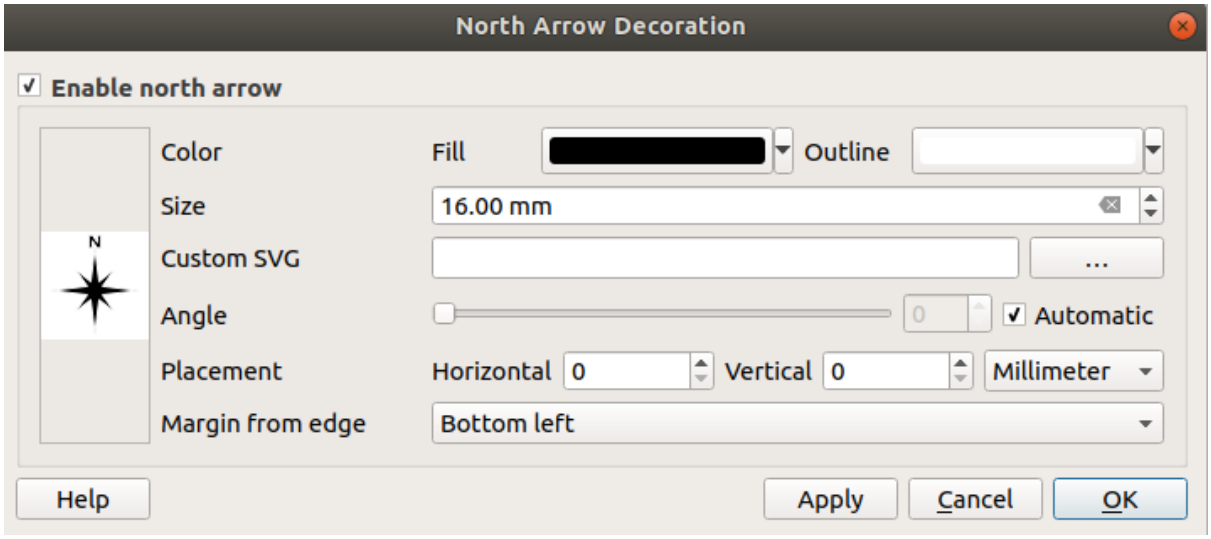


Figura 7.9: El Diálogo Flecha Norte

2. Asegúrese de que *Habilitar flecha norte* está marcada
3. Opcionalmente cambie el color y tamaño, o escoja un SVG personalizado
4. Opcionalmente cambie el ángulo o elija **Automático** para dejar que QGIS determine la dirección
5. Opcionalmente elija la ubicación desde el cuadro combinado de Emplazamiento
6. Opcionalmente refina el emplazamiento de la flecha estableciendo un `Margen desde el borde (lienzo)`horizontal y/o vertical. Estos valores pueden ser en **Milimetros** p **Pixels** o establecidos como un **Porcentaje** de la anchura o altura del lienzo del mapa.
7. Click en *Aplicar* para verificar que se acerca a lo esperado y pulse *OK* si está satisfecho.

Barra de escala

Barra de escala añade una simple barra de escala al lienzo del mapa. Puede controlar el estilo y la ubicación, así como el etiquetado de la barra. La barra de escala respeta la *unidad de distancia del proyecto* activo como se define en *Propiedades del proyecto* ► *General* ► *Unidades para medir distancias*.

Para añadir una barra de escala:

1. Seleccione la opción de menú *Ver* ► *Decoraciones* ► *Barra de Escala…* para abrir el diálogo

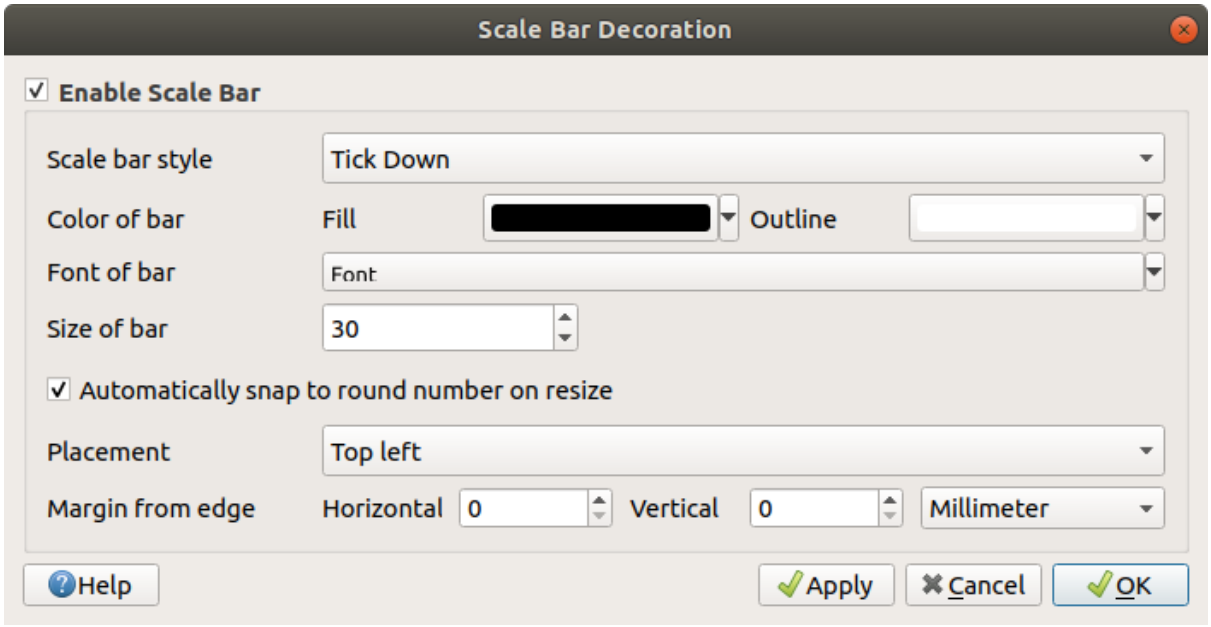


Figura 7.10: El Diálogo Barra de Escala

2. Asegúrese de que *Habilitar Barra de Escala* está marcada
3. Elija un estilo desde el cuadro combinado *Estilo de Barra de Escala*
4. Seleccione el *Color de la barra* eligiendo un color de relleno (predeterminado: negro) y un color de contorno (predeterminado: blanco). El relleno y el contorno de la barra de escala pueden hacerse opacos haciendo click en la flecha hacia abajo a la derecha de la entrada de color.
5. Seleccione la fuente para la barra de escala desde el cuadro combinado *Font of bar*
6. Establece el *Tamaño de la barra* en la unidad activa
7. Optionalmente marque *Ajustar automáticamente a un número redondo al redimensionar* para mostrar valores de fácil lectura
8. Elija la ubicación desde el cuadro combinado *Emplazamiento*
9. Puede refinar la ubicación del elemento configurando un `Margen desde el borde (lienzo) horizontal o vertical. Estos valores pueden estar en **Milímetros** o **Píxeles** o configurarse como un **Porcentaje** del ancho o alto del lienzo del mapa.
10. Click en *Aplicar* para verificar que se ve como se esperaba u *OK* si está satisfecho.

Extensión del diseño

Extensiones de diseño agrega las extensiones de :ref:`map item(s) <layout_map_item>`en los diseños de impresión al lienzo. Cuando está habilitado, las extensiones de todos los elementos del mapa dentro de todos los diseños de impresión se muestran usando un borde ligeramente punteado etiquetado con el nombre del diseño de impresión y el elemento del mapa. Puede controlar el estilo y el etiquetado de las extensiones de diseño mostradas. Esta decoración es útil cuando modifica la posición de los elementos del mapa, como las etiquetas, y necesita saber la región visible real de los diseños de impresión.



Figura 7.11: Ejemplo de extensiones de diseño que se muestran en un proyecto QGIS con dos diseños de impresión. El diseño de impresión llamado “Vistas” contiene dos elementos de mapa, mientras que el otro diseño de impresión contiene un elemento de mapa.

Para añadir la extensión(es) de diseño:

1. Seleccione *Ver* ► *Decoraciones* ► *Extensiones de diseño* para abrir el diálogo

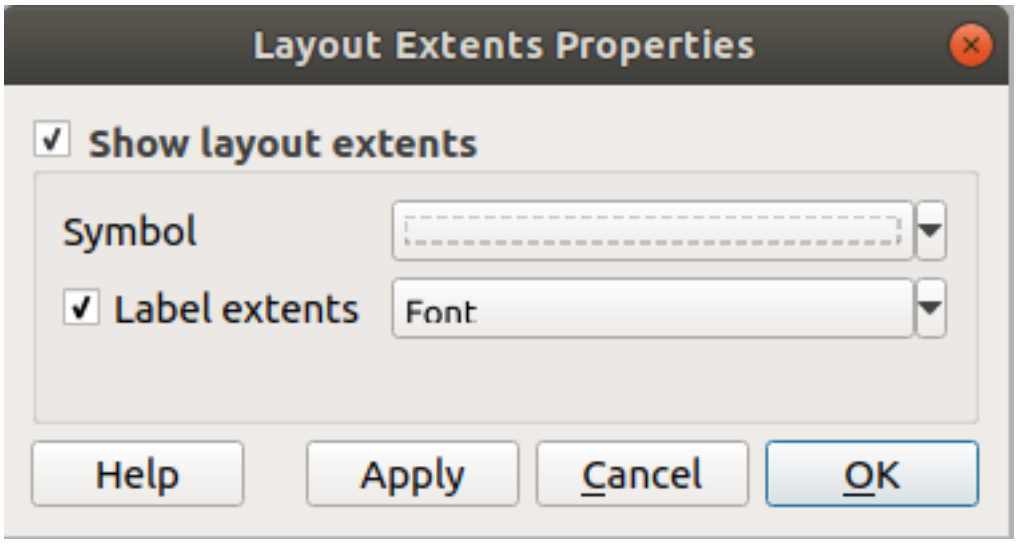


Figura 7.12: El diálogo de Extensión de Diseño

2. Asegúrese de que *mostra extensión diseño* está marcada.

3. Opcionalmente cambie el símbolo y etiquetado de las extensiones.
4. Click en *Aplicar* para verificar que se acerca a lo esperado y pulse *OK* si está satisfecho.

> **Truco**
>
> **Ajustes de Ornamentación**
>
> Cuando graba un archivo de proyecto de QGIS, algunos cambios que haya hecho a la Cuadrícula, Flecha Norte,, Barra de Escala,, Derechos de Copia y Extensiones de diseño serán guardados en el proyecto y recuperados la próxima vez que cargue el proyecto.

7.1.7 Herramientas de anotaciones

Las anotaciones son otro tipo de elementos que se añaden al lienzo del mapa para proporcionar información adicional que no puede ser representada por las capas renderizadas. A diferencia de *etiquetas* que se basan en valores de atributos almacenados en capas vectoriales, las anotaciones son detalles independientes, almacenados en el propio archivo del proyecto.

QGIS dispone de dos familias de anotaciones:

- **Anotaciones de obejtos espaciales**: son objetos espaciales georreferenciados reales de tipo texto, marcador, línea o polígono almacenadas en un tipo de capa especial denominada «capa de anotaciones». Están vinculadas a una ubicación geográfica concreta, lo que significa que mover el mapa, cambiar la escala o cambiar la proyección no hará que las anotaciones salten por el mapa. Por el contrario, permanecerán fijas en el lugar donde las hayas dibujado.
- **Anotaciones de globo**: son anotaciones individuales HTML o de formulario dentro de una burbuja. Pueden asociarse a cualquier capa para su visibilidad y dibujarse sobre el lienzo del mapa. El tamaño depende de la escala del lienzo del mapa, y su posición puede anclarse.

> **Truco**
>
> **Diseño del mapa con anotaciones**
>
> Puede imprimir o exportar anotaciones con su mapa a varios formatos usando:
>
> - las herramientas de exportación del lienzo de mapa disponibles en el menú *Proyecto*
> - *print layout*, en cuyo caso debe verificar *Dibujar elementos del lienzo del mapa* en las propiedades del elemento del mapa correspondiente

La *Barra de herramientas de anotaciones* proporciona un conjunto de herramientas para crear e interactuar con ambas familias de anotaciones.

Tabla 7.3: Acciones de la barra de herramientas de anotaciones

Herramienta	Uso	Alcance
Nueva capa de anotación.	Crear una nueva capa para almacenar anotaciones	*Feature annotations*
Propiedades principales de la capa de anotación	Ajustes de control de la capa de anotación principal	
Modificar Anotaciones	Seleccionar, desplazar, redimensionar y modificar las propiedades simbológicas de las anotaciones	
Crear Anotación de Polígono	Crear una anotación como objeto espacial polígono	
Crear Anotación de Línea	Crear una anotación como objeto espacial polilínea	
Crear Anotación de Marcador	Crear una anotación como objeto espacial punto	

continúe en la próxima página

Tabla 7.3 – proviene de la página anterior

Herramienta	Uso	Alcance
Crear anotación de texto en un punto	Create an annotation as a text label anchored at a point	*Balloon annotations*
Crear anotación de texto a lo largo de la línea	Crear una anotación como texto curvo a lo largo de un objeto espacial línea	
Create Text Annotation in Rectangle	Create an annotation as a text label in a rectangular text box	
Crear anotación de imagen	Create an annotation representing an `.SVG` or raster image	
Anotación HTML	Select and create annotation with a `HTML` formatted text	
Anotación de formulario	Seleccionar y crear una anotación que muestre los atributos de una capa vectorial en un archivo de formulario personalizado	

Nota

A partir de QGIS 3.40, ya no es posible crear anotaciones de globo de tipo SVG o texto. Al cargar proyectos antiguos, cualquiera de estas anotaciones se convertirá automáticamente a los nuevos tipos de elementos de anotación de imagen y texto rectangular.

Anotaciones de objetos espaciales

Las anotaciones de objetos espaciales se almacenan en **capas de anotación**. A diferencia de las capas convencionales, una capa de anotación sólo está disponible en el proyecto actual y puede contener características de diferentes tipos (texto, marcador, línea, polígono, imagen). La capa no tiene atributos ni simbología asociada, sino que cada característica puede simbolizarse elemento por elemento, a través del panel *Estilos de capa*.

QGIS dispone de dos tipos de capas de anotación:

- Una *Capa de anotación* común: puede crear una con la herramienta Nueva capa de anotación. Aparece en el panel *Capas*, permitiéndole controlar la visibilidad de sus objetos espaciales, moverla para que se muestre por encima o por debajo de determinadas capas de su mapa, como cualquier capa común. Haga doble clic en la capa para acceder a sus propiedades.
- La *Capa principal de anotaciones*: De forma predeterminada, aquí es donde se almacenan las anotaciones cuando no hay ninguna capa de anotaciones disponible en el proyecto o está seleccionada en el momento de la creación. Esta capa se dibuja siempre en la parte superior del mapa y no aparecerá en el panel *Capas* junto a las demás capas del proyecto, lo que significa que sus características son siempre visibles. La entrada *Propiedades de la capa principal de anotaciones* de la barra de herramientas *Anotaciones* le ayuda a abrir su cuadro de diálogo de propiedades.

Propiedades de la capa

El diálogo de propiedades de una capa de anotación proporciona las siguientes pestañas:

- *Información*: un cuadro de diálogo de solo lectura que representa un lugar interesante para obtener rápidamente información resumida y metadatos en la capa actual. Esto puede incluir la extensión de la capa, el recuento de elementos por tipo de anotación y el recuento total, detalles de SRC, etc.
- *Fuente*: define la configuración general de la capa de anotaciones. Usted puede:
  - Establezca un *Nombre de capa* que se utilizará para identificar la capa en el proyecto (en el *Panel de capas*, con expresiones, …)

  - Mostrar el sistema de referencia de coordenadas (SRC) *asignado a la capa*: puede cambiar el SRC de la capa, seleccionando uno utilizado recientemente en la lista desplegable o pulsando el botón  Seleccionar SRC (véase *Selector del Sistema de Coordenadas de Referencia*). Utilice este proceso sólo si el SRC aplicado a la capa es incorrecto o si no se ha aplicado ninguno.

- *Renderizado*:
  - Puede establecer la escala *Maximum (inclusive)* y *Minimum (exclusive)* , definiendo un rango de escala en el cual las características serán visibles. Fuera de este rango, están ocultos. El botón  Set to current canvas scale le ayuda a usar la escala del lienzo del mapa actual como límite de la visibilidad del rango. Ver *Selector de escala de visibilidad* para mas información.
  - *Capa vinculada*: permite establecer opcionalmente una capa de visibilidad vinculada para la capa de anotaciones. Si se establece, las anotaciones sólo se dibujarán cuando la capa vinculada sea visible en el mapa.
  - *Opacidad*: Con esta herramienta puede hacer visible la capa subyacente en el lienzo del mapa. Utilice el control deslizante para adaptar la visibilidad de la capa vectorial a sus necesidades. También puede definir con precisión el porcentaje de visibilidad en el menú situado junto al deslizador.
  - *Modo de mezcla* en el nivel *Capa*: Con estas herramientas puede conseguir efectos de renderizado especiales que antes sólo conocía de los programas gráficos. Los píxeles de las capas superpuestas y subyacentes se mezclan mediante los ajustes descritos en *Modos de Mezcla*.
  - Aplicar *paint effects* en todas las entidades de la capa con el botón *Draw Effects*.

Algunas de estas opciones son accesibles desde las propiedades *Symbology* de la anotación de objeto espacial.

Interactuando con entidades

Las anotaciones de objetos espaciales disponen de herramientas específicas para su creación en función de su tipo.

Tabla 7.4: Creación de anotación de objeto espacial

Familia / basado en	Herramienta de Anotación	Procedimiento
Geometría	 Crear Anotación de Polígono	Es como digitalizar un objeto espacial vectorial de punto, línea o polígono. Todos los métodos abreviados habituales para crear objetos espaciales se aplican al crear elementos de anotación. Una anotación lineal o poligonal se dibuja haciendo clic con el botón izquierdo del ratón una vez para cada vértice, y un último clic con el botón derecho para completar la forma. Se puede activar el ajuste mientras dibuja, puede utilizar la *Herramientas avanzadas de digitalización* para colocar los vértices con precisión, e incluso cambiar las *herramientas de dibujo* a, por ejemplo, el modo streaming para formas completamente libres.
	 Crear Anotación de Línea	
	 Crear Anotación de Marcador	
Texto	Crear anotación de texto en un punto	Haz clic con el botón izquierdo del ratón en el lienzo del mapa.
	 Crear anotación de texto a lo largo de la línea	Haga clic con el botón izquierdo para añadir vértices a la geometría de la línea de base de texto y haga clic con el botón derecho para terminar. Como en el caso anterior, también están disponibles las funciones de digitalización.

continúe en la próxima página

Tabla 7.4 – proviene de la página anterior

Familia / basado en	Herramienta de Anotación	Procedimiento
	 Crear anotación de texto en rectángulo.	Dibuje la caja rectangular del texto, haciendo clic con el botón izquierdo para añadir una primera esquina y haciendo clic de nuevo con el botón izquierdo para añadir la opuesta.
Imágen	 Crear anotación de imagen	Dibuje la caja rectangular de la imagen, haciendo clic con el botón izquierdo para añadir una primera esquina y haciendo clic de nuevo con el botón izquierdo para añadir la opuesta. A continuación, seleccione una imagen o un archivo SVG del explorador de archivos.

A diferencia de las capas comunes, una capa de anotación no necesita estar activa antes de seleccionar sus características. Basta con utilizar la herramienta  Modificar anotaciones para interactuar con cualquier anotación:

- **Selección**: haga clic izquierdo en la anotación. De forma predeterminada, las anotaciones se representan en el orden de creación, lo que significa que las anotaciones recientes se colocarán encima de las más antiguas. Es posible que tengas que jugar con la propiedad del índice Z de las funciones para seleccionar las que se ubicarían encima.
- **Mover**: Haga clic con el botón izquierdo del ratón en un elemento de anotación seleccionado para empezar a moverlo. Un clic con el botón derecho o pulsando la tecla `Esc` cancela el desplazamiento, mientras que un segundo clic con el botón izquierdo lo confirmará. El desplazamiento también puede controlarse pulsando las teclas del cursor:
  - `Mayús+tecla` para grandes movimientos
  - `Alt+tecla` para movimiento `1 px`
- **Modificación de la geometría**: desplázate sobre una anotación y aparecerán cuadrados morados en los nodos de su geometría subyacente. Haga clic con el botón izquierdo en el cuadrado, desplácese y vuelva a hacer clic. Para las anotaciones lineales o poligonales, al hacer doble clic en un segmento se añadirá un nuevo vértice.
- **Borrar**: Si pulsa la tecla `Supr` o `Retroceso` mientras una anotación está seleccionada, se eliminará dicha anotación.
- *Cambiar la simbología de los objetos espaciales*

Simbología de los objetos espaciales

Una anotación seleccionada mostrará sus propiedades *Simbología* en el panel *Estilos de capa*.

- Para las anotaciones de polígonos, polilíneas y marcadores, puede modificar la apariencia utilizando todas las capacidades de las propiedades *symbol*
- Para las anotaciones basadas en texto, un área con herramientas de edición de texto le ayuda a construir la cadena a mostrar. Puede ser multilínea, utilizar formato HTML y basarse en funciones de expresión de QGIS. Además, puede modificar el aspecto utilizando todas las funciones de las *propiedades de formato de texto*.

  Dependiendo del tipo de anotación, existen opciones adicionales.
  - Para la anotación de texto en un punto, también es posible:
    - establece el texto *Alineación* (izquierda, centro o derecha del punto de anclaje)
    - configura si el texto debe *Ignorar la rotación del mapa* o *Rotar con el mapa*. En ambos casos, se puede establecer un *Ángulo* personalizado para la orientación de la característica.
  - Para anotaciones de texto a lo largo de una línea, configure un *Desplazamiento desde línea* en la unidad de su elección
  - Para la anotación de texto en rectángulo, también es posible:

    * establece el texto *Alineación horizontal* (izquierda, centro, derecha o justificar) y *Alineación vertical* (arriba, centro vertical o abajo) en la caja rectángulo
    * configure para el rectángulo, la distancia *Márgenes* en las unidades de su elección, los colores *Marco* y *Fondo*.

- Las anotaciones de imágenes te permiten mostrar una *imagen raster* o una *imagen SVG*, utilizando el selector de archivos incrustados. La ubicación y el tamaño de la imagen se pueden configurar como:
  - *Tamaño dependiente de la escala*: la imagen se escala con el mapa, apareciendo más grande al acercar y más pequeña al alejar. El tamaño se define mediante un único tamaño de referencia en unidades del mapa (por ejemplo, metros o grados, dependiendo del SRC de la capa). Con  *Bloquear relación de aspecto*, puede mantener el tamaño de la anotación proporcional al tamaño de la imagen incrustada, o estirar la imagen para que ocupe toda la extensión de la anotación.
  - *Tamaño fijo*: la anotación de la imagen mantiene un tamaño constante en unidades de pantalla o de salida (por ejemplo, píxeles o milímetros), independientemente de la escala del mapa. El *Ancho* y el *Alto* de la anotación se pueden rellenar por separado o ajustarse proporcionalmente al tamaño de la imagen incrustada.
  - *Relativo al mapa*: la anotación de la imagen siempre se mostrará en la misma posición relativa a los límites del lienzo del mapa, independientemente de la escala del mapa. El *Ancho* y el *Alto* de la anotación se pueden rellenar por separado o ajustar proporcionalmente al tamaño de la imagen incrustada.

  Además, puede activar la visualización de *Marco* y *Fondo* de la caja rectangular, y configurarlos mediante *símbolos de relleno*.
- Para las anotaciones de texto en un punto, texto dentro de un rectángulo e imagen, puede *Mostrar llamada* cuando el punto de anclaje de la imagen o el texto se desplaza de su ubicación predeterminada. Para crear una llamada para una anotación:
  1. Seleccione el elemento de anotación
  2. Mueva el punto de anclaje, es decir, haga clic en el nodo central y verde `X` y haga clic en la nueva colocación
  3. En las propiedades de la anotación, pulse el botón … para configurar las *propiedades de la llamada*.

  Truco

  Al aplicar una llamada de globo a una anotación de texto o imagen, obtendrá su visualización de anotación de globo al estilo antiguo.
- Configure una etiqueta *Escala de referencia*: indica la escala del mapa a la que se refieren los tamaños de símbolos o texto que utilizan unidades basadas en papel (como milímetros o puntos). Los tamaños se escalarán en consecuencia siempre que el mapa se visualice a una escala diferente. Por ejemplo, una línea de 2 mm de ancho a 1:2000 *Escala de referencia* se representará con 4 mm cuando el mapa se visualice a 1:1000.
- Establecer un *Z-index*: un objeto espacial con un índice más alto se coloca encima de los objetos espaciales con un índice más bajo. Una configuración conveniente tanto para la visualización como para la selección de funciones.
- Modificar algunos de los ajustes de *Representación de capas*

Anotaciones de globos

Puede añadir anotaciones de globo a través del menú *Editar* ► *Añadir anotación* ► o desde la *Barra de herramientas de anotaciones*:

- HTML Annotation to display a HTML-formatted string or a whole `HTML` file as an annotation.
- Form Annotation: useful to display attributes of a vector layer in a customized `.ui` file. This is similar to the *custom attribute forms*, but displayed in an annotation item.

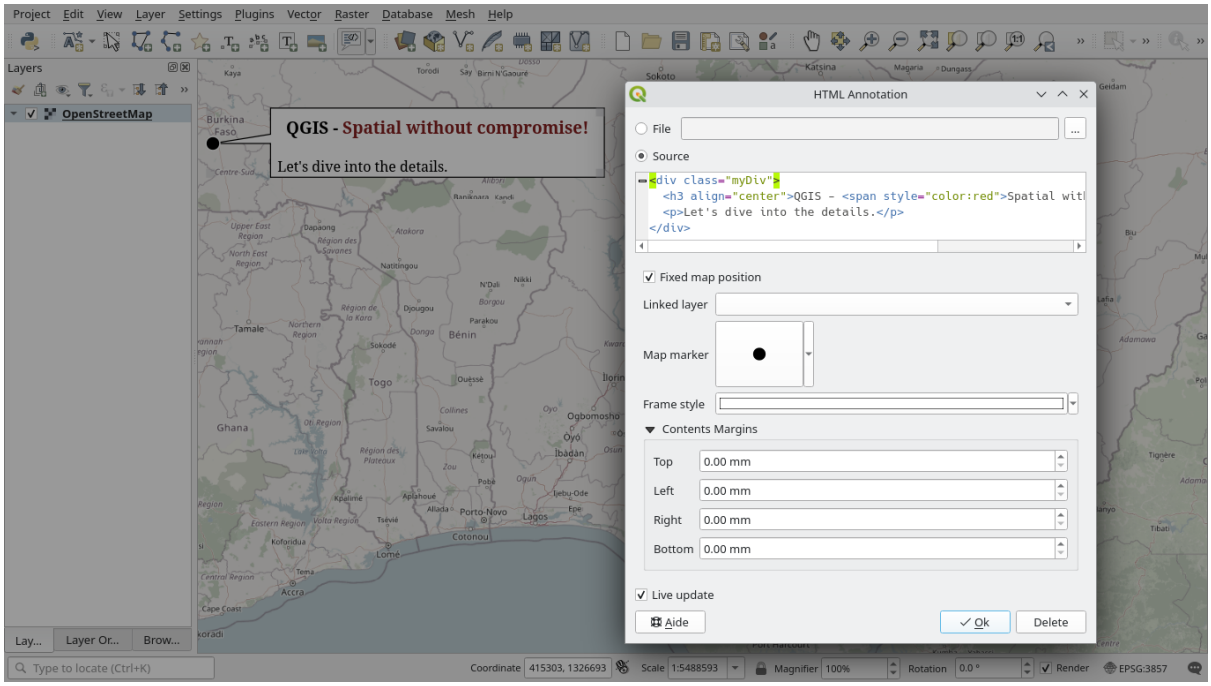


Figura 7.13: Example of a HTML balloon annotation

Para añadir una anotación de globo, seleccione la herramienta correspondiente y haga clic en el lienzo del mapa. Se añade un globo vacío. Haga doble clic sobre él y se abrirá un cuadro de diálogo con varias opciones. Este diálogo es casi el mismo para todos los tipos de anotación:

- *File*: Fill it with the path to a `.html` or `.ui` file, depending on the type of annotation, to serve as contents of the annotation.
- *Source*: Only available for HTML annotations, it allows to serve the annotation with a plain or HTML-formatted text instead of a file.
- *Fixed map position*: when unchecked, the balloon placement is based on a screen position (instead of the map), meaning that it's always shown regardless of the map canvas extent.
- *Capa vinculada*: asocia la anotación con una capa de mapa, haciéndola visible solo cuando esa capa es visible.
- *Marcador de Mapa*: using *QGIS symbols*, configura el símbolo para que se muestre en la posición de anclaje del globo (solo se muestra cuando la :guilabel:"Posición del mapa fija" está marcada).
- *Estilo de Marco*: establece el color de fondo del marco, la transparencia, el color del trazo o el ancho del globo utilizando símbolos QGIS.
- *Margenes de contenido*: establece los márgenes interiores del marco de anotación.
- *Actualización en vivo* le permite obtener una vista previa en vivo de sus cambios.

Cuando una herramienta de anotación de globos está activa, también puede:

- Seleccione una anotación

- Cambiar el tamaño de una anotación
- Mueva una anotación por posición en el mapa (arrastrando el marcador del mapa) o moviendo solo el globo.
- Eliminar una anotación: selecciónela y presione el botón `Supr` o `Retroceder`, o haga doble clic en ella y presione el botón *Suprimir* en el cuadro de diálogo de propiedades.
- Haga clic derecho y en el menú contextual:
  - *Copiar coordenada* del marcador del mapa de la anotación en varios SRC
  - *Editar* propiedades de la anotación. Lo mismo que hacer doble clic en la anotación.
  - *Eliminar* la anotación

7.1.8 Mediciones

Información general

QGIS proporciona cuatro maneras de medir geometrías:

- Herramientas de medición interactivas
- midiendo en la Calculadora de campos
- mediciones derivadas en la herramienta: ref:*identify*
- la herramienta de análisis vectorial: *Vectorial* ► *Herramientas de Geometría* ► *Exportar/Añadir Columnas de Geometría*

La medición funciona dentro de los sistemas de coordenadas proyectadas (por ejemplo, UTM) y los datos no proyectados. Las primeras tres herramientas de medición se comportan igualmente con la configuración global del proyecto:

- A diferencia de la mayoría de los otros SIG, la métrica de medición predeterminada es elipsoidal, utilizando el elipsoide definido en *Proyecto -> Propiedades … -> General*. Esto es cierto tanto cuando se definen sistemas de coordenadas geográficas como proyectadas para el proyecto.
- Si desea calcular el área proyectada/planimétrica o la distancia utilizando las matemáticas cartesianas, el elipsoide de medición debe establecerse en «Ninguno/Planimétrico» (*Proyecto -> Propiedades … -> General*). Sin embargo, con un CRS geográfico (es decir, no proyectado) definido para los datos y el proyecto, la medición del área y la distancia será elipsoidal.

Sin embargo, ni la herramienta de identificación ni la calculadora de campo transformarán sus datos al CRS del proyecto antes de medir. Si desea lograr esto, debe utilizar la herramienta de análisis vectorial: *Vector -> Herramientas de geometría -> Agregar atributos de geometría ….* Aquí, la medición es planimétrica, a menos que elija la medición elipsoidal.

Mida la longitud, las áreas, los rumbos y los ángulos de forma interactiva

Haga clic en la en la barra de herramientas de atributos para comenzar las mediciones. La flecha hacia abajo cerca del icono cambia entre longitud, área, rumbo o ángulo. La unidad predeterminada utilizada en el cuadro de diálogo es la establecida en el menú *Project* ► *Properties…* ► *General*

Para la *Measure Line* y la *Measure Area* las medidas se pueden realizar en *Cartesian* o *Ellipsoidal* measure.

> **Nota**
>
> **Configurando la herramienta medición**
>
> Al medir la longitud o el área, al hacer click en el botón *Configuración* en la parte inferior del widget se abre el menú *Configuración -> Opciones -> Herramientas de mapa*, donde puede seleccionar el color de la banda elástica, la precisión de las medidas y el comportamiento de la unidad. También puede elegir sus unidades de medida o ángulo preferidas, pero tenga en cuenta que esos valores se anulan en el proyecto actual mediante la selección realizada en *Proyecto -> Propiedades … -> General*, y por la selección realizada en el widget de medición.

Todos los módulos de medición utilizan la configuración de ajuste del módulo de digitalización (consulte la sección *Configuración de la tolerancia de ajuste y el radio de búsqueda*). Por lo tanto, si desea medir exactamente a lo largo de una entidad de línea o alrededor de una entidad de polígono, primero configure su tolerancia de ajuste de capa. Ahora, cuando use las herramientas de medición, cada click del ratón (dentro de la configuración de tolerancia) se ajustará a esa capa.

La herramienta **|medir|** Medir línea mide distancias entre puntos dados. La herramienta permite hacer clic en los puntos del mapa. La longitud de cada segmento, así como el total, se muestran en la ventana de medida. En la ventana de medidas, verá las coordenadas de todos sus puntos y distancias. Tenga en cuenta que la primera fila sólo contendrá coordenadas, ya que representa su punto de partida. Ahora es posible copiar todas las medidas de sus líneas al portapapeles utilizando el botón *Copiar*. Haciendo clic en el botón *Configuración* accederá a *Configuración de copia de la herramienta de medición* donde podrá configurar las opciones de *copia*. Para detener la medición, pulse el botón derecho del ratón.

Tenga en cuenta que puede utilizar la lista desplegable situada junto al total para cambiar las *unidades de medida* interactivamente mientras se trabaja con la herramienta de medida. Esta unidad se conserva para el widget hasta que se crea un nuevo proyecto o se abre otro proyecto.

La sección *Información* en el cuadro de diálogo explica cómo se realizan los cálculos de acuerdo con la configuración de CRS disponible.

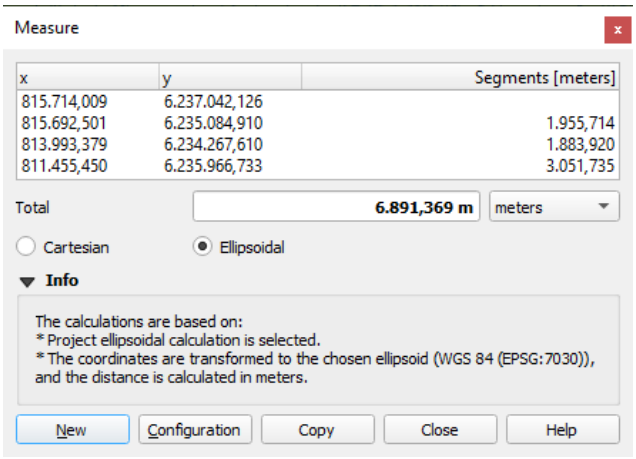


Figura 7.14: Medir Distancia

|medirÁrea| Medir Área: Las áreas también se pueden medir. En la ventana de medida aparece el tamaño acumulado del área. Haga clic con el botón derecho del ratón para detener el dibujo. La sección Información también está disponible, así como la posibilidad de cambiar entre diferentes *unidades de área*.

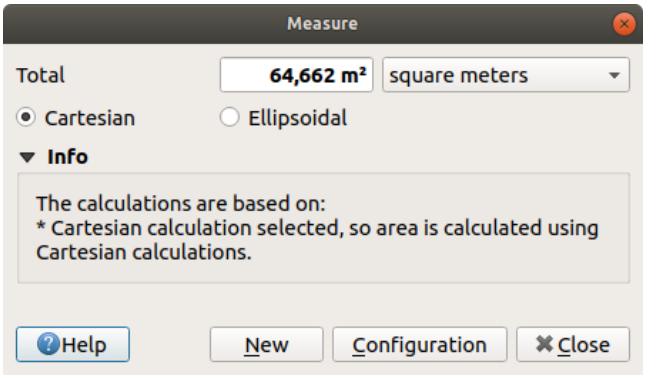


Figura 7.15: Medir Área

Measure Bearing: También puede medir rodamientos. El cursor adquiere forma de cruz. Haga clic para dibujar el primer punto del rumbo, luego mueva el cursor para dibujar el segundo punto. La medida se muestra en un cuadro de diálogo emergente.

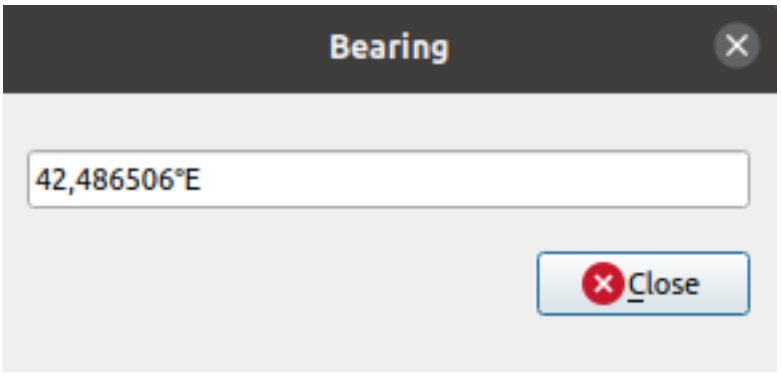


Figura 7.16: Medir llevando

Medir ángulo: También puede medir ángulos. El cursor se vuelve en forma de cruz. Haga click para dibujar el primer segmento del ángulo que desea medir, luego mueva el cursor para dibujar el ángulo deseado. La medida se muestra en un cuadro de diálogo emergente.

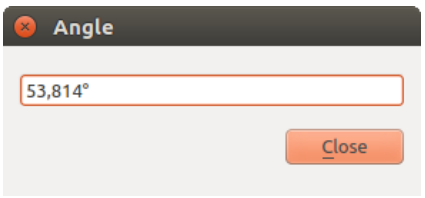


Figura 7.17: Medir Ángulo

7.1.9 Ajustando vistas de mapa adicionales

También es posible abrir vistas de mapa adicionales cuyo contenido podría divergir del estado actual del panel *Capas*. Para añadir una nueva vista de mapa, vaya a *Vista* ► |nuevoMapa| *Nueva vista de mapa*. Se añade a QGIS un nuevo widget flotante, que imita el renderizado de la vista de mapa principal. Puede añadir tantas vistas de mapa como necesite. Pueden mantenerse flotantes, colocadas una al lado de la otra o apiladas una encima de la otra.

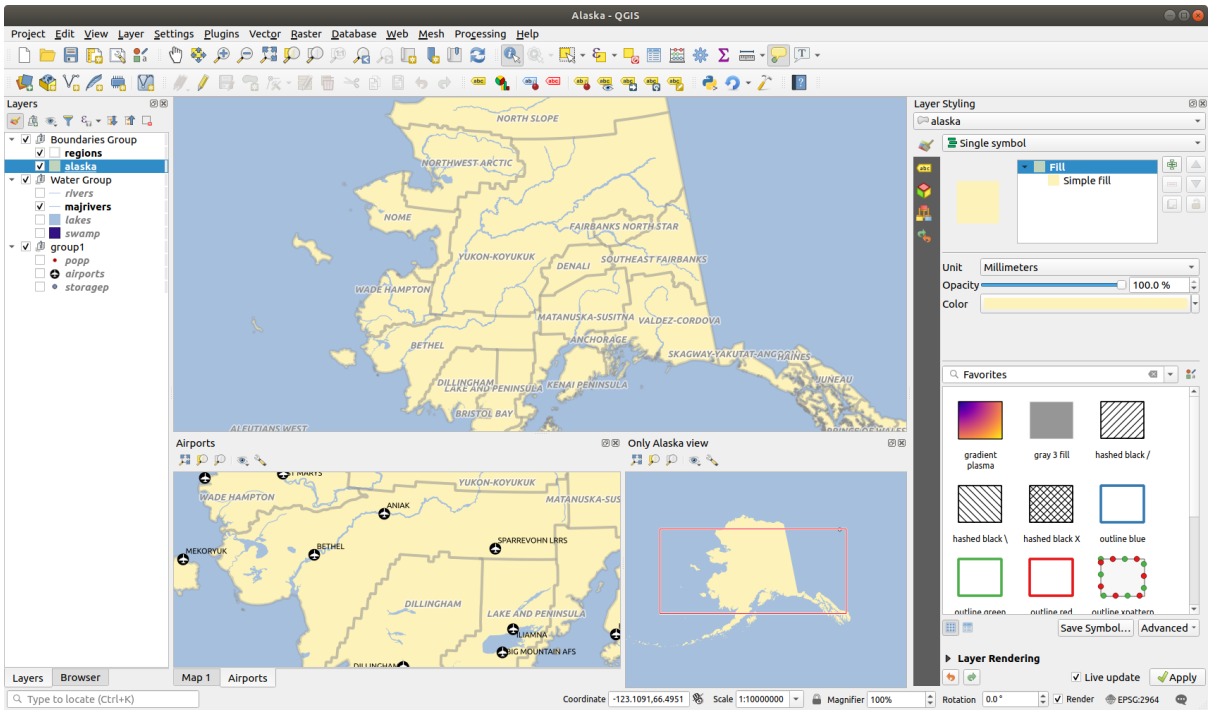


Figura 7.18: Múltiples vistas de mapas con diferentes ajustes

En la parte superior de un lienzo de mapa adicional, hay una barra de herramientas con las siguientes capacidades:

- zoomFullExtent| Zoom Full, Zoom to Selection and Zoom to Layer(s) para navegar dentro de la vista
- Ajustar Tema Visual para seleccionar el *tema de mapa* a mostrar en el visor de mapa. Si se establece a `(none)`, la vista seguirá los cambios del panel *Capas*.
- Configuración de visualización para configurar la vista de mapa:
  - *Sincronizar centro de vista con mapa principal*: sincroniza el centro de las vistas del mapa sin cambiar la escala. Esto le permite tener un estilo de vista general o un mapa ampliado que sigue al centro del lienzo principal.
  - *Sincronizar vista a selección*: igual que el zoom a la selección
  - *Escala*
  - *Rotación*
  - *Magnificación*
  - *Sincronizar Escala* con la escala del mapa principal. Una *Factor de Escala* puede entonces ser aplicado, permitiéndole tener una vista que es p.ej. siempre 2 veces la escala del lienzo principal.
  - *Mostrar Anotaciones*
  - *MOstrar Posición Cursor*
  - *MOstrar extensión del lienzo principal*

- *Mostrar Etiquetas*: permite ocultar etiquetas independientemente de que estén establecidas en las propiedades de las capas mostradas
- *Cambiar SCR del mapa…*
- *Renombrar Vista…*

7.1.10 Exportando la vista de mapa

Los mapas que hagas se pueden diseñar y exportar a varios formatos utilizando las capacidades avanzadas de *imprimir diseño o reportar*. También es posible exportar directamente el renderizado actual, sin un diseño. Esta rápida «captura de pantalla» de la vista del mapa tiene algunas características convenientes.

Para exportar el lienzo de mapa con la representación actual:

1. Vaya a *Proyecto* ► *Importar/Exportar*
2. Dependiendo de su formato de salida, seleccione
   - *Exportar Mapa a Imagen…*
   - or *Exportar Mapa a PDF…*

Las dos herramientas le brindan un conjunto común de opciones. En el cuadro de diálogo que se abre:

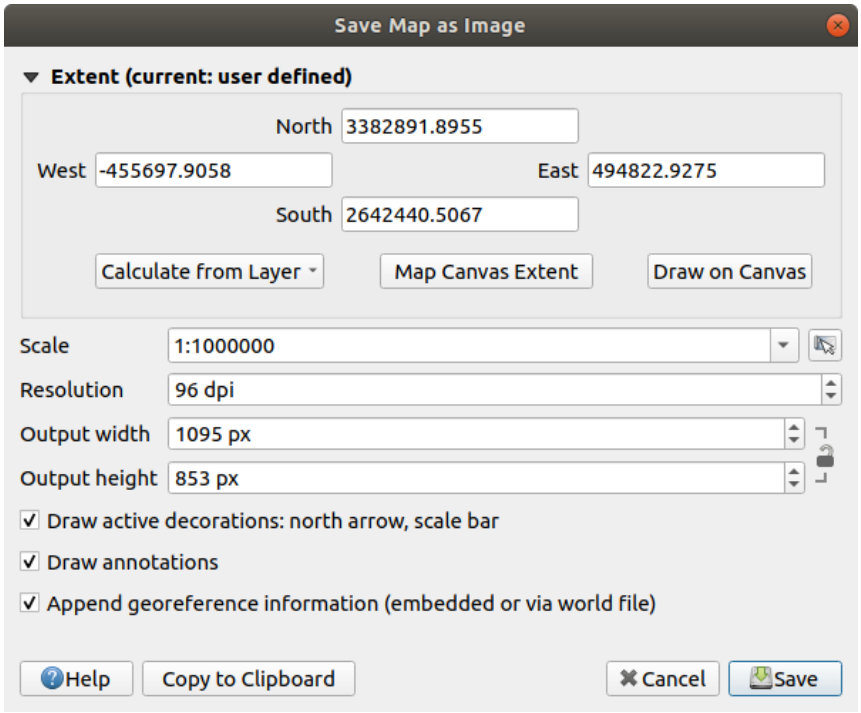


Figura 7.19: El cuadro de diálogo Guardar mapa como imagen

1. Escoja la *Extent* a exportar: puede ser la extensión de la vista actual (la predeterminada), la extensión de una capa o una extensión personalizada dibujada sobre el lienzo del mapa. Las coordenadas del área seleccionada se muestran y se pueden editar manualmente.
2. Introduzca la *Escala* del mapa o seleccionela desde *escalas predefinidas*: cambiando la escala cambiará el tamaño de la exportación (desde el centro).
3. Establezca la *Resolución* de la salida

4. Controle la *anchura de Salida* y *altura de Salida* en pixels de la imagen: basada por defecto en la resolución y extensión actual, se pueden personalizar y redimensionar la extensión del mapa (desde el centro). La relación de tamaño se puede bloquear, lo que puede ser particularmente conveniente al dibujar la extensión en el lienzo.

5. ☑ *Dibujar decoraciones activas*: en uso *decoraciones* (barra de escala, título, malla, flecha del nortea..) son exportadas con el mapa

6. ☑ *Dibujar anotaciones* para exportar alguna *anotación*

7. ☑ *Agregar información de georreferencia (incrustada o vía archivo de mundo)*: dependiendo del formato de salida, un archivo mundial con el mismo nombre (con la extensión `PNGW` para imágenes `PNG`, `JPGW` para `JPG`, …) se guarda en el mismo carpeta como tu imagen. El formato `PDF` incrusta la información en el archivo PDF.

8. Al exportar a PDF, están disponibles mas opciones en el Cuadro de diálogo *Guardar mapa como PDF…* :

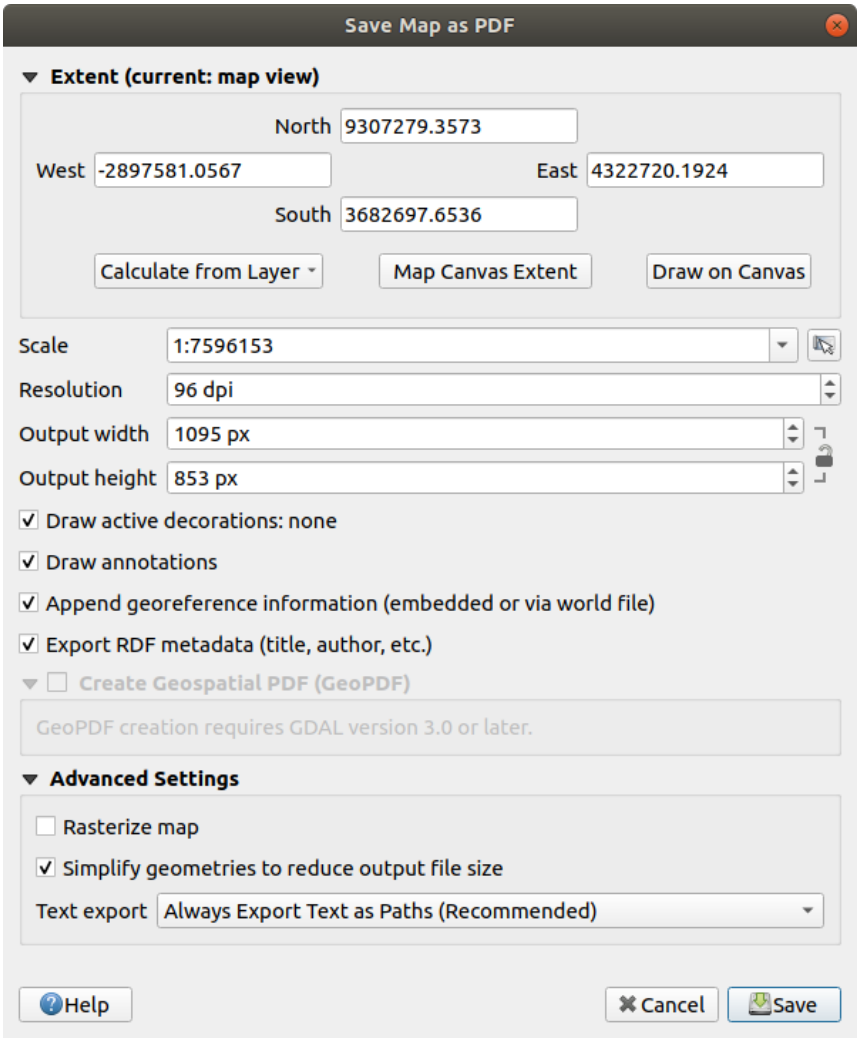


Figura 7.20: El diálogo Guardar Mapa como PDF

- ☑ *Exportar metadatos RDF* del documento, como el título, autor, fecha, descripción…
- ☐ *Crear PDF geoespacial*: Genera un archivo PDF georreferenciado. Puede:
  - Elija el formato PDF geoespacial *Format*.
  - ☑ *Incluir información de características vectoriales* en el archivo PDF Geoespacial: incluirá toda la información de geometría y atributos de las características visibles dentro del mapa en el archivo PDF Geoespacial de salida.

> **Nota**
>
> Un archivo PDF geoespacial también puede utilizarse como fuente de datos. Para obtener más información sobre la compatibilidad con PDF geoespaciales en QGIS, consulte https://north-road.com/2019/09/03/qgis-3-10-loves-geospatialpdf/.

- *Rasterize map*
- *Simplificar geometrías para reducir el tamaño del archivo de salida*: Las geometrías se simplificarán al exportar el mapa eliminando los vértices que no sean perceptiblemente diferentes a la resolución de exportación (por ejemplo, si la resolución de exportación es `300 dpi`, se eliminarán los vértices que estén a menos de `1/600 pulgadas` de distancia). Esto puede reducir el tamaño y la complejidad del archivo de exportación (los archivos muy grandes pueden no cargarse en otras aplicaciones).
- Establecer la *Exportación de texto*: controla si las etiquetas de texto se exportan siempre o preferentemente como *objetos texto o contorno.*.

9. Click *Save* para seleccionar la ubicación, el nombre y el formato del archivo.

   Al exportar a imagen, también es posible *Copy to clipboard* el resultado esperado de la configuración anterior y pegue el mapa en otra aplicación como LibreOffice, GIMP …

7.2 Vista de Mapa 3D

El soporte de visualización 3D se ofrece a través de la vista de mapa 3D. Puede crear, gestionar y abrir vistas de mapa 3D a través del menú *Vista* ► *Vistas de mapa 3D* ►:

1. Haciendo clic en *Nueva vista de mapa en 3D* puede crear una nueva vista de mapa 3D. Aparecerá un panel QGIS flotante y acoplable (véase *El diálogo de Visor de Mapa 3D*). Tiene la misma extensión y vista que el lienzo del mapa principal 2D y proporciona un conjunto de herramientas de navegación para convertir la vista en 3D.
2. Haciendo clic en **:menuselección:`Gestionar vistas de mapas 3D`** se accede al Gestor de vistas de mapas 3D. Aquí podrá abrir, duplicar, eliminar y renombrar vistas de mapas 3D.
3. Si ha creado una o más vistas de mapa 3D, las verá listadas en *Vistas de mapa 3D*. Puede activarlas y desactivarlas haciendo clic en. Se guardarán al guardar el proyecto, incluso si están desactivadas.

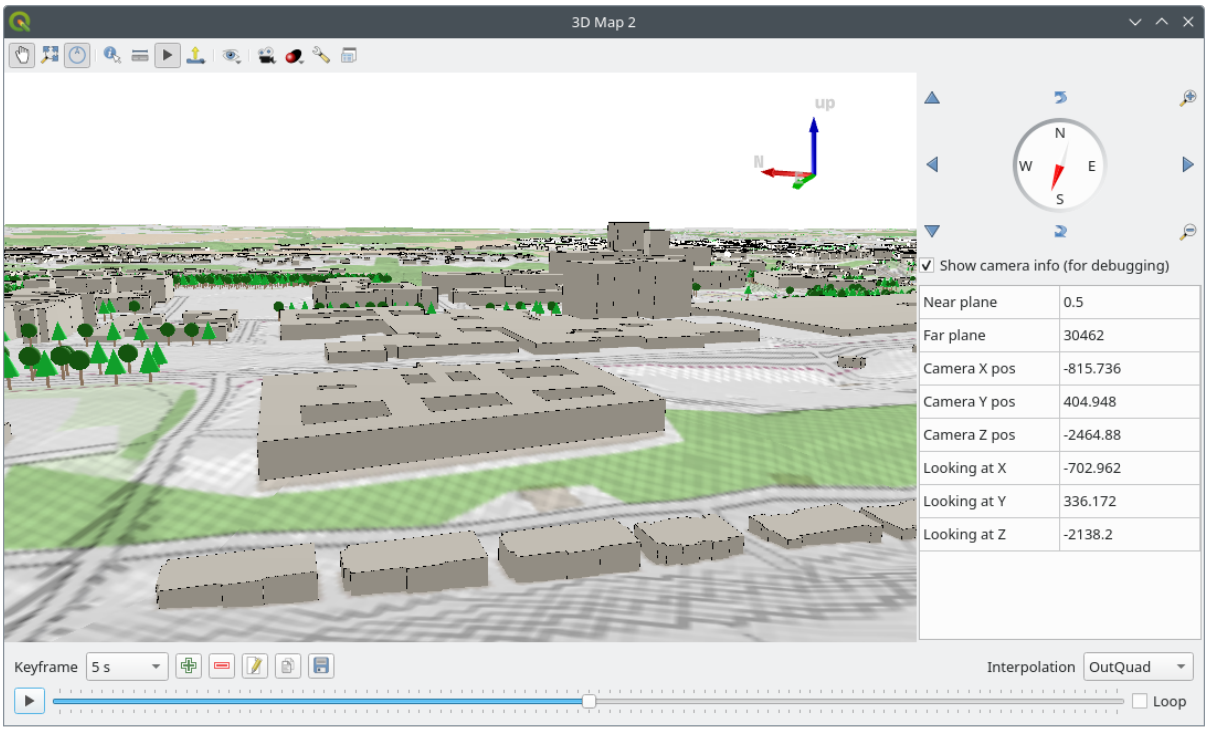


Figura 7.21: El diálogo de Visor de Mapa 3D

Las siguientes herramientas son proporcionadas en la parte superior del panel del Visor de Mapa 3D:

- Control de la cámara: mueve la vista, manteniendo el mismo ángulo y dirección de la cámara
- Zoom completo: redimensiona la vista a la extensión completa de las capas, o a la *extensión de la referencia* si se define
- Conmutar las notificaciones en pantalla: muestra/oculta el widget de navegación (que está pensado para facilitar el control de la vista del mapa).
- Identify: devuelve información sobre el punto del terreno en el que se hizo click o las entidades 3D en las que se hizo click - Más detalles en *Identificando entidades*
- Measurement Line: measures the horizontal, vertical, and total 3D distance between points
  - Click to add points
  - Right-click to finish the measurement
  - Right-clicking twice or pressing *New* will reset the table
- Animations: muestra/oculta el widget *animation player*
- Exportar El menú permite exportar la escena a varios formatos:
  - Guardar como imagen…: exporta la vista actual a un formato de archivo de imagen.
  - Exportar escena 3D: exporta la vista actual como una escena 3D (archivo:file:*.obj*), permitiendo el post-procesado en aplicaciones como Blender… El terreno y las características vectoriales se exportan como objetos 3D. Los ajustes de exportación, que anulan las capas *propiedades* o vista de mapa *configuración*, incluir:
    * *nombre de Escena* y *Carpeta* de destino

    - *Resolución del Terreno*
    - *Resolución de la textura del Terreno*
    - *Escala del Modelo*
    - *Suavizar formas*
    - *Exportar normales*
    - *Exportar texturas*
- Establecer Tema de Vista: Le permite seleccionar el conjuntod e capas a mostrar en la vista de mapa a partir de unos *temas de mapa* predefinidos.
- El menú **|cámara|** Cámara le ayuda a controlar la relación entre las vistas 2D y 3D:
  - Sincronizar las vistas (*La vista de mapa 2D sigue a la cámara 3D* y/o *La cámara 3D sigue a la vista de mapa 2D*)
  - *Mostrar el área visible de la cámara en la vista de mapa 2D*
  - *Establecer escena 3D en vista de mapa 2D*: permite recortar la escena 3D y mostrar solo el terreno y los elementos que se cruzan con una extensión dibujada en el lienzo del mapa 2D. Hay más opciones disponibles en la pestaña *Configuración general*.
  - Cross Section Tool creates a cross section in 3D scene by defining an area on the terrain in the 2D map canvas. Click to set the start point, move the mouse to define the line direction and click again to set the end point. You can adjust the size of the section by moving the mouse after selecting the start and end points. Press `Esc` or right-click to reset the selection.
  - Use Disable Cross Section to remove the cross section.
- **|sombra|** Efectos añade efectos visuales al renderizado 3D, como mostrar *sombras*, *eye dome lighting* o *oclusión ambiental*.
- The Configure… button opens the dialog to configure the 3D map view *settings*.
- Acoplar vista 3D de mapa: cambiar de widget acoplado a ventana de nivel superior

7.2.1 Configuración de la escena

The 3D map view opens with some default settings you can customize. To do so, press the Configure… button at the top of the 3D canvas panel to open the *3D configuration* window.

En la ventana de Configuración 3D hay varias opciones para afinar la escena 3D:

General

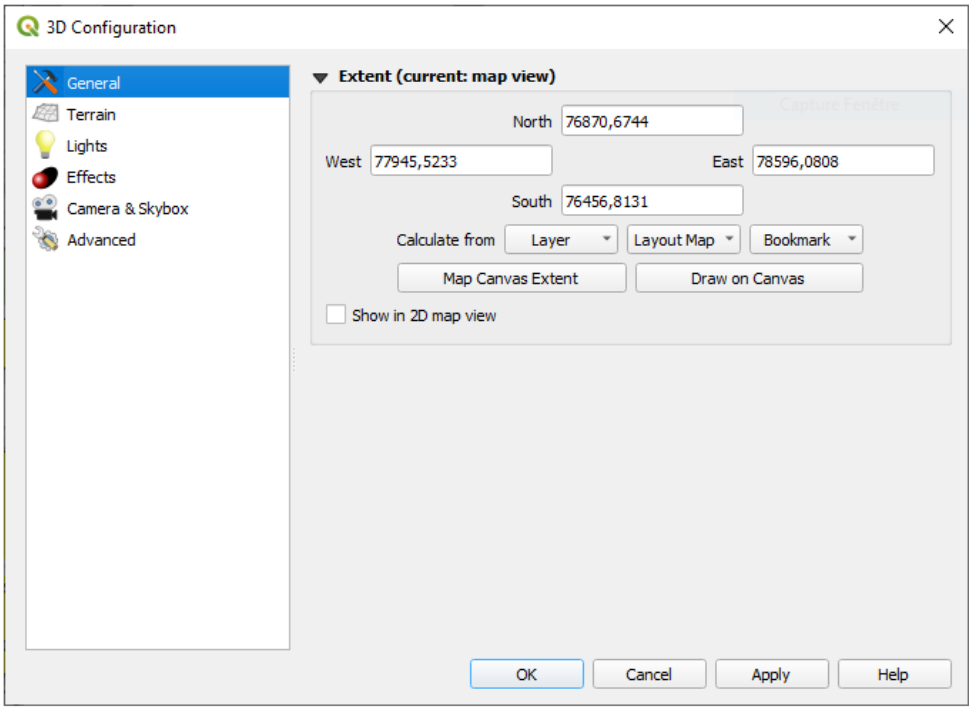


Figura 7.22: El cuadro de diálogo Configuración general del mapa 3D

En la pestaña  *General*, puede:

- Limitar el contenido de la escena 3D a la extensión de un mapa 2D, utilizando el *selector de extensión espacial*: el terreno y los objetos espaciales se recortan en la extensión especificada y sólo se cargan en la escena las partes que caen dentro de dicha extensión. También se tiene en cuenta el rango de elevación de las capas de terreno, malla y nubes de puntos para que la cámara no se sitúe por debajo del contenido de la escena, especialmente cuando se utiliza el ajuste de escala vertical del terreno para exagerar las diferencias de elevación.

  La extensión establecida también se utiliza como extensión de referencia al pulsar el botón  Zoom todo en la vista de mapa 3D.

- Marque  *Mostrar en vista de mapa 2D* para mostrar en el lienzo del mapa principal una banda elástica correspondiente a la extensión actual de la escena 3D.

Terreno

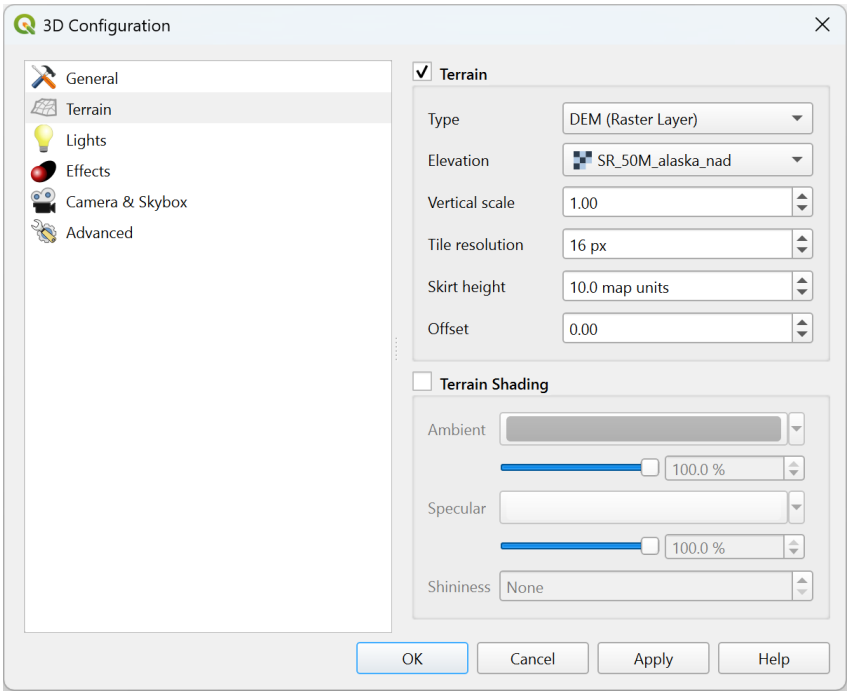


Figura 7.23: El cuadro de diálogo Configuración del terreno del mapa 3D

- *Terreno*: Antes de entrar en mas detalles, vale la pena señalar que el terreno en una vista 3D está representado por una jerarquía de mosaicos de terreno y, a medida que la cámara se acerca al terreno, los mosaicos existentes que no tienen suficientes detalles son reemplazados por mosaicos más pequeños con más detalles. Cada mosaico tiene una geometría de malla derivada de la capa ráster de elevación y la textura de las capas del mapa 2D.
  - El *Tipo* de terreno de elevación puede ser:
    - a *Terreno plano*
    - una *MDE (Capa Ráster)* cargada
    - un servicio *Online*, cargando ` teselas de elevación <http://s3.amazonaws.com/elevation-tiles-prod/>`_ producidas por – mas detalles en https://registry.opendata.aws/terrain-tiles/
    - un conjunto de datos de *Malla*
    - una capa *Malla cuantificada* del proyecto
  - *Elevación*: Ráster o capa de malla a ser usada para la generación del terreno. La capa ráster puede contener una banda que represente la elevación. Para una capa de malla, se usan los valores Z de los vértices.
  - *Escala Vertical*: Facto de escala para eje vertical. Incrementando la escala exagerará la altura de los accidentes geográficos.
  - *Resolución de Teselas*: Cuántas muestras de la capa ráster del terreno se utilizarán para cada mosaico. Un valor de 16 px significa que la geometría de cada mosaico consistirá en 16x16 muestras de elevación. Los números más altos crean mosaicos de terreno más detallados a expensas de una mayor complejidad de representación.
  - *Skirt height*: A veces es posible ver pequeñas grietas entre las baldosas del terreno. Al aumentar este valor, se agregarán paredes verticales («skirts») alrededor de las baldosas del terreno para ocultar las grietas.
  - *Desplazamiento*: desplaza el terreno hacia arriba o hacia abajo, por ejemplo, para ajustar su elevación con respecto al nivel del suelo de otros objetos de la escena.

Esto puede ser útil cuando hay una discrepancia entre la altura del terreno y la altura de las capas en su escena (por ejemplo, nubes de puntos que usan solo una altura vertical relativa). En este caso, ajustar la elevación del terreno manualmente para que coincida con la elevación de los objetos en su escena puede mejorar la experiencia de navegación.

- Cuando una capa de malla se usa como terreno, usted puede configurar los *Ajustes de triángulos* (visualización de la trama, triángulos suaves, nivel de detalle) y los *Ajustes de color de renderización* (como color uniforme o *basado en rampa de color*). Más detalles en la sección *Propiedades de la capa de malla 3D* .
- *Sombreado del Terreno*: Le permite elegir cómo se debe representar el terreno:
  - Sombreado deshabilitado: el color del terreno se determina solo a partir de la textura del mapa
  - Sombreado habilitado: el color del terreno se determina utilizando el modelo de sombreado de Phong, teniendo en cuenta la textura del mapa, el vector normal del terreno, las luces de la escena y el material del terreno *Ambient* y color *Specular* y *Shininess*

Luces

Desde la pestaña *Luces*, presioneel menú para añadir

- hasta ocho *luces puntuales*: emite luz en todas las direcciones, como una esfera de luz rellenando un área. Los objectos cercanos a la luz serán mas brillantes, y los objetos mas alejados estarán mas oscuros. Una luz puntual tiene una posición establecida (*X*, *Y* y *Z*), un *Color*, una *Intensidad* y una *Atenuación*
- hasta cuatro *Luces Direccionales*: imita la iluminación que obtendrías de un flash gigante muy alejado de tus objetos, siempre centrado y que nunca se apaga (por ejemplo, el sol). Emite rayos de luz paralelos en una sola dirección, pero la luz llega al infinito. Una luz direccional se puede girar dado un *Azimuth*, tiene una *Altitud*, un *Color* y una *Intensidad*.

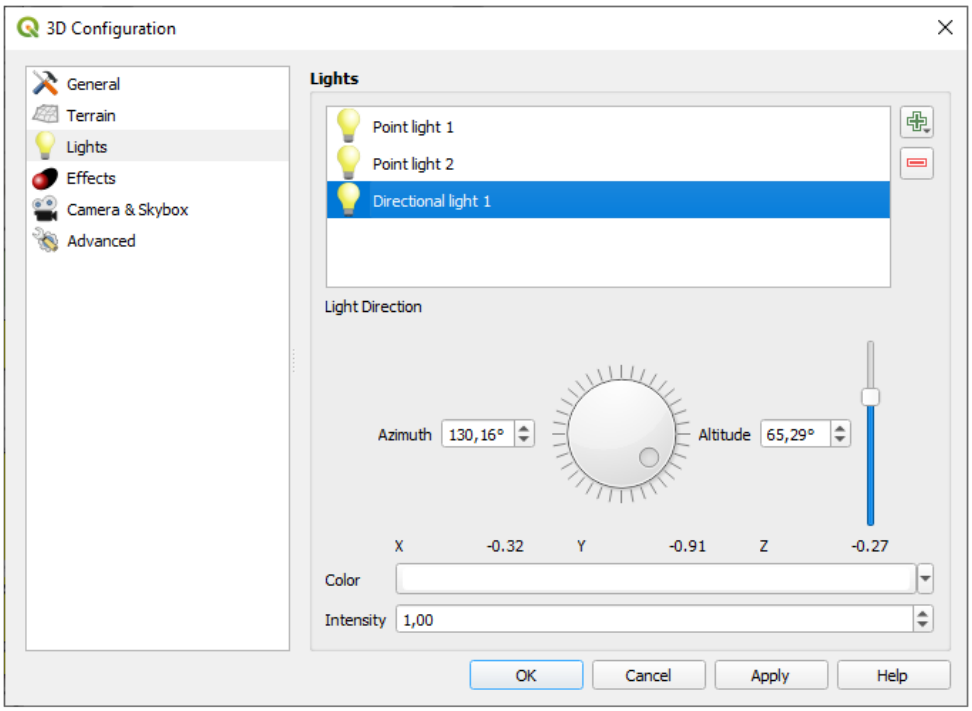


Figura 7.24: El cuadro de diálogo Configuración de luces de mapa 3D

Efectos

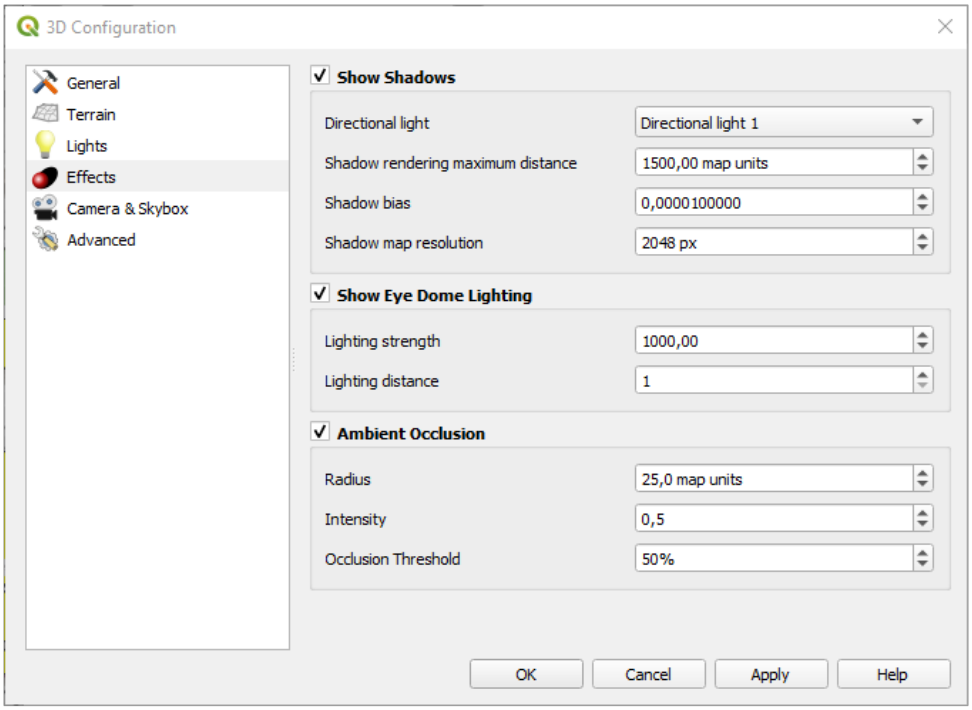


Figura 7.25: El cuadro de diálogo Configuración de efectos del mapa 3D

- Marque *Mostrar sombras* para mostrar sombras dentro de su escena, dada:
  - una *luz Direccional*
  - un *distancia máxima de representación de Sombra*: para evitar la generación de sombras de objetos demasiado distantes, especialmente cuando la cámara mira hacia el horizonte
  - un *Sesgo de sombra*: para evitar efectos de auto-sombreado que podrían oscurecer algunas áreas que otras, debido a las diferencias entre los tamaños de los mapas. Cuanto más bajo mejor
  - una *resolución de Sombra de mapa*: para hacer que las sombras se vean más nítidas. Puede derivar en un menor rendimiento si el parámetro de resolución es demasiado alto.
- :guilabel:»Mostrar iluminación de cúpula ocular» (EDL): efecto de postprocesado que mejora la percepción de la profundidad. La profundidad de cada píxel (distancia de la cámara) se compara con la profundidad de sus píxeles vecinos y se resalta según esa diferencia de profundidad, haciendo que los bordes destaquen. Afecta a toda la escena y puede combinarse con *Oclusión ambiental en el espacio de la pantalla*. Se pueden controlar los siguientes parámetros:
  - *Fuerza de iluminación*: aumenta el contraste, permitiendo una mejor percepción de la profundidad.
  - *Distancia de iluminación*: representa la distancia de los píxeles utilizados respecto al píxel central y tiene el efecto de hacer los bordes más gruesos.
- Añadir espacio en pantalla *Oclusión ambiental* (SSAO): efecto de postprocesado que también mejora la percepción de la profundidad aplicando un sombreado más oscuro a las zonas menos expuestas a la iluminación ambiental. Afecta a toda la escena y puede combinarse con *Iluminación de la cúpula del ojo*. Se pueden controlar los siguientes parámetros:
  - *Radio*: hasta dónde llegaremos para calcular la oclusión ambiental
  - *Intensidad*: intensidad del efecto (los valores más altos oscurecen)

– *Umbral de oclusión*: cuántos puntos vecinos deben estar ocluidos para que aparezca el efecto (valores inferiores al 50% harán que la salida sea más oscura, pero posiblemente proporcionando un mayor rango de oclusión).

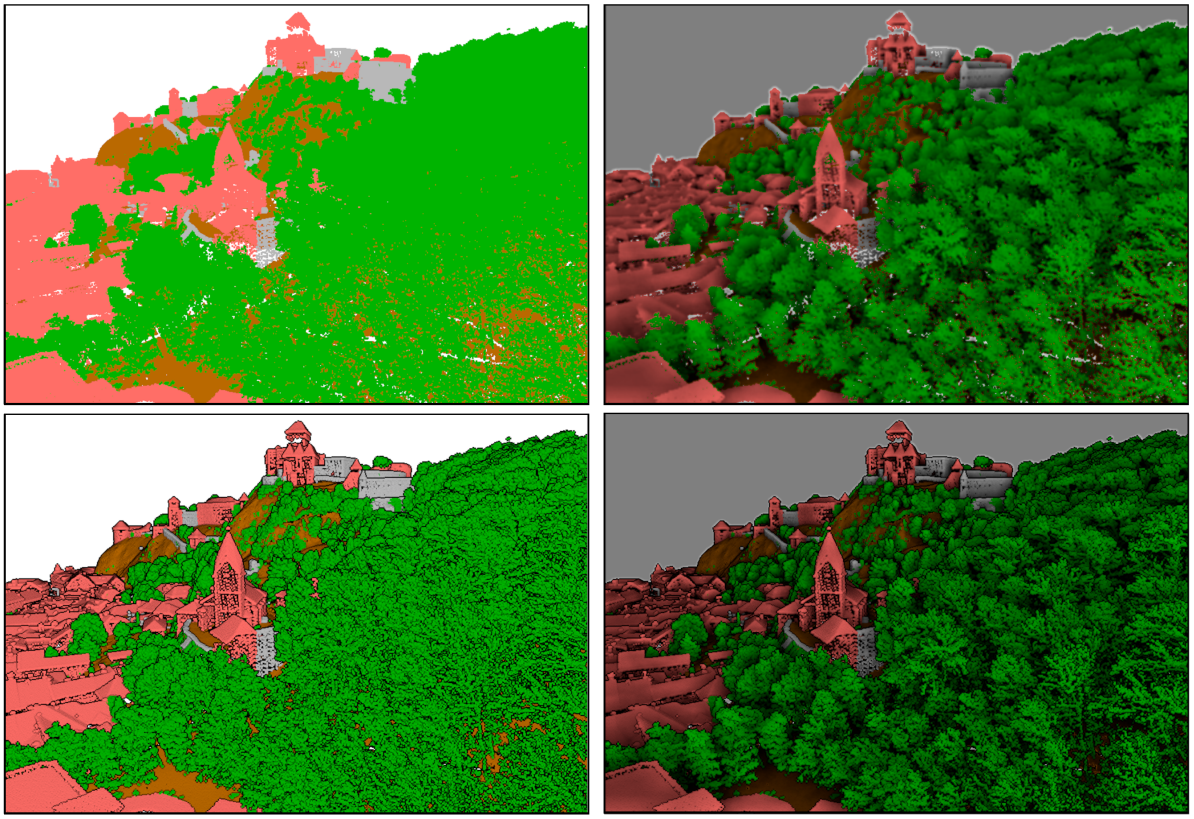


Figura 7.26: Renderizado de nubes de puntos en mapas 3D mediante iluminación Eye Dome (EDL) y oclusión ambiental en el espacio de la pantalla (SSAO)
Desde arriba, de izquierda a derecha: Sin efecto – Sólo SSAO – Sólo EDL – SSAO y EDL

Camera & Skybox

En esta pestaña, puedes controlar diferentes parámetros como la cámara, el eje 3D, la sincronización de la navegación y el skybox.

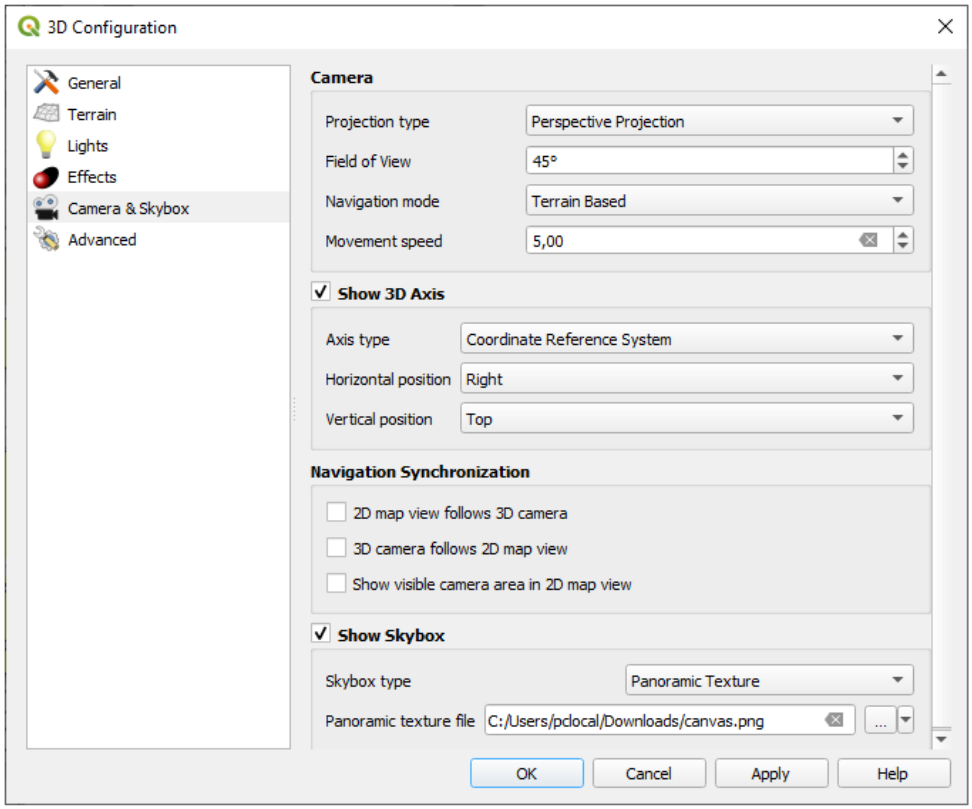


Figura 7.27: El cuadro de diálogo Configuración de la cámara del mapa 3D

- El grupo de parámetros *Cámara* anula algunas *ajustes predeterminados de la cámara* realizados en el diálogo *Configuración* ► *Opciones* ► *3D*.
- Marque ☐ *Mostrar eje 3D* para activar la herramienta de ejes 3D. Este grupo de parámetros permite establecer el tipo de eje y su posición.
  - Con el tipo *Sistema de referencia de coordenadas* se representará un eje ortogonal.
  - Con el tipo *Cubo*, se representará un cubo 3D. Las caras del cubo se pueden utilizar para cambiar la vista de la cámara: por ejemplo, haga clic en la cara *north* para que la cámara vea desde el norte.

Truco

Haga clic con el botón derecho del ratón en el eje 3D para establecer rápidamente su posición y tipo, así como la vista de la cámara.

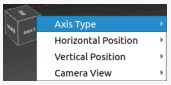


Figura 7.28: El menú contextual Eje 3D

- El grupo de parámetros *Sincronización de la navegación* añade opciones para sincronizar la vista 2D con la posición de la cámara 3D o la posición de la cámara 3D con la vista 2D o la sincronización bidireccional. La última opción muestra la extensión visible desde la cámara 3D sobre la vista del mapa 2D.
- Marque ☐ *Mostrar skybox* para habilitar el renderizado de skybox en la escena. El tipo de skybox puede ser:
  - *textura Panorámica*, con un solo archivo que proporciona visión de 360°
  - *Distintas caras*, con un archivo de textura para cada una de las seis caras de un cubo que contiene la escena

Los archivos de imagen de textura del skybox pueden ser archivos en el disco, direcciones URL remotas o incrustados en el proyecto (*more details*).

Avanzado

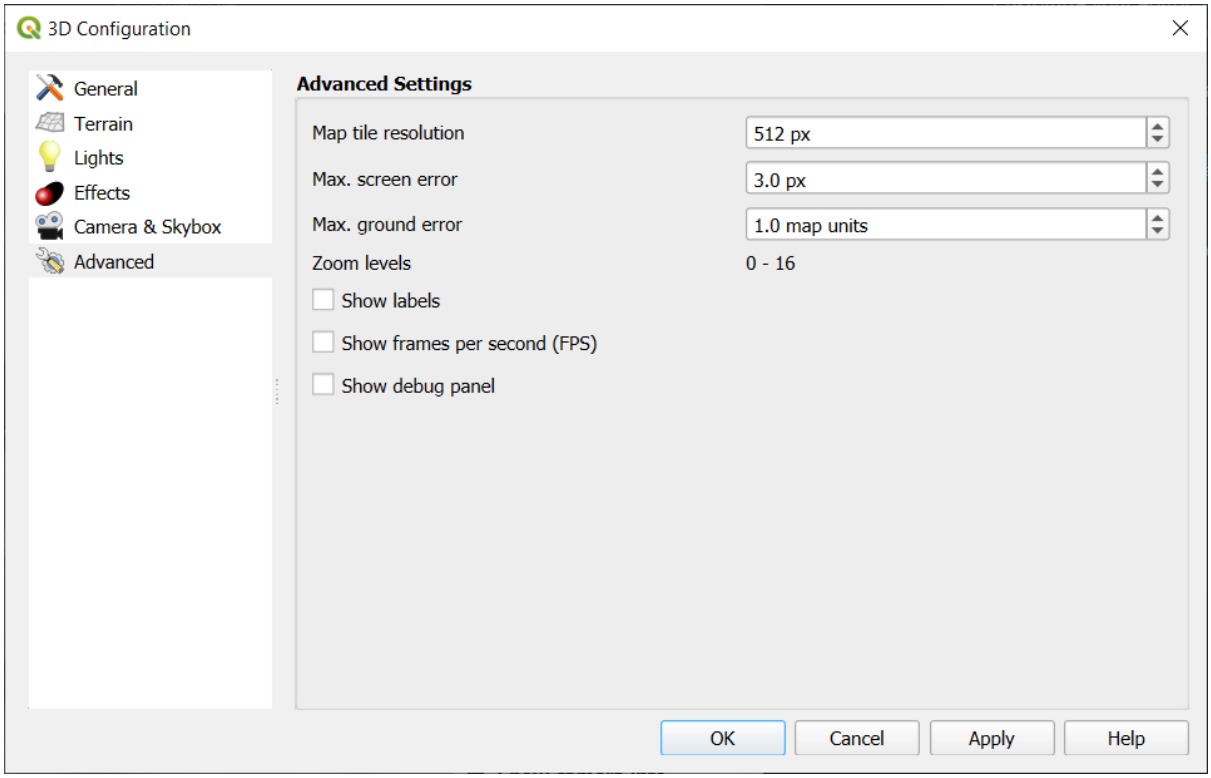


Figura 7.29: El cuadro de diálogo Configuración avanzada del mapa 3D

- *resolución de tesela de mapa*: Ancho y alto de las imágenes del mapa 2D utilizadas como texturas para los mosaicos del terreno. 256 px significa que cada mosaico se renderizará en una imagen de 256x256 píxeles. Los números más altos crean mosaicos de terreno más detallados a expensas de una mayor complejidad de representación.
- *error de pantalla Max.*: Determina el umbral para intercambiar mosaicos de terreno con más detallados (y viceversa), es decir, qué tan pronto la vista 3D utilizará mosaicos de mayor calidad. Los números más bajos significan más detalles en la escena a expensas de una mayor complejidad de representación.
- *Max. ground error*: La resolución de los mosaicos de terreno en los que se detendrá la división de mosaicos en otros más detallados (dividirlos no introduciría ningún detalle adicional de todos modos). Este valor limita la profundidad de la jerarquía de mosaicos: los valores más bajos hacen que la jerarquía sea profunda, lo que aumenta la complejidad de la representación.
- *niveles de Zoom*: Muestra el número de niveles de zoom (depen de la resolución en el mosaico de mapa y error de tierra max.).
- ☐ *Mostrar etiquetas*: Activa/desactiva las etiquetas del mapa
- ☐ *Mostrar imágenes por segundo (FPS)*
- ☐ *Show debug panel*: allows to show or hide a panel next to the 3D map view displaying options to better control map navigation and debug information:
  - ☐ *Show map tile info*: Incluya números de borde y mosaico para los mosaicos de terreno (útil para solucionar problemas de terreno)

- ☐ *Mostrar cuadros delimitadores*: Mostrar cuadros delimitadores 3D de los mosaicos de terreno (útil para solucionar problemas de terreno)
- ☐ *Show camera's view center*
- ☐ *Mostrar el centro de rotación de la cámara*
- ☐ *Mostrar fuentes luminosas*: muestra una esferea en el origen de la fuente de luz, permitiendo fácilmente el reposicionado y emplazamiento de las fuentes de luz relativas al contenido de la escena
- ☐ *Stop scene updates*
- ☐ *Show debug overlay*: visual overlay which displays some useful debugging and profiling information. This allows in particular to quickly see the frame graph and the scene graph
- ☐ *Depurar mapa de sombras*: representa la escena como una imagen rojo-negro desde el punto de vista de la luz utilizada para las sombras (para solucionar problemas). El widget se establece con un *Tamaño* proporcional al de la vista del mapa 3D, y se acopla en una *Esquina*.
- ☐ *Mapa de profundidad de depuración*: representa el mapa de profundidad de la escena como una imagen en la que los píxeles más cercanos son más oscuros (para solucionar problemas). El widget tiene un *Tamaño* proporcional al de la vista del mapa 3D, y se acopla en una *Esquina*.
- *Show camera info*:
  - ☐ *Far plane*: controls how far from the camera rendering stops
  - ☐ *Near plane*: controls how close to the camera rendering starts
  - ☐ *Camera X/Y/Z pos*: sets the camera's position in 3D space
  - ☐ *Looking at X/Y/Z*: sets the target point the camera is looking at

Nota

When your 3D map view is open in a standalone window, you can use `Ctrl+Shift+d` keyboard shortcut to access the debug panel.

7.2.2 Opciones de Navegación

Para explorar la vista de mapa en 3D:

- Incline el terreno (girándolo alrededor de un eje horizontal que atraviesa el centro de la ventana)
  - Presione las herramientas  Tilt up y  Tilt down
  - Presione `Shift` y use las teclas arriba/abajo
  - Arrastre el mouse hacia adelante/atrás con el botón central del ratón presionado
  - Presione `Shift` y arrastre el mouse hacia adelante/atrás con el botón izquierdo del ratón presionado
- Gire el terreno (alrededor de un eje vertical que atraviesa el centro de la ventana)
  - Gire la brújula del widget de navegación hacia la dirección de observación.
  - Presione `Shift` y use las teclas izquierda/derecha
  - Arrastre el mouse hacia la derecha/izquierda con el botón central del ratón presionado
  - Presione `Shift` y arrastre el mouse hacia la derecha/izquierda con el botón iquierdo del ratón presionado
- Cambie la posición de la cámara (y el centro de la vista), moviéndola en un plano horizontal

  - Arrastre el mouse con el botón izquierdo presionado y el  Camera control habilitado
  - Presione las flechas direccionales del widget de navegación
  - Use las teclas arriba/abajo/izquierda/derecha para mover la cámara hacia adelante, atrás, derecha e izquierda, respectivamente.
- Cambiar la altitud de la cámara: presione las teclas `Page Up`/`Page Down`
- Cambie la orientación de la cámara (la cámara se mantiene en su posición pero el punto central de la vista se mueve)
  - Presione `Ctrl` y use las teclas de flecha para girar la cámara hacia arriba, abajo, izquierda y derecha
  - Presione `Ctrl` y arrastre el mouse con el botón izquierdo presionado
- Zoom dentro y fuera
  - Presione las correspondientes herramientas  Zoom In y  Zoom Out del widget de navegación
  - Desplazar la rueda del ratón (mantenga presionada `Ctrl` resulta en zooms mas finos)
  - Arrastre el mouse con el botón derecho presionado para acercar (arrastrar hacia abajo) y alejar (arrastrar hacia arriba)

Para restablecer la vista de la cámara, haga clic en el botón  Zoom Full en la parte superior del panel del lienzo 3D.

7.2.3 Creando una animación

Una animación está basada en un conjunto de keyframes - posiciones de cámara a tiempos particulares. Para crear una animación:

1. Alternar en la herramienta  Animations , mostrando el widget del reproductor de animación
2. Click en el botón  Add keyframe e introduzca en segundos *Keyframe time*. El *Keyframe* combo box ahora muestra el tiempo establecido.
3. Usando las herramientas de navegación, mueva la cámara a la posición a asociar con el tiempo del keyframe actual.
4. Repita los pasos previos para añadir tantos keyframes (con tiempo y posición) como sea necesario.
5. Click en el botón  para previsualizar la animación. QGIS generará escenas utilizando las posiciones / rotaciones de la cámara en los tiempos establecidos, y las interpolará entre estos fotogramas clave. Varios modos *Interpolation* para las animaciones están disponibles (p.ej., linear, inQuad, outQuad, inCirc… – mas detalles en https://doc.qt.io/archives/qt-5.15/qeasingcurve.html#EasingFunction-typedef).

   La animación también se puede previsualizar moviendo el deslizador de tiempo. Si se mantiene marcada la casilla *Ciclo* se ejecutará repetidamente la animación mientras que si se pulsa se detiene una animación en curso.

Click  Export animation frames para generar una serie de imágenes que representan la escena. Aparte del nombre del archivo *Template* y el *Output directory*, puede establecer el número de *Frames per second*, la *Output width* y *Output height*.

7.2.4 Capas vectoriales 3D

Se puede mostrar una capa vectorial con valores de elevación en la vista del mapa 3D marcando *Enable 3D Renderer* en la sección *3D View* de las propiedades de la capa vectorial. Hay varias opciones disponibles para controlar la representación de la capa vectorial 3D.

7.3 Vista del perfil de elevación

El panel *Perfil de elevación* es una herramienta de trazado para vista lateral, para visualizar datos de elevación a lo largo de una línea. Admite capas vectoriales, ráster, de malla y de nube de puntos. Los datos pueden ser de tipo 2D o 3D.

To add an elevation profile view, go to *View* ►  *Elevation Profile* menu. You can add as many profile views as you want, and they can be docked, piled on top of each other, or floating.

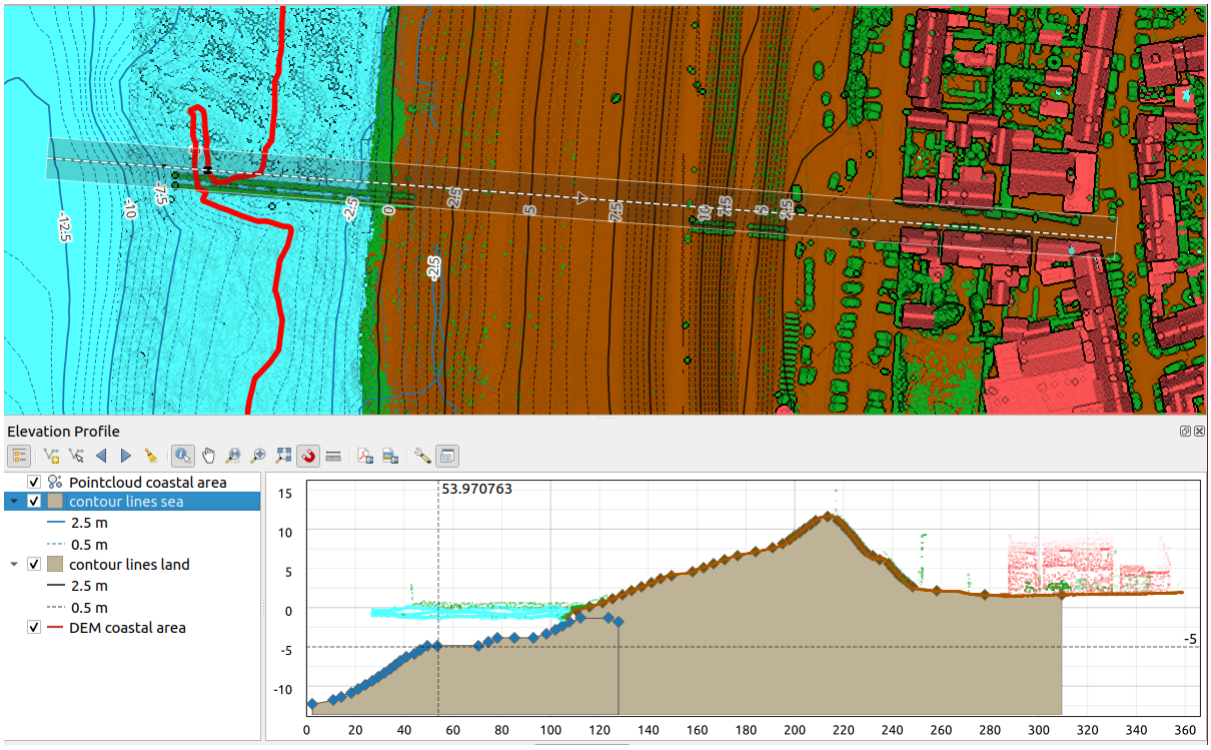


Figura 7.30: Diálogo de perfil de elevación incrustado debajo del lienzo principal del mapa
La figura muestra una costa con una ciudad detrás de un dique. Una tolerancia mayor aplicada a la línea del perfil de elevación devuelve más características de la nube de puntos.

7.3.1 La interfaz

En la parte superior del panel *Perfil de elevación*, una barra de herramientas pone a su disposición las siguientes herramientas:

Tabla 7.5: Barra de herramientas de la vista de perfil de elevación

Herramienta	Atajos	Descripción
 Añadir Capas	mantenga pulsado `Ctrl` mientras arrastra y suelta	Permite añadir nuevas capas a un gráfico.
 Mostrar árbol de capas		Muestra u oculta una lista de capas del proyecto para configurar el renderizado en la vista de perfil.
 Curva de captura		Dibuja interactivamente una línea sobre el lienzo del mapa para representar la curva del perfil.
 Capturar curva a partir de objeto espac		Selecciona una línea existente en el lienzo del mapa y genera una curva de perfil a lo largo de esa línea.
 Empujar a la izquierda	`Ctrl+Alt+,`	Permite desplazar lentamente la línea de captura por el mapa hacia la izquierda (por ejemplo, para encontrar la línea de perfil óptima en función de la elevación).
 Empujar a la derecha	`Ctrl+Alt+.`	Permite desplazar lentamente la línea de captura por el mapa hacia la derecha (por ejemplo, para encontrar la línea de perfil óptima en función de la elevación).
 Limpiar		Elimina la línea de perfil y cualquier gráfico mostrado en la vista *Perfil de elevación*
 Identificar objetos espaciales		Identifica los objetos espaciales en el lienzo del gráfico mediante un solo clic o haciendo clic y arrastrando el rectángulo. Los resultados se muestran en el acople estándar *Identificar resultados*.
 Paneo	`Espacio`	Haga clic y arrastre para desplazar el lienzo. También puede hacerse con el botón central del ratón.
 Zoom Eje X		Acerca/aleja el zoom a lo largo del eje horizontal, manteniendo la relación vertical
 zoom	`Ctrl+Space`	Pulse o haga clic y arrastre un rectángulo sobre el gráfico para ampliarlo. Pulse `Alt` y haga clic para alejar la imagen.
 Zoom Completo		Amplía la vista *Elevation Profile* hasta la extensión de la línea de captura
 Habilitar ajuste		Permite ajustarse al borde o vértice de los objetos espaciales de la parcela en la vista de perfil. Conveniente para la recuperación precisa de coordenadas o mediciones de distancia.
 Medir Distancias		Mide distancias horizontales y verticales
 Exportar como PDF		Exportación de gráficos a PDF (como objetos vectoriales de alta calidad)
 Exportar como imagen		Exporta gráficos a varios formatos de imagen
 Exportar resultados		Exporta gráficos como características 3D, perfil 2D y tabla de distancia/elevación. (Los resultados pueden guardarse como archivos DXF, CSV o cualquiera de los formatos espaciales vectoriales estándar). Más información en *Exportar desde el perfil de elevación*.
 Options ►		Permite acceder a los parámetros de configuración de la línea de elevación del perfil.

continúe en la próxima página

Tabla 7.5 – proviene de la página anterior

Herramienta	Atajos	Descripción
► *Bloquear escalas de distancia/elevación*		Garantiza que las escalas horizontal y vertical se mantengan siempre iguales (de modo que, por ejemplo, una pendiente de 45° aparecerá como una pendiente de 45° en la vista de perfil).
► *Unidades de distancia*		Permite representar las distancias en el gráfico de perfil con unidades distintas de las del lienzo del mapa.
► *Tolerancia*		Sets how far from the actual profile line a feature (vector point, line or polygon, point cloud) can reside to be included in the results. Uses the map units and is ignored by other layer or geometry types. For some layer types (e.g. points, lines, polygons), a custom tolerance can be set in the layer's Elevation properties tab, which replaces the global tolerance and allows more control over feature inclusion based on distance to the profile line.
► *Renombrar Perfil…*		Permite cambiar el nombre de la vista de perfil.
Acoplar vista del perfil de elevación		Cambiar entre el estado anclado y flotante de la vista

En la parte inferior izquierda, se puede mostrar una copia del panel *Capas* pulsando el botón *Mostrar árbol de capas*. Este es, sin embargo, un widget independiente, con su propio conjunto de capas visibles, en un orden de apilamiento personalizado. Le permite controlar la representación y el comportamiento de las capas dentro del lienzo del gráfico:

- Marque la casilla situada junto al nombre de la capa para establecer si debe mostrarse en el lienzo del gráfico.
- Arrastra y suelta capas hacia arriba o hacia abajo para cambiar el orden de las capas
- renderización de estilo de capas en la vista de perfil: haga doble clic en una capa o haga clic con el botón derecho y seleccione *Propiedades…* para abrir la pestaña de propiedades *Elevación* de la capa para su configuración. Al pasar el ratón por encima de la capa, se muestra un resumen de los ajustes de elevación.

A la derecha del árbol de capas, el lienzo de trazado es el lugar principal donde se puede previsualizar el perfil de elevación de las capas habilitadas. Se basa en una cuadrícula graduada en la que el eje horizontal muestra la longitud de la línea de perfil y el eje vertical muestra la elevación Z de las características observadas. También permite una serie de interacciones como hacer zoom, desplazarse, medir, identificar características, … utilizando las herramientas de la parte superior.

7.3.2 Creando un perfil de elevación

Para crear un perfil, puedes:

1. Vaya al menú *Ver* ► *Perfil de elevación*. Se abre el panel *Perfil de elevación*.
2. Cree la línea de perfil a lo largo de la cual se renderizarán el terreno y los elementos. Seleccione una herramienta de dibujo:
   - Capturar Curva: pulsa a la izquierda sobre el lienzo principal del mapa para añadir vértices y pulsa a la derecha para finalizar una línea que se usará como línea de perfil
   - o Capturar curva a partir de objeto espacial: haga clic en una característica lineal del lienzo del mapa para seleccionarla como línea de perfil. Si hay varias características presentes en el punto sobre el que se ha hecho clic, aparecerá un menú emergente que le permitirá seleccionar una de ellas.

   Todas las funciones de digitalización de líneas, como las opciones de *ajuste*, *trazado*, *técnicas de digitalización* o el panel de *panel de digitalización* avanzado están disponibles para su uso.

El lienzo de trazado puede empezar a renderizar algunos objetos espaciales.

3. El próximo paso es configurar las propiedades de elevación de las capas que quiere visualizar.

   1. Pulse el botón  Mostrar árbol de capas para mostrar la lista de capas. Predeterminado, no todas las capas del proyecto son cargadas y referenciadas en la herramienta de perfil; por ejemplo, las capas ráster que no están marcadas como que tienen datos de elevación en sus propiedades son ignoradas. Al hacer clic en el botón  Añadir capas se mostrará una lista filtrada de posibles capas que se pueden añadir al trazado, pero que actualmente no están en el trazado. Aplicando el diálogo con las capas seleccionadas se marcará automáticamente que tienen datos de elevación y se añadirán inmediatamente al gráfico.

      Truco

      Manteniendo pulsada la tecla `Ctrl`, es posible arrastrar y soltar capas adicionales desde el árbol de capas principal a una en la herramienta de perfil de elevación.

   2. Alterne la visibilidad de las capas que le interesan. Estas son las únicas que se muestran en la vista de perfil y las capas seleccionadas pueden ser diferentes de las del panel principal *Capas*.
   3. Haga doble clic en el nombre de una capa o haga clic con el botón derecho y seleccione *Propiedades*. Se abre la pestaña *Propiedades de elevación* de la capa. Este es el lugar donde se configura cómo debe aparecer cada elemento o terreno en la vista de perfil. Las propiedades disponibles dependen del tipo de capa:

      - *Propiedades de elevación raster*
      - *Propiedades de elevación del vector*
      - *Propiedades de elevación de la nube de puntos*
      - *Propiedades de elevación de la malla*

      La vista de perfil comienza a renderizar el terreno o los objetos espaciales de las capas activas que cruzan la curva de perfil dada, tan pronto como tengan configuradas las propiedades de elevación.

4. Under  Options drop-down menu, you can:

   - set the *Tolerance* value. This value is used to create a flat buffer around the elevation profile line, visible in the main map canvas. Any visible feature (point, line, polygon, etc.) overlapping that buffer will be captured in the plot canvas.
   - select the  *Show Subsections Indicator* to display vertical lines at the distance of the vertices of the profile curve. This highlights the position of the vertices of the curve on the Elevation Profile.
   - select the *Subsections Symbology* to set the color, opacity, width and style of the Subsections Indicator lines.

Nota

Limitaciones con la extrusión de polígonos

La extrusión de geometría puede establecerse en las propiedades  *Elevación* de una capa, y renderizarse en la vista de perfil. Cuando la tolerancia está activada, no es trivial renderizar polígonos extruidos, por lo que, por ahora, la extrusión de polígonos se ignora.

7.3.3 Interactuar con el perfil Vistas

Cuando se crea una línea de perfil de elevación, el lienzo del gráfico se amplía en toda su extensión. En el eje X se puede ver la longitud del perfil y, en el eje Y el rango de altura entre la altura mínima y máxima capturada, ambas en unidades de mapa.

Cuando mueva el puntero del ratón en la vista del perfil de elevación, verá dos líneas de puntos que se cruzan:

- la linea vertical muestra la información de la altura
- la línea horizontal muestra la distancia desde el inicio del perfil de elevación Línea

Cuando mueva el puntero del ratón en el lienzo del perfil de elevación, también podrá ver cómo se desplaza un punto negro a lo largo de la línea del perfil de elevación en el lienzo principal del mapa. En el centro de la línea del perfil, una flecha indica su dirección.

En cuanto al lienzo principal del mapa, QGIS proporciona medios para navegar en el lienzo de la parcela:

- Desplazar se utiliza para mover la extensión del perfil de elevación en cualquier dirección que desee. Manteniendo pulsada la tecla `Espacio` mientras se mueve el ratón también se desplaza la extensión del lienzo.
- Ampliar Eje X se utiliza para ampliar a lo largo del eje horizontal, manteniendo la misma relación del eje vertical (la elevación). Haga clic con el botón izquierdo del ratón para ampliar el gráfico a lo largo del eje X, con el punto pulsado en el centro del eje, o arrastre un rectángulo para ampliar el gráfico a lo largo del eje X hasta la anchura del rectángulo. Mantenga pulsado `Alt` mientras utiliza Zoom X Axis para alejar el zoom a lo largo de ese eje.
- Zoom se utiliza para hacer zoom en un punto determinado (haciendo clic con el botón izquierdo), o hasta cierto punto (arrastrando un rectángulo sobre el área). Mantenga pulsado `Alt` mientras utiliza Zoom para alejar la imagen. En combinación con la tecla `Ctrl` puede acercar o alejar el zoom más suavemente.
- Acercamiento completo es el nivel de zoom por defecto utilizado al principio y muestra la extensión completa de la línea de perfil, con todas los objetos espaciales devueltas. Utilícelo para restablecer el nivel de zoom.

También es posible interactuar con los elementos que aparecen en el lienzo del gráfico:

- Pulse el botón Habilitar ajuste para capturar con precisión los puntos, vértices o aristas de las características, para obtener una medición precisa o un informe de coordenadas.
- Identificar objeto espaciales se utiliza para identificar objetos espaciales de las capas visibles en el árbol de capas. Puede arrastrar un rectángulo sobre varios objetos espaciales en la vista de perfil para consultarlas todas. Cuando sea compatible con el formato (por ejemplo, vectorial, nube de puntos), estas características se resaltarán en el lienzo principal del mapa.
- Distancias de medición: haga clic o seleccione dos puntos en el lienzo del gráfico para informar de la *Distancia* horizontal, la *Elevación* y la *Distancia total* entre ellos, en unidades de mapa.
- Desplazar a la izquierda y Desplazar a la derecha se utilizan para desplazar la posición de la línea de perfil de elevación en el lienzo de mapa hacia su izquierda o derecha. El lienzo del mapa se redibujará, mostrando las características y el terreno que se superponen al búfer de la línea de perfil. La línea se desplaza lateralmente utilizando el valor *Tolerancia* del menú Opciones.

> **Advertencia**
>
> Al cerrar una vista de perfil de elevación o el proyecto actual se elimina la vista del proyecto.

¡Para más detalles, consulte QGIS elevation profile/cross section tool – a deep dive!, una presentación realizada por Nyall Dawson.

7.3.4 Exportar desde el perfil de elevación

La barra de herramientas superior del cuadro de diálogo *Perfil de elevación* ofrece una variedad de herramientas para exportar un perfil de elevación, lo que le permite reutilizar los gráficos en otro software:

-  Exportar como PDF: Exporta los gráficos a PDF (como objetos vectoriales de alta calidad). Se solicitan las siguientes opciones para generar el archivo de salida:
  - La etiqueta *Tamaño de página*, seleccionando una página de una lista predefinida o estableciendo dimensiones personalizadas, y la orientación de la página.
  - El *Rangos del gráfico* para controlar la extensión del gráfico que se va a exportar:
    - en el eje X, la *Distancia mínima* y la *Distancia máxima* desde el punto inicial de la curva de perfil.
    - En el eje Y, la *Elevación mínima* y la *Elevación máxima*.
  - La opción *Configuración del eje de distancia*: le ayuda a configurar la graduación horizontal de la cuadrícula sobre la que se representa el perfil de elevación. Puede personalizar el *Intervalo de etiquetas*, el *Intervalo de líneas de cuadrícula principales* y el *Intervalo de líneas de cuadrícula secundarias*.
  - La etiqueta *Configuración del eje de elevación*, igual que arriba, para la graduación vertical.
-  Exportar como imagen: Exporta los gráficos a un formato de archivo de imagen. Se solicitan opciones de exportación específicas, como el ancho y el alto de la imagen, junto con los rangos del gráfico, la configuración del eje de distancia y la configuración del eje de elevación mencionados anteriormente.
-  Exportar resultados: Exporta los resultados del perfil como un conjunto de características de una capa vectorial (`.DXF`, `.CSV`, `.SHP`, `.GPKG`, ...). Dependiendo del formato de salida, se pueden devolver varias capas cuando los tipos de geometría de las características exportadas difieren.
  - *Exportar objetos espaciales 3D*: Exporta las líneas de perfil como características 3D, con valores de elevación tomados de los cortes de elevación y almacenados en los valores Z de la geometría exportada.
  - *Exportar perfil 2D*: Exporta los perfiles como líneas de perfil 2D, con la elevación almacenada en la dimensión Y de la geometría exportada y la distancia en la dimensión X (tal y como se muestra en el widget del perfil de elevación).
  - *Exportar tabla de distancia/elevación*: Exporta perfiles como una tabla de valores de distancia y elevación muestreados.

CAPÍTULO 8

Herramientas generales

8.1 Ayuda de contexto

Siempre que necesite ayuda sobre un tema específico, puede acceder a la página correspondiente en el Manual del usuario actual a través del botón *Ayuda* disponible en la mayoría de los cuadros de diálogo — tenga en cuenta que los complementos de terceros pueden apuntar a páginas web dedicadas.

8.2 Paneles

Por defecto, QGIS proporciona muchos paneles para trabajar. Algunos de estos paneles se describen a continuación, mientras que otros se pueden encontrar en diferentes partes del documento. Una lista completa de paneles predeterminados proporcionados por QGIS está disponible a través de *View* ► *Panels* ► menu and mentioned at *Paneles*.

8.2.1 Panel de capas

El panel *Capas* (también llamado *leyenda del mapa*) lista todas las capas del proyecto y le ayuda a gestionar su visibilidad y a dar forma al mapa. Puede mostrar u ocultar el panel pulsando `Ctrl+1`.

QGIS ofrece varias formas de añadir capas a un proyecto:

- utilizando el botón *Añadir* de la pestaña dedicada al proveedor de datos en el diálogo *Administrador de fuentes de datos*
- de QGIS *Panel del navegador* o *Administrador de BD*: haga doble clic, arrastre y suelte archivos y capas en QGIS o utilice el menú contextual
- arrastrar y soltar archivos desde el explorador de archivos del sistema operativo a QGIS

En todos estos escenarios, puede abrir una o varias capas a la vez. Las nuevas capas se añaden al panel *Capas*:

1. si se sueltan sobre el panel *Capas*, en el lugar exacto donde se sueltan
2. en los demás casos, en un lugar que respete los ajustes globales *comportamiento utilizado al añadir nuevas capas*
3. y en caso de capas múltiples, se ordenan de forma que aumente la probabilidad de que su apilamiento sea lógico y las características sean visibles el mayor tiempo posible, utilizando la siguiente lógica (de arriba a abajo):

- capas de puntos vectoriales
- capas de líneas vectoriales
- capas de polígonos vectoriales
- capas de nubes de puntos
- capas de malla
- capas ráster

En la parte superior del panel *Capas*, una barra de herramientas le permite:

- Abrir el panel de diseño de capas (F7): conmuta el panel *Estilo de Capa* activo e inactivo.
- Add new group: ver *Interactuar con grupos y capas*
- Administrar Temas de Mapas: controla la visibilidad de las capas y su organización en diferentes *temas de mapa*.
- filtrar capas en el árbol de leyenda:
  - *Filter Legend by Map Content*: solo las capas que están configuradas como visibles y cuyas características se cruzan con el lienzo del mapa actual tienen su estilo representado en el panel de capas. De lo contrario, se aplica un símbolo NULL genérico a la capa. Basándose en la simbología de la capa, esta es una forma conveniente de identificar qué tipo de características de qué capas cubren su área de interés.
  - *Show Private Layers*: un atajo conveniente para mostrar e interactuar con *private layers* en el panel *Layers* sin modificar la configuración del proyecto.
  - *Mostrar sólo capas rotas*: sólo se muestran las capas con fuentes de datos rotas.
- Filtrar Leyenda por Expresión: aplica una expresión para eliminar estilos del árbol de capas seleccionado que no tienen ninguna entidad que satisfaga la condición. Esto se puede usar para resaltar entidades que están dentro de un área/entidad dada de otra capa. En la lista desplegable, puede editar y borrar la expresión actualmente aplicada.
- Expandir Todo o Colapsar >Todo capas y Grupo en el panel de capas.
- Borrar Capa/Grupo actualmente seleccionada.

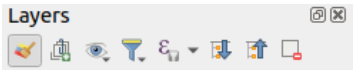


Figura 8.1: Barra de herramientas de capa en panel de capas

Nota

Herramientas para Administrar el panel de Capas están también disponibles para elementos de mapa y leyenda en el diseñador de impresión..

Configurando temas de mapa

El botón desplegable Administrar Temas de Mapa dá acceso a convenientes atajos para manipular la visibilidad de las capas en el panel *Capas*:

- *Mostrar todas las capas*
- *Ocultar todas las capas*
- *Mostrar Capas Seleccionadas*
- *Ocultar Capas Seleccionadas*
- *Conmutar Capas Seleccionadas*: cambia la visibilidad de la primera capa seleccionada en el panel y aplica ese estado a las otras capas seleccionadas. También accesible a través del atajo `Space`.
- *Conmutar Capas Seleccionadas Independientemente*: cambia el estado de visibilidad de cada capa seleccionada
- *Ocultar Capas Deseleccionadas*

Más allá del simple control de la visibilidad de la capa, el menú :sup:`Administrar temas de mapa" le permite configurar **Temas de mapa** en la leyenda y cambiar de un tema de mapa a otro. Un tema de mapa es una **instantánea** de la leyenda del mapa actual que registra:

- las capas establecidas como visibles en el panel :guilabel:"Capas"
- **y** para cada capa visible:
  - la referencia al *style* aplicado a la capa
  - las clases visibles del estilo, p.ej. los elementos de la capa marcada en el *Panel de Capas*. Esto aplica a *symbologies* otra que no sea la representación de un solo símbolo
  - el estado colapsado/expandido de los nodos de capa y los grupos que se encuentran dentro

Para crear un tema de mapa:

1. Marca una capa que quiera mostrar
2. Configure las propiedades de capa (simbología, diagrama, etiquetas…) como siempre
3. Expanda el menú *Estilo* ► al fondo y haga click en *Añadir…* para almacenar los ajustes como *a new style embedded in the project*

   > **Nota**
   >
   > Un tema de mapa no recuerda los detalles actuales de las propiedades: solo se guarda una referencia al nombre del estilo, por lo que cada vez que aplica modificaciones a la capa mientras este estilo está habilitado (por ejemplo, cambie la representación de la simbología), el tema del mapa se actualiza con nueva información.

4. Repetir los pasos previos como sea necesario para las otras capas
5. si es aplicable, expanda o colapse grupos o nodos de la capa visible en el panel *Capas*
6. Click en el botón Administrar Temas de Mapa en la parte superior del panel, y *Añadir Tema…*
7. Introduzca el nombre del tema del mapa y haga click en :guilabel:"OK"

El nuevo tema se listará en la parte inferior del menú desplegable .

Puede crear tantos temas de mapa como necesite: siempre que la combinación actual en la leyenda del mapa (capas visibles, su estilo activo, los nodos de leyenda del mapa) no coincida con ningún contenido de tema de mapa existente como se define arriba, haga clic en *Agregar Tema…* para crear un nuevo tema de mapa, o use *Reemplazar Tema* ►

para actualizar un tema de mapa. Puede renombrar el tema de mapa activo con *Renombrar Tema Actual…* o usando el botón *Borrar Tema Actual* para borrarlo.

Los temas de mapas son útiles para cambiar rápidamente entre diferentes combinaciones preconfiguradas: seleccione un tema de mapa en la lista para restaurar su combinación. Todos los temas configurados también son accesibles en el diseño de impresión, lo que le permite crear diferentes elementos de mapa basados en temas específicos e independientes de la representación del lienzo principal actual (consulte *Map item layers*).

Repaso del menú contextual del panel de Capas

En la parte inferior de la barras de herramientas el componente principal del panel Capas es el marco donde se listan todas las capas añadidas al proyecto organizadas, opcionalmente, en grupos. Una capa con la caja de confirmación marcada muestra su contenido superpuesto a la extensión del lienzo del mapa, a no ser que se establezca una:ref:*visibilidad en base a la escala <label_scaledepend>* . Se puede seleccionar una capa y arrastrarla arriba o abajo en la leyenda para cambiar la ordenación Z. La ordenación Z quiere decir que las capas listadas más cerca de la parte superior de la lista se dibujan sobre las capas que se listan más abajo. Asimismo una capa o grupo de capas puede ser arrastrado a través de diversas instancias QGIS.

> **Nota**
>
> El comportamiento de ordenamiento Z puede ser anulado por el *Layer Order* panel.

Depending on the item selected in the panel, a right-click shows a dedicated set of options presented below. Some of these entries are not available when multiple selections are made (e.g., *Filter*, *Rename*, *Properties*).

Tabla 8.1: Menús contextuales de elementos del panel *Capas*

Opción	Grupo	Capa vectorial	Capa Ráster	Capa de malla	Capa de nubes de puntos	Capa 3D
Zoom a Capa(s)/Grupo	✓	✓	✓	✓	✓	✓
Zoom a Selección		✓				
Mostrar en Resumen		✓	✓	✓	✓	✓
Mostrar Recuento de Entidades		✓				
Mostrar etiqueta		✓		✓		
Copiar Capa/Grupo	✓	✓	✓	✓	✓	✓
Renombrar Capa/Grupo	✓	✓	✓	✓	✓	✓
Zoom a la Resolución Nativa (100%)			✓			
Estrechar usando la extensión actual			✓			
Create Raster Attribute Table			✓			
Open Raster Attribute Table			✓			
Load Raster Attribute Table from VAT.DBF			✓			
Actualizar Capa SQL…		✓				
Execute SQL…		✓				
Editar Capa Virtual…		✓				

continúe en la próxima página

Tabla 8.1 – proviene de la página anterior

Opción	Grupo	Capa vectorial	Capa Ráster	Capa de malla	Capa de nubes de puntos	Capa 3D
Añadir Grupo	✓					
Duplicar Capa		✓	✓	✓	✓	✓
Borrar Capa/Grupo…	✓	✓	✓	✓	✓	✓
Sacar del Grupo		✓	✓	✓	✓	
Mover hasta el tope	✓	✓	✓	✓	✓	
Mover al Fondo	✓	✓	✓	✓	✓	
Marcar y todos sus padres		✓	✓	✓	✓	
Grupo Seleccionado		✓	✓	✓	✓	
Abrir tabla de atributos		✓				
Conmutar edición		✓		✓		
Ediciones actuales ►		✓		✓		
Filtrar…		✓	✓		✓	
Cambiar Fuente de Datos…		✓	✓	✓	✓	
Reparar Fuente de Datos…		✓	✓	✓	✓	
Acciones en Selecciones ► (en modo edición)		✓				
► *Duplicar Objeto*		✓				
► *Duplicar Objeto y Digitalizar*		✓				
Establecer la Visibilidad de la Capa…		✓	✓	✓	✓	✓
Zoom a Escala Visible		✓	✓	✓	✓	
SRC de capa ►		✓	✓	✓	✓	✓
► *Set Project CRS from Layer*		✓	✓	✓	✓	✓
► *Definir en..* (SRCs recientes)				✓	✓	✓
► *Establecer SRC de capa…*		✓	✓	✓	✓	✓
Establecer SRC de Grupo…	✓					
Establecer Grupo de Datos WMS…	✓					
Grupo Mutuamente Exclusivo	✓					
Marcar y todos sus hijos (Ctrl-click)	✓					
Desmarcar y todos sus hijos (Ctrl-click)	✓					
Hacer Permanente		✓				
Exportar ►	✓	✓	✓	✓	✓	✓
► *Guardar Como…*			✓			
► *Guardar Objetos como…*		✓				
► *Guardar Objetos Seleccionados como…*		✓				

continúe en la próxima página

Tabla 8.1 – proviene de la página anterior

Opción	Grupo	Capa vectorial	Capa Ráster	Capa de malla	Capa de nubes de puntos	Capa 3D
► *Guardar como Archivo de Definición de Capa…*	☑	☑	☑	☑	☑	☑
► *Guardar como Archivo de Estilo de Capa de QGIS…*		☑	☑	☑	☑	☑
Estilos ►		☑	☑	☑	☑	☑
► *Copiar Estilo*		☑	☑	☑	☑	☑
► *Pegar Estilo*	☑	☑	☑	☑	☑	
► *Agregar…*		☑	☑	☑	☑	☑
► *Renombrar Actual…*		☑	☑	☑	☑	☑
► *Editar símbolo…*		☑				
► *Copiar Símbolo*		☑				
► *Pegar Símbolo*		☑				
Añadir notas de capa…		☑	☑	☑	☑	☑
Editar notas de capa…		☑	☑	☑	☑	
Eliminar notas de capa		☑	☑	☑	☑	
Propiedades…		☑	☑	☑	☑	☑

Para las capas vectoriales de GRASS, Toggle editing no está disponible. Ver la sección *Digitalizar y editar una capa vectorial GRASS* para información de la edición de capas vectoriales de GRASS.

Interactuar con grupos y capas

Las capas de la ventana de leyenda pueden organizarse en grupos. Hay diferentes maneras de hacerlo:

1. Presione el icono para añadir un nuevo grupo. Escriba un nombre para el grupo y presione `Enter`. Ahora haga click en una capa existente y arrástrela al grupo.
2. Seleccione más de una capa y pulse el icono **|carpeta|**. Las capas seleccionadas se insertan automáticamente en el nuevo grupo.
3. Seleccione algunas capas, haga click derecho en la ventana de la leyenda y elija *Grupo seleccionado*. Las capas seleccionadas se colocarán automáticamente en un nuevo grupo.

Para mover una capa fuera de un grupo, arrástrela o haga click derecho sobre ella y elija *Mover fuera del grupo*: la capa se mueve del grupo y se coloca encima de ella. Los grupos también se pueden anidar dentro de otros grupos. Si una capa se coloca en un grupo anidado, *Mover fuera del grupo* moverá la capa fuera de todos los grupos anidados.

Para mover un grupo o capa a la cumbre del panel de capas, o bien arrástrela a la parte superior, o elija *Mover Arriba*. Si usa está opción en una capa anidad en un grupo, la capa es movida a la cúspide del grupo actual. La opción *MOver abajo* sigue la misma lógica para mover capas y grupos abajo.

La casilla de verificación para un un grupo mostrará u ocultará las capas marcadas en el grupo con un click. Con `Ctrl` presionado, la casilla de verificación activará o desactivará todas las capas en el grupo y sus subgrupos..

`Ctrl`-click en una capa marcada/desmarcada desmarcará/marcará la capa y todos sus padres.

Habilitar la opción **Grupo mutuamente exclusivo** significa que puede hacer que un grupo tenga solo una capa visible al mismo tiempo. Siempre que una capa dentro del grupo sea visible, las demás se alternarán como no visibles.

Es posible seleccionar más de una capa o grupo al mismo tiempo manteniendo presionada la tecla `Ctrl` mientras hace click en capas adicionales. Luego puede mover todas las capas seleccionadas a un nuevo grupo al mismo tiempo.

También puede eliminar más de una capa o grupo a la vez seleccionando varios elementos con la tecla `Ctrl` y luego presionando `Ctrl + D`: todas las capas o grupos seleccionados se eliminarán de la lista de capas.

Mas información de capas y grupos usando el icono indicador

En algunas circunstancias, el formato o los iconos junto a la capa o grupo en el panel *Capas* cambian para dar más información sobre la capa/grupo. Estos elementos son:

-  para indicar que la capa está en modo edición y puede modificar los datos
-  para indicar que la capa que se está editando tiene algunos cambios sin guardar
-  indicar *a filter* aplicado a la capa. Pase el cursor sobre el icono para ver la expresión de filtro y haga doble clic para actualizar la consulta
-  para identificar las capas que son *requeridas* en el proyecto, por lo tanto no removibles
-  para identificar en *embedded group or layer* y la ruta a su archivo de proyecto original
-  para identificar una capa cuya fuente datos no estaba disponible en el archivo de apertura del proyecto (ver *Manejo de rutas de archivo rotas*). Click en el icono para actualizar la ruta de la fuente o seleccione la entrada *Reparar Fuente de Datos…* del menú contextual de capa.
-  para recordarle que la capa es una *capa de borrador temporal* y su contenido será descartado cuando cierre este proyecto. Para evitar la perdida de datos yh hacer la capa permanente, pulse el icono para almacenar la capa en cualquiera de los formatos vectoriales GDAL soportados por QGIS.
-  para identificar una capa utilizada en *offline editing mode*.
-  para identificar una capa que no tiene un CRS o es desconocido
-  para capas con coordenadas almacenadas en un sistema de referencia de coordenadas que es inherentemente de baja precisión (requiere que la *corresponding setting* esté habilitada)
-  para identificar una capa temporal controlada por la animación del lienzo
-  para identificar una capa que tiene *notes* asociada
- Un nombre en gris, cuando la escala actual del lienzo del mapa está fuera del rango de escala de visibilidad de la capa (como se establece en sus propiedades *Renderizado*). Seleccione la opción *Zoom a escala visible* del menú contextual para ampliar el mapa hasta el límite de escala de visibilidad más cercano de la capa.

Control de la representación de capas mediante agrupación

Los grupos son un medio de estructurar las capas dentro de un árbol en el proyecto, pero también pueden influir en el modo en que se renderizan las capas que los componen, es decir, como un único objeto aplanado durante el renderizado de mapas.

La opción para dicho renderizado está disponible dentro del panel *Estilización de capas* siempre que se seleccione un grupo. En la pestaña  Simbología, marque la casilla  *Renderizar capas como grupop* para activar un conjunto de opciones para controlar la apariencia de las capas hijas como un todo, en lugar de capas individuales:

- *Opacidad*: Los objetos espaciales de las capas secundarias que están oscurecidas por otras capas secundarias permanecen oscurecidas, y la opacidad se aplica sólo a «todo el grupo».

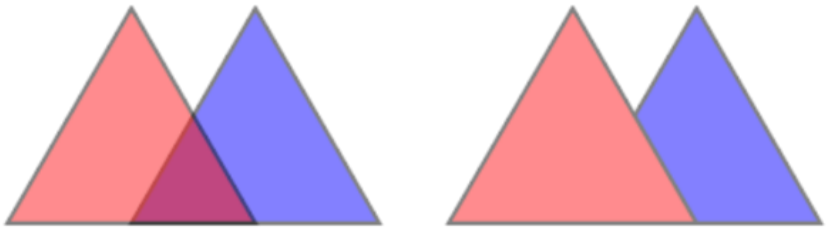


Figura 8.2: Establecer opacidad en capas frente a un grupo
La imagen de la izquierda muestra dos capas renderizadas al 50% de opacidad (los objetos espaciales subyacentes son visibles, pero semiocultos por el rasgo rojo al 50% situado encima). La segunda imagen muestra el resultado de ajustar la opacidad en el grupo (partes de la capa azul subyacente hijo está completamente oscurecida por la capa roja en la parte superior y luego el resultado se renderiza al 50% de opacidad).

- *Modos de mezcla*: Al igual que opacidad configurando un *modo de mezcla* (como multiplicar, superponer, …) para un grupo completo da como resultado primero aplanar las características de las capas secundarias, con las superiores oscureciendo las más bajas. La representación se obtiene combinando el grupo plano y las capas que se encuentran debajo del grupo.
  - Cuando las capas hijas tienen modos de fusión asignados, se aplica antes de aplanar, pero el alcance se restringe a afectar sólo a otras capas hijas de ese grupo, y no a otras capas situadas por debajo de todo el grupo.
  - Algunas opciones más de *modos de mezcla* están disponibles para las capas secundarias en los grupo, en su pestaña *Simbología* que lleva a cabo operaciones de estilo «recorte» sobre otras capas secundarias durante la representación. Usted puede, p.ej. recortar la representación del contenido de una capa con el contenido de una segunda que actúa como capa «máscara».
- *Efectos de capa*: aplica *efectos* solo a la representación aplanada de las capas secundarias; de este modo, p.ej. So e.g. un efecto de sombra paralela aplicado al grupo no sería visible para las capas secundarias oscurecidas.

Cuando un grupo se establece en *Renderizar capas como grupo*, sólo se mostrará el grupo en la lista del panel *Orden de capas*. Los hijos del grupo no serán visibles en esta lista de orden, ya que su orden viene determinado por la colocación de la capa del grupo.

Editando estilo de capa

Desde el panel *Capas*, dispone de accesos directos para cambiar el renderizado de las capas de forma rápida y sencilla.

Haga clic con el botón derecho en una capa y seleccione *Estilos* ► en la lista para:

- véase la *estilos* actualmente disponible para la capa. Si ha definido muchos estilos para la capa, puede cambiar de uno a otro y la representación de la capa se actualizará automáticamente en el lienzo del mapa.
- copiar parte o todo el estilo actual, y cuando sea aplicable, pegar un estilo copiado desde otra capa
- *Renombrar actual…* estilo
- *Añadir* un nuevo estilo (que en realidad es una copia del actual)
- o *Quitar estilo actual* (sólo cuando hay varios estilos disponibles).

> **Truco**
>
> **Compartir ágilmente un estilo de capa**
>
> En el menú contextual, copie el estilo de una capa y péguelo en un grupo o en una selección de capas: el estilo se aplica a todas las capas que sean del mismo tipo (vectorial, ráster, malla, nube de puntos, …) que la capa original y, en el caso de las capas vectoriales, que tengan el mismo tipo de geometría (punto, línea o polígono).

Cuando se utiliza una simbología basada en la clasificación de objetos espaciales (por ejemplo, *categorized*, *graduated* or *rule-based* para capas vectoriales, o *clasificación* para nubes de puntos), hacer clic con el botón derecho en una entrada de clase en los paneles *Capas* permite editar la visibilidad de las clases (y sus características) y evitar (des)marcarlas una a una:

- *Toggle Items*
- *Show All Items*
- *Hide All Items*

Con capa vectorial, el menú contextual de una entrada de hoja de clase también da acceso a:

- *Seleccionar objetos espaciales*: seleccione en la capa todos los objetos espaciales que coincidan con esa clase.
- *Mostrar en tabla de atributos*: abra una tabla de atributos filtrada sólo a los objetos espaciales que coincidan con esa clase.
- actualiza el *color del símbolo* usando una **Rueda de Color**. Para mayor comodidad, los colores usados recientemente también están disponibles en la parte inferior de la rueda de colores.
- *Editar Símbolo…*: abra el diálogo *Selector de Símbolos* para cambiar el símbolo del objeto espacial (símbolo, tamaño, color…).
- *Copy Symbol*
- *Paste Symbol*

> **Truco**
>
> Al hacer doble click en una entrada de hoja de clase, también se abre el cuadro de diálogo *Selector de símbolo*.

8.2.2 Panel de Estilizado de Capa

El panel *Estilizado de capa* (también habilitado con `Ctrl + 3`) es un acceso directo a algunas de las funcionalidades del diálogo *Propiedades de capa*. Proporciona una manera rápida y fácil de definir la representación y el comportamiento de una capa, y visualizar sus efectos sin tener que abrir el cuadro de diálogo de propiedades de capa.

Además de evitar el cuadro de diálogo de propiedades de capa de bloqueo (o «modal»), el panel de estilo de capa también evita saturar la pantalla con cuadros de diálogo y contiene la mayoría de las funciones de estilo (selector de color, propiedades de efectos, edición de reglas, sustitución de etiquetas …): por ejemplo, al hacer click en los botones de color dentro del panel de estilo de capa, se abre el cuadro de diálogo selector de color dentro del panel de estilo de capa en lugar de hacerlo como un cuadro de diálogo separado.

Desde una lista desplegable de capas actuales en el panel de capas, seleccione un elemento y:

- En función del elemento activo, ajuste:
  - *Simbología* para grupos (véase *Control de la representación de capas mediante agrupación*)
  - Propiedades *Symbología*, *Transparencia*, e *Histograma* para capa ráster. Estas opciones son las mismas que en *Dialogo de Propiedades Ráster*.
  - Propiedades *Symbología*, *Etiquetas*, *Máscara* y *Vista 3D* para capa Vectorial. Estas opciones son las mismas que en *El Diálogo de las Propiedades del Vector* y pueden ampliarse por propiedades personalizadas introducidas por complementos de terceros.

  - *Symbology*, *Labels* and *3D View* properties for mesh layer. These options are the same as in the *Propiedades del Conjunto de Datos tipo Malla*.
  - *Symbology*, *3D View* and *Elevation* properties for point cloud or 3D Tiles layer. These options are the same as in the *Propiedades de las nubes de puntos* or *Propiedades de las teselas 3D*.
  - *Symbology* and *Labels* properties for vector tiles layer. These options are the same as in the *Propiedades del conjunto de datos de teselas vectoriales*.
  - *Symbology* properties for annotations layer. These options are the same as in the *feature annotation properties*.

- Activar y configurar las propiedades *sombreado global de mapas*
- Administre los estilos asociados en *Administrador de Estilo* (más detalles en *Administrando Estilos Personalizados*).
- Ver la *Historial* de los cambios que aplicó al estilo de capa en el proyecto actual: por lo tanto, puede cancelar o restaurar a cualquier estado seleccionándolo en la lista y haciendo click en :guilabel:` Aplicar`.

Otra característica poderosa de este panel es la casilla de verificaión *Actualización en vivo*. Tíquelo para representar sus cambios inmediatamente en el lienzo del mapa: ya no necesita hacer click en el botón *Aplicar*.

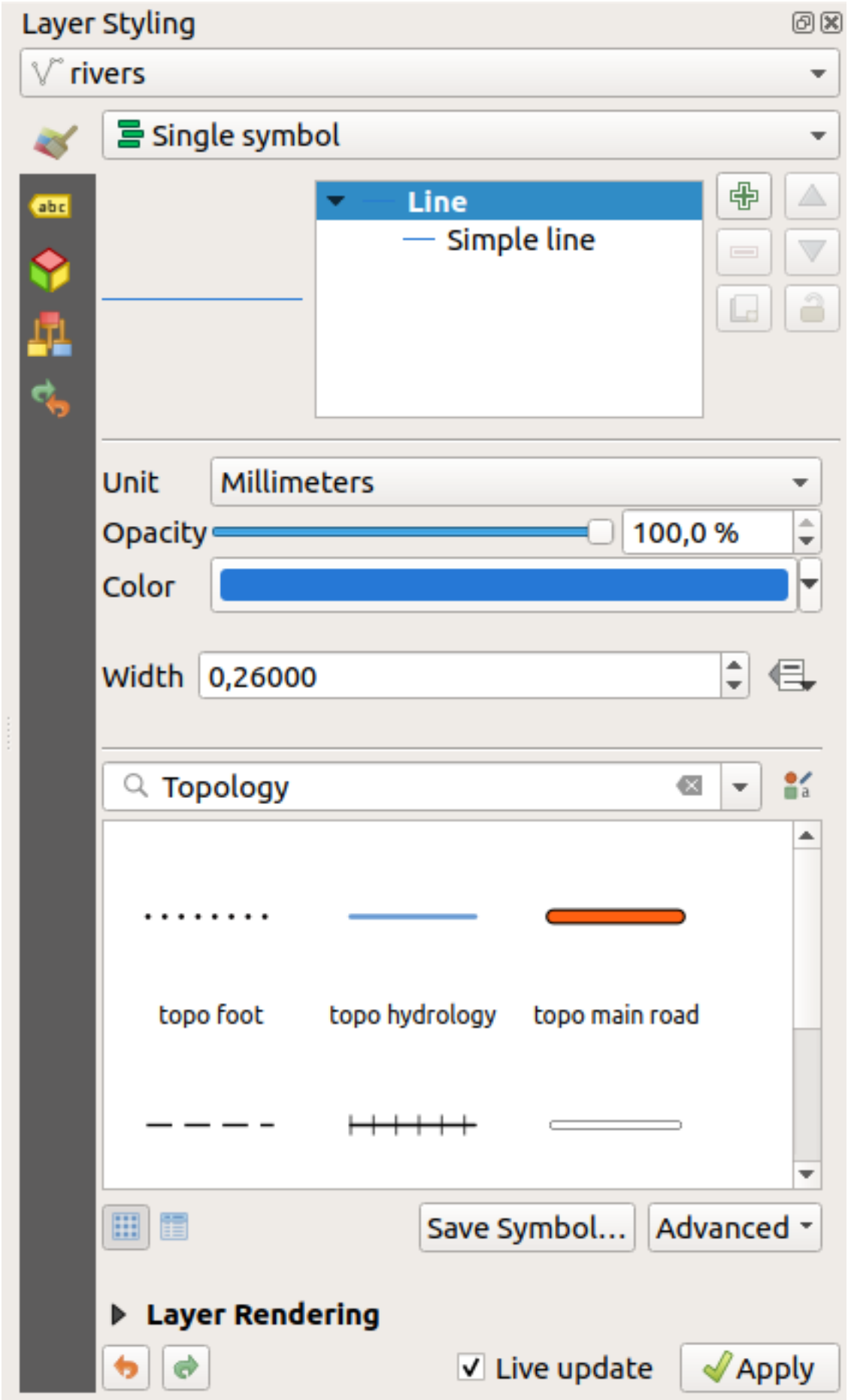


Figura 8.3: Definir la simbología de una capa desde el panel de estilo de capa

8.2.3 Panel de orden de capa

Por defecto, las capas que se muestran en el lienzo del mapa QGIS se dibujan siguiendo su orden en el panel *Capas*: cuanto más alta sea una capa en el panel, más alta (por lo tanto, más visible) estará en la vista del mapa .

Puede definir un orden de dibujo para las capas independientemente del orden en el panel de capas con el panel *Orden de Capas* activado en el menú *Ver* ► *Paneles* ► o con `Ctrl+9`. Marque *Control del orden de representación* debajo de la lista de capas y reorganice las capas en el panel como desee. Este orden será el que se aplica al lienzo del mapa. Por ejemplo, en Figura 8.4, puede ver que los objetos de `airports` son mostrados sobre el poligno de `alaska` a pesar de la ubicación de esas capas" en el panel de Capas.

Desmarcando *Controlar orden de representación* volverá al comportamiento predeterminado.

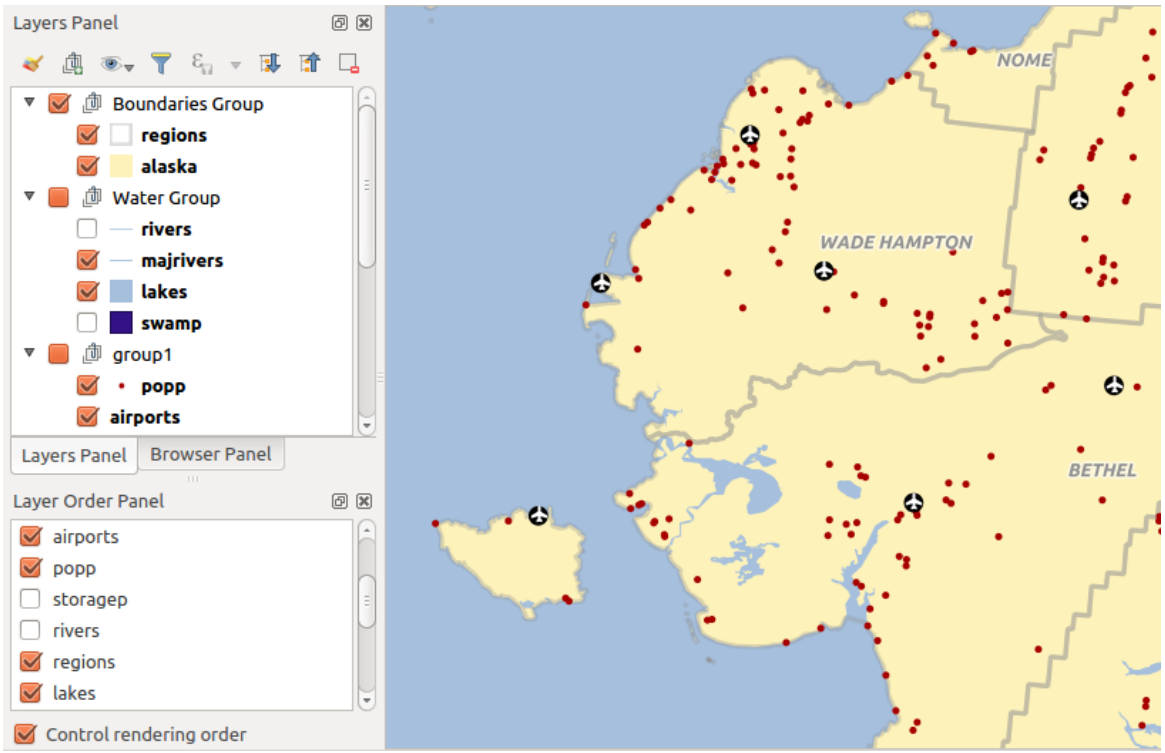


Figura 8.4: Definir un orden de capas independiente de la leyenda.

8.2.4 Panel de información general

El panel *Visión general`(:kbd:`Ctrl+8*) muestra un mapa con la extensión completa de la vista de algunas de las capas. El mapa de la visión general se llena de capas con la opción *Mostrar en vista general* del menú *Capa* o en el menú contextual de capa. Dentro de la vista, un rectángulo rojo muestra la extensión actual del lienzo del mapa, lo que le ayuda a determinar rápidamente qué área del mapa completo está viendo actualmente. Si hace click y arrastra el rectángulo rojo en el marco general, la extensión de la vista del mapa principal se actualizará en consecuencia.

Tenga en cuenta que las etiquetas no se representan en la vista general del mapa, incluso si las capas utilizadas en la vista general del mapa se han configurado para el etiquetado.

8.2.5 Panel de mensajes de registro

Al cargar o procesar algunas operaciones, puede rastrear y seguir mensajes que aparecen en diferentes pestañas usando el *Panel de Mensajes de Registro*. Se puede activar utilizando el icono más a la derecha en la barra de estado inferior.

8.2.6 Panel de deshacer/rehacer

Para cada capa que se está editando, el panel *Deshacer/Rehacer* (`Ctrl + 5`) muestra la lista de acciones realizadas, lo que le permite deshacer rápidamente un conjunto de acciones seleccionando la acción mencionada anteriormente. Más detalles en *Undo and Redo edits*.

8.2.7 Panel de resumen estadístico

El panel *Estadísticas* (`Ctrl+6`) proporciona información resumida sobre cualquier capa vectorial. Este panel le permite seleccionar:

- la capa vectorial sobre la que se calcularán las estadísticas: puede seleccionarse en el menú desplegable superior, o sincronizarse con la capa activa en el panel *Capas* utilizando la casilla *Seguir capa seleccionada* situada en la parte inferior de la lista desplegable de estadísticas.
- el campo o *expresión* a usar: para cada capa, la última entrada se recuerda y se procesa automáticamente al volver a seleccionar la capa.
- las estadísticas a devolver usando el botón desplegable en la parte inferior derecha del cuadro de diálogo. Dependiendo del tipo de campo (o valores de expresión), las estadísticas disponibles son:

Tabla 8.2: Estadísticas disponibles para cada tipo de campo

Estadísticas	Cadena	Entero	Coma flotante	Fecha
Contar	☑	☑	☑	☑
Contar valores distintos	☑			☑
Contar valores faltantes	☑	☑	☑	☑
Sum		☑	☑	
Media		☑	☑	☑
Desviación estándar		☑	☑	
Desviación estándar en ejemplo		☑	☑	
Valor mínimo	☑	☑	☑	☑
Valor máximo	☑	☑	☑	☑
Rango		☑	☑	☑
Minoría	☑	☑	☑	
Mayoría	☑	☑	☑	
Variedad		☑	☑	
Primer cuartil		☑	☑	
Tercer cuartil		☑	☑	
Rango intercuartil		☑	☑	
Longitud mínima	☑			
Longitud máxima	☑			
Longitud Media	☑			

El resumen estadístico puede ser:

- devuelto para toda la capa o *Sólo objetos espaciales seleccionados*
- copiado al portapapeles y pegado como una tabla en otra aplicación
- recalculado utilizando el botón cuando cambia la fuente de datos subyacente (por ejemplo, características / campos nuevos o eliminados, modificación de atributos)

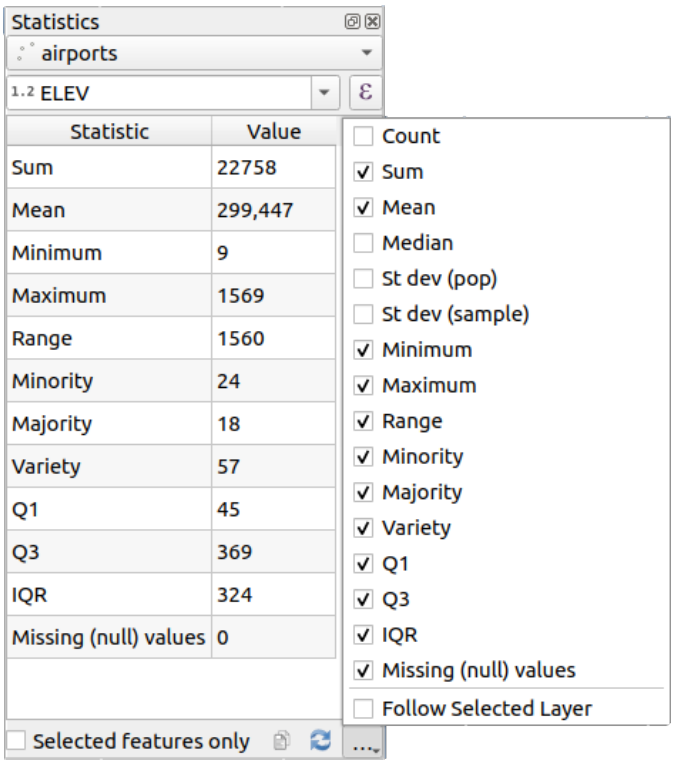


Figura 8.5: Mostrar estadística en un campo

8.2.8 Panel de herramientas de desarrollo/depuración

El panel *Herramientas de depuración/desarrollo* (`F12`) proporciona un lugar unificado para el manejo y depuración de acciones dentro de QGIS. Las herramientas disponibles están organizadas en las siguientes pestañas:

- *API Documentation*: embeds developer oriented documentation
- *Registrador de red*: registra y muestra las solicitudes e interacciones de red.
- *Registrador de consultas*: registra los comandos SQL emitidos por los proveedores de datos y la duración de la ejecución.
- *Perfilador*: proporciona los tiempos de carga de cada acción con el fin de detectar las causas de la ralentización.

Nota

Los autores de complementos pueden ampliar el panel con pestañas personalizadas para depurar y desarrollar sus propios complementos. Esto se hace usando el método `registerDevToolWidgetFactory`.

API Documentation

The *API Documentation* tab can display either the PyQGIS API documentation or the Qt API documentation. These are embedded in the widget, allowing you to browse and search the docs without leaving QGIS. It is also the place you can display specific help page for objects or classes in your code sample; from the Python interactive console or text editor dialog, select or hover a text and press `F1` or right-click and select Contextual help to get its API help. If you wish to read the active page in the system web browser, then press Open in web browser.

Read *contextual help* for details on the widget configuration.

Registrador de red

La pestaña *Registrador de red* le ayuda a registrar y mostrar las peticiones de red, junto con un montón de detalles útiles como el estado de la petición y la respuesta, cabecera, errores, errores de configuración SSL, tiempos de espera, estado de la caché, etc.

Desde su barra de herramientas superior, puede:

- *Registro*: que iniciará o detendrá el registro.
- *Borrar Registro*: borrará el historial de registros.
- *Guardar registro…*: primero mostrará una gran advertencia de que el registro es sensible y debe ser tratado como confidencial y luego le permitirá guardar el registro.
- Pulse el menú desplegable *Ajustes* para seleccionar si desea *Mostrar solicitudes aceptadas*, *Mostrar tiempos de espera* y *Mostrar respuestas servidas desde la caché*.
- *Desactivar caché*: desactivará la caché para que haya que realizar cada petición.
- *Filtrar peticiones* en función de subconjuntos de cadenas URL o del estado de la petición

Al hacer clic con el botón derecho en una solicitud, puede:

- *Open URL* que abrirá la URL en su navegador predeterminado.
- *Copiar URL*;
- *Copy As cURL* para usarlo en la terminal.
- *Copiar como JSON*: copia los valores del árbol como una cadena json al portapapeles, para facilitar su pegado en informes de errores o para asistencia remota.

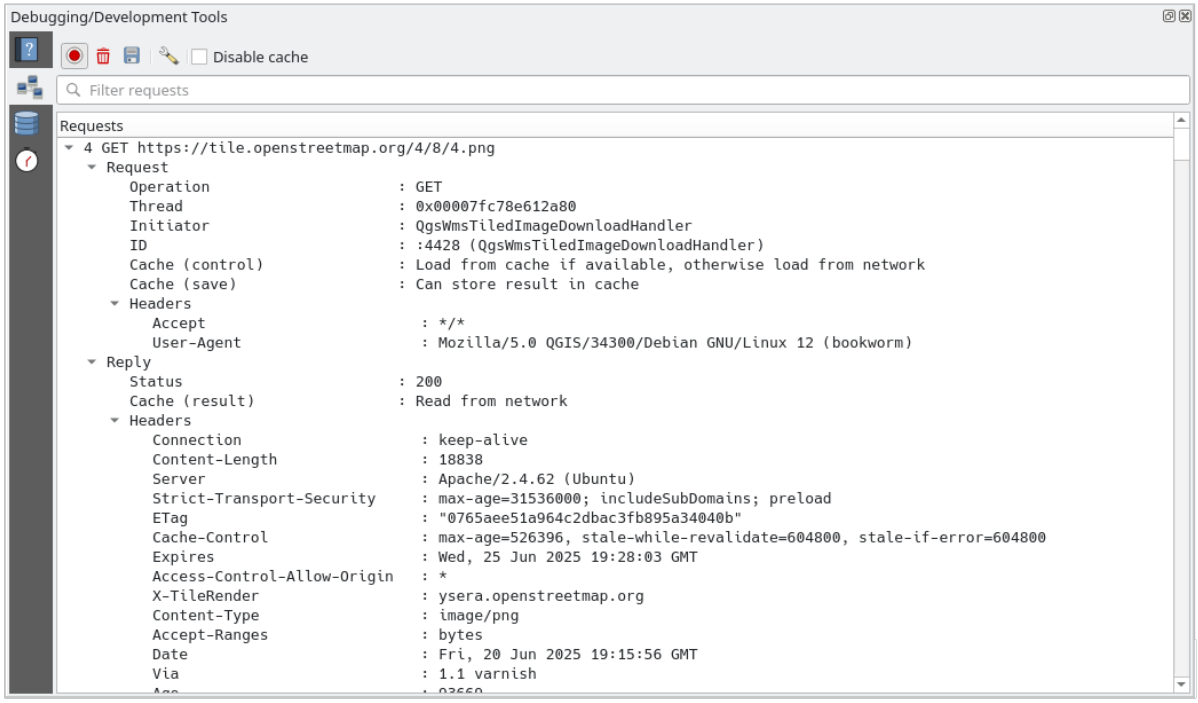


Figura 8.6: Salida del registrador de red para solicitud GET

Registrador de consultas

El *Registrador de consultas* es el lugar para registrar los comandos SQL enviados por el proveedor de datos y la API de conexiones a la base de datos backend, junto con su tiempo de ejecución medido por QGIS (es decir, en el cliente que envió los comandos). Esto puede ser útil cuando se investiga el rendimiento de una capa en particular durante la depuración o el desarrollo de un algoritmo o plugin de QGIS.

Desde su barra de herramientas superior, puede:

- *Registro*: que iniciará o detendrá el registro.
- *Borrar Registro*: borrará el historial de registros.
- *Guardar registro…*: primero mostrará una gran advertencia de que el registro es sensible y debe ser tratado como confidencial y luego le permitirá guardar el registro.
- *Filtrar consultas* basándose en los subconjuntos de la cadena de consulta o en detalles como el tipo de proveedor, la hora de inicio, el iniciador, …

Haga clic con el botón derecho del ratón en una consulta notificada:

- *Copiar comando SQL* llamado por QGIS en la base de datos
- *Copiar como JSON*: copia los valores del árbol como una cadena json al portapapeles, para facilitar su pegado en informes de errores o para asistencia remota.

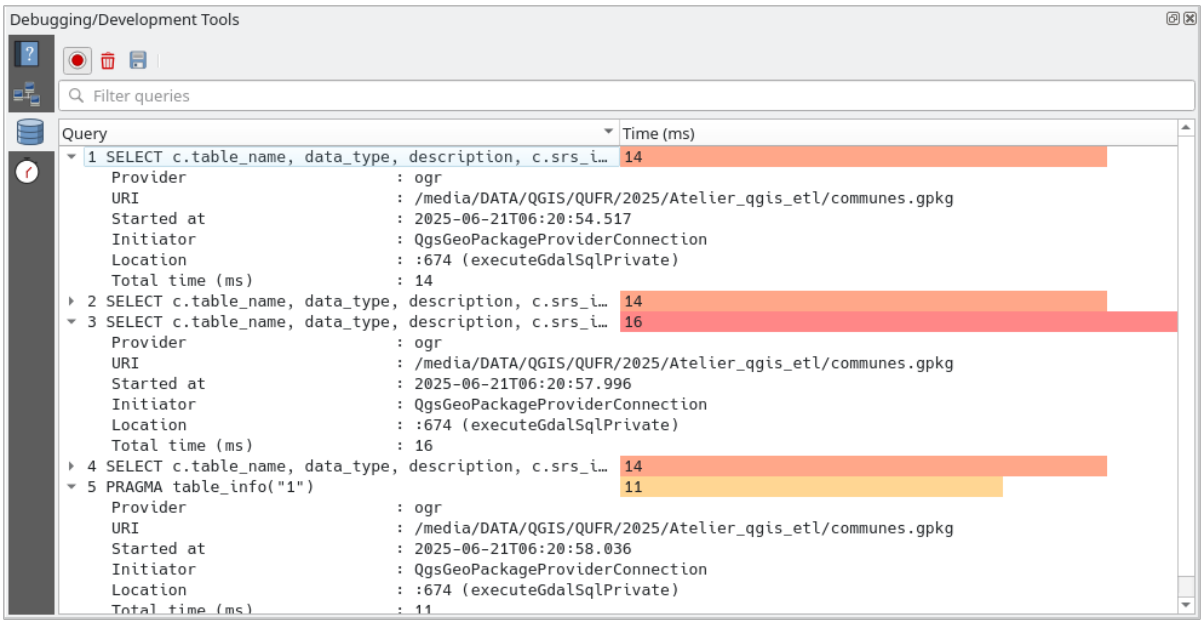


Figura 8.7: Salida del Registrador de Consultas

Perfilador

La pestaña  *Perfilador* permite obtener los tiempos de carga de cada una de las operaciones implicadas en las acciones solicitadas por el usuario. Dependiendo del contexto, estas operaciones pueden ser la lectura de configuraciones, menús, creación de lienzos de mapas o vistas 3D, resolución de referencias de capas de mapas, carga de marcadores o diseños, … Esto ayuda a identificar las causas de los tiempos de carga lentos.

Las acciones predeterminadas compatibles se pueden seleccionar en el menú desplegable *Categoría*:

- *Inicio* le ayuda a diagnosticar (y solucionar) los largos tiempos de inicio ocasionales de QGIS.
- *Carga del proyecto* le permite obtener un desglose de las distintas etapas de la carga del proyecto, con el fin de identificar las causas de los tiempos de carga lentos del proyecto.
- La herramienta de *Renderizador de mapas* le permite identificar los puntos débiles en el renderizado de sus mapas y localizar exactamente qué capas están causando largos redibujados de mapas.

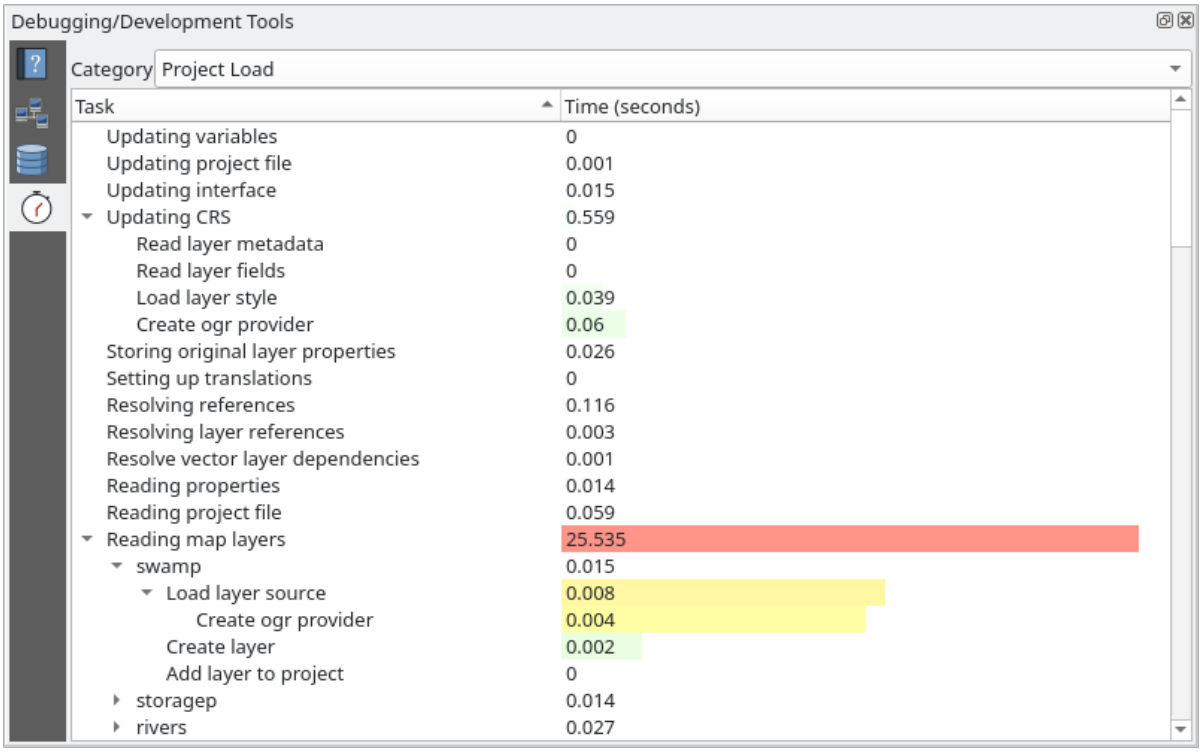


Figura 8.8: Perfilador para el inicio de QGIS

8.3 Incorporación de capas de proyectos externos

A veces, le gustaría mantener algunas capas en diferentes proyectos, pero con el mismo estilo. Puede crear un *default style* para estas capas o incrustarlas desde otro proyecto para ahorrar tiempo y esfuerzo.

Incrustar capas y grupos de un proyecto existente tiene algunas ventajas sobre el estilo:

- Todo tipo de capas (vectoriales o ráster, local o en línea…) pueden ser añadidos
- Al obtener grupos y capas, puede mantener la misma estructura de árbol de las capas de «fondo» en sus diferentes proyectos
- Si bien las capas incrustadas son editables, no puede cambiar sus propiedades, como simbología, etiquetas, formularios, valores predeterminados y acciones, lo que garantiza la coherencia entre los proyectos.
- Modifique los elementos en el proyecto original y los cambios se propagan a todos los demás proyectos.

Si desea incrustar contenido de otros archivos de proyecto en su proyecto, seleccione *Capa* ► *Incrustar Capas y Grupos*:

1. Click en el botón … para buscar un proyecto: puede ver el contenido del proyecto (ver Figura 8.9)
2. Mantenga pulsado `Ctrl` (o `Cmd`) y haga clic en las capas y grupos que desea recuperar
3. Haga clic en *Aceptar*.

Las capas y grupos seleccionados se incrustan en el panel *Capas* y se muestran en el lienzo del mapa. Un ícono se agrega al lado de su nombre para reconocimiento y al pasar el cursor sobre él se muestra una herramienta de información con la ruta original del archivo del proyecto.

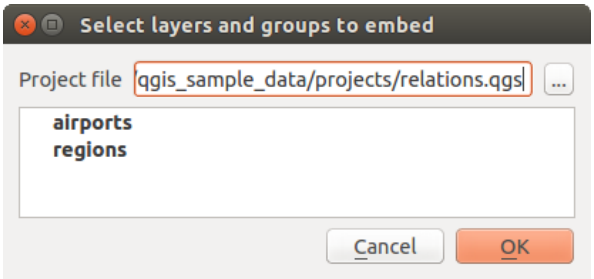


Figura 8.9: Seleccionar capas y grupos para empotrar

Al igual que cualquier otra capa, una capa incrustada se puede eliminar del proyecto haciendo clic derecho en la capa y haciendo clic en Borrar.

Truco

Cambiar la representación de una capa incrustada

No es posible cambiar la representación de una capa incrustada, a menos que realice los cambios en el archivo original del proyecto. Sin embargo, al hacer clic con el botón derecho en una capa y seleccionar:guilabel:*Duplicar* crea una capa con todas las funciones y que no depende del proyecto original. Luego puede eliminar con seguridad la capa vinculada.

8.4 Interactuando con entidades

8.4.1 Seleccionando objetos espaciales

QGIS proporciona múltiples herramientas para seleccionar entidades en el lienzo del mapa. Las herramientas de selección están disponibles en el menú *Editar* ► *Selección* o en la:guilabel:*Barra de Herramientas de Selección*.

Nota

Las herramientas de selección funcionan con la capa actualmente activa.

Seleccionando manualmente en el lienzo del mapa

Para seleccionar una o más funciones con el ratón, puede usar una de las siguientes herramientas:

- Seleccionar entidades por área o un solo clic
- Seleccionar entidades por Polígono
- Seleccionar entidades a mano alzada
- Seleccionar entidades por Radio

Nota

Aparte de Seleccionar entidades por polígono, estas herramientas de selección manual le permiten seleccionar entidades en el lienzo del mapa con un solo click.

Nota

Use la herramienta  Seleccionar objetos mediante polígono para usar un objeto poligonal existente (de cualquier capa) para seleccionar capas superpuestas en la capa activa. Click-derecho en el polígono y elijala desde el menú contextual que muestra una lista de todos los polígonos que contienen el punto clickado. Todos los objetos de la capa activa que superponen al polígono son seleccionados.

Truco

Use la herramienta *Editar -> Seleccionar -> Reseleccionar entidades* para rehacer su última selección. Muy útil cuando ha realizado una selección minuciosa y luego hizo click accidentalmente en otro lugar y borró su selección.

Mientras se utiliza la herramienta  *Seleccionar Objeto(s) espacial(es)*, manteniendo pulsado `Shift` o `Ctrl` (`Cmd` en macOS) se alterna si una característica está seleccionada (es decir, se añade a la selección actual o se elimina de ella).

Para las otras herramientas, se pueden obtener diferentes comportamientos presionando:

- `Shift`: añadir entidades a la selección actual
- `Ctrl`/`Cmd`: resta objetos espaciales de la selección actual
- `Ctrl+Mayús`/`Cmd+Mayús`: intersecar con la selección actual, es decir, mantener sólo las características superpuestas de la selección actual.
- `Alt`: selecciona objetos espaciales que están totalmente dentro de la forma de la selección. Combinado con las teclas `Shift` o `Ctrl`/`Cmd`, puede añadir o sustraer elementos a/de la selección actual.

Selección automática

Las otras herramientas de selección, la mayoría de ellas disponibles en *Attribute table*, realice una selección basada en el atributo de una entidad o su estado de selección (tenga en cuenta que la tabla de atributos y el lienzo del mapa muestran la misma información, por lo que si selecciona una entidad en la tabla de atributos, también se seleccionará en el lienzo del mapa):

-  Select Features By Expression… or press `Ctrl+F3` to select features using the expression dialog
-  Seleccionar Entidades por Valor… o presione `F3`
-  Deseleccionar Objetos de Todas las Capas o presiona `Ctrl+Alt+A` para deseleccionar todos los objetos en todas las capas
-  Deseleccionar Objetos de la Capa Activa Actual o presione `Ctrl+Shift+A`
-  Seleccionar Todas las Entidades o presione `Ctrl+A` para seleccionar todas las entidades en la capa actual
-  Invertir Selección de Entidades para invertir la selección el la capa actual
-  Seleccionar por ubicación para seleccionar las entidades en función de su relación espacial con otras entidades (en la misma capa o en otra; consulte *Seleccionar por ubicacion*)
-  Seleccionar dentro de la distancia para seleccionar características siempre que se encuentren dentro de la distancia máxima especificada desde las características referenciadas - véase *Seleccione dentro de la distancia*)

Por ejemplo, si desea buscar regiones que son distritos de: file: *region.shp* de los datos de muestra de QGIS, puede:

1. Usar el icono  Seleccionar Entidades Usando una Expresión

2. Expanda el grupo *Campos y Valores*
3. Doble-click en el campo que quiera consultar («TYPE_2»)
4. Click en *Todos Únicos* en el panel que se muestra arriba a la derecha
5. De la lista, doble-click “Borough”. En el editor de campos *Expresion*, escriba la siguiente consulta:

```
"TYPE_2"  =  'Borough'
```

6. Click en *Seleccionar Entidades*

Desde el cuadro de diálogo del generador de expresiones, también puede usar *Lista de funciones -> Reciente (Selección)* para hacer una selección que ha usado antes. El diálogo recuerda las últimas 20 expresiones utilizadas. Ver *Expresiones* para más información y ejemplos.

Truco

Guarde su selección en un nuevo archivo

Los usuarios pueden guardar las funciones seleccionadas en una **Nueva capa temporal temporal** o una **Nueva capa vectorial** usando *Editar -> Copiar entidades* y :menuselection:` Editar -> Pegar entidades como` en el formato deseado.

Seleccionar objetos por valor

Esta herramienta de selección abre el formulario de características de la capa que permite al usuario elegir qué valor buscar para cada campo, si la búsqueda debe distinguir entre mayúsculas y minúsculas y la operación que se debe utilizar. La herramienta también se completa automáticamente, rellenando automáticamente el cuadro de búsqueda con los valores existentes.

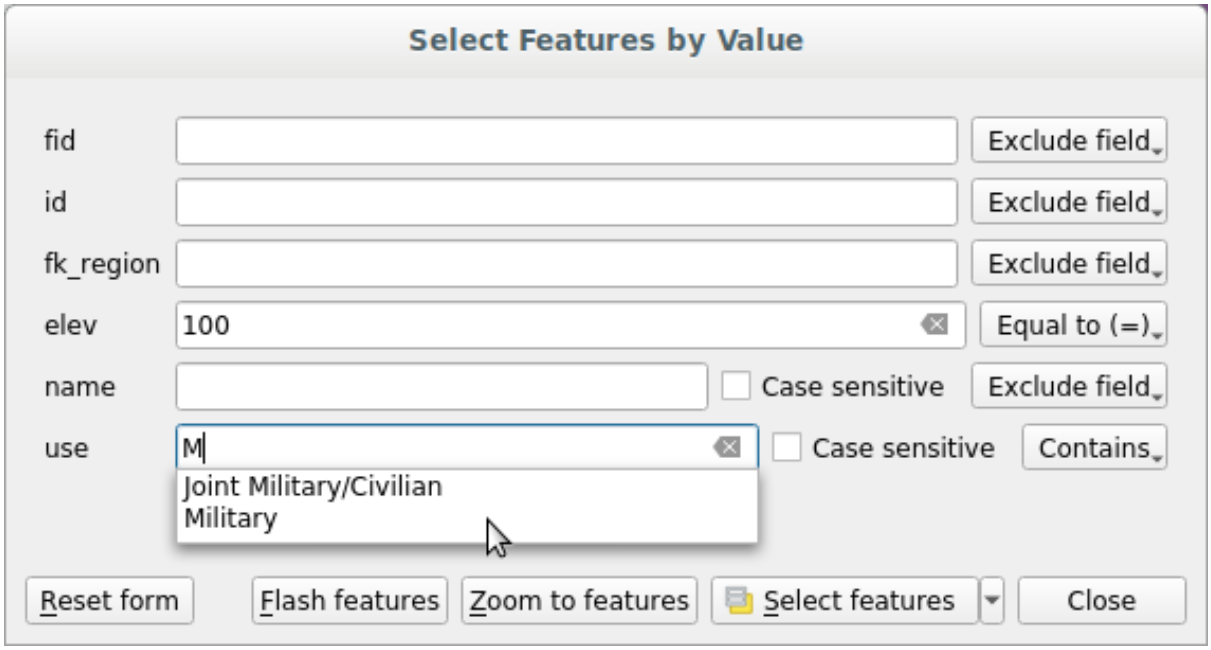


Figura 8.10: Filtrar / Seleccionar entidades usando el diálogo de formulario

Junto a cada campo, hay una lista desplegable con opciones para controlar el comportamiento de búsqueda:

Tabla 8.3: Operadores de consulta por tipo de datos

Opción de búsqueda de campo	Cadena	Numeric	Fecha
Excluir campo de la búsqueda	☑	☑	☑
Igual a (=)	☑	☑	☑
No igual a (≠)	☑	☑	☑
Mayor que (>)		☑	☑
Menor que (<)		☑	☑
Mayor o igual a (≥)		☑	☑
Menor o igual que (≤)		☑	☑
Entre (inclusivo)		☑	☑
No entre (inclusivo)		☑	☑
Contiene	☑		
No contiene	☑		
Falta (nulo)	☑	☑	☑
No Falta (no nulo)	☑	☑	☑
Empieza con	☑		
Acaba con	☑		

Para las comparaciones de cadenas, también es posible utilizar la opción ☑ *mayúsculas y minúsculas*.

Después de configurar todas las opciones de búsqueda, haga click en *Seleccionar entidades* para seleccionar las entidades coincidentes. Las opciones desplegables son:

- *Seleccionar objetos espaciales*
- *Añadir a una selección actual*
- *Borrar de la selección actual*
- *Filtrar la selección actual*

También puede borrar todas las opciones de búsqueda con el botón *Restablecer formulario*.

Una vez que las condiciones están establecidas, también puedes:

- *Zoom a las entidades* en el lienzo del mapa sin la necesidad de una preselección
- *Funciones de flash*, destacando las funciones coincidentes. Esta es una forma práctica de identificar una característica sin seleccionar o usar la herramienta Identificar. Tenga en cuenta que el flash no altera la extensión del lienzo del mapa y sería visible solo si la entidad se encuentra dentro de los límites del lienzo del mapa actual.

8.4.2 Identificando entidades

La herramienta  Identificar objetos espaciales permite interactuar con el lienzo del mapa y obtener información sobre características o píxeles en una ventana emergente. Se puede utilizar para consultar la mayoría de los tipos de capas soportados por QGIS (vector, ráster, malla, nube de puntos, wms, wfs, …). Para identificar un elemento, utilice

- *Ver* ► *Identificar Entidades*
- `Ctrl+Shift+I` (o `Cmd+Shift+I`),
-  Identificar objetos espaciales de la barra de herramientas *Atributos*

A continuación, haga clic en un objeto espacial o píxel de la capa activa. El elemento identificado se resalta en el lienzo del mapa mientras se abre el cuadro de diálogo *Identificar resultados* con información detallada sobre el mismo. El diálogo también muestra un conjunto de botones para la configuración avanzada.

El Dialogo de resultados de la identificación

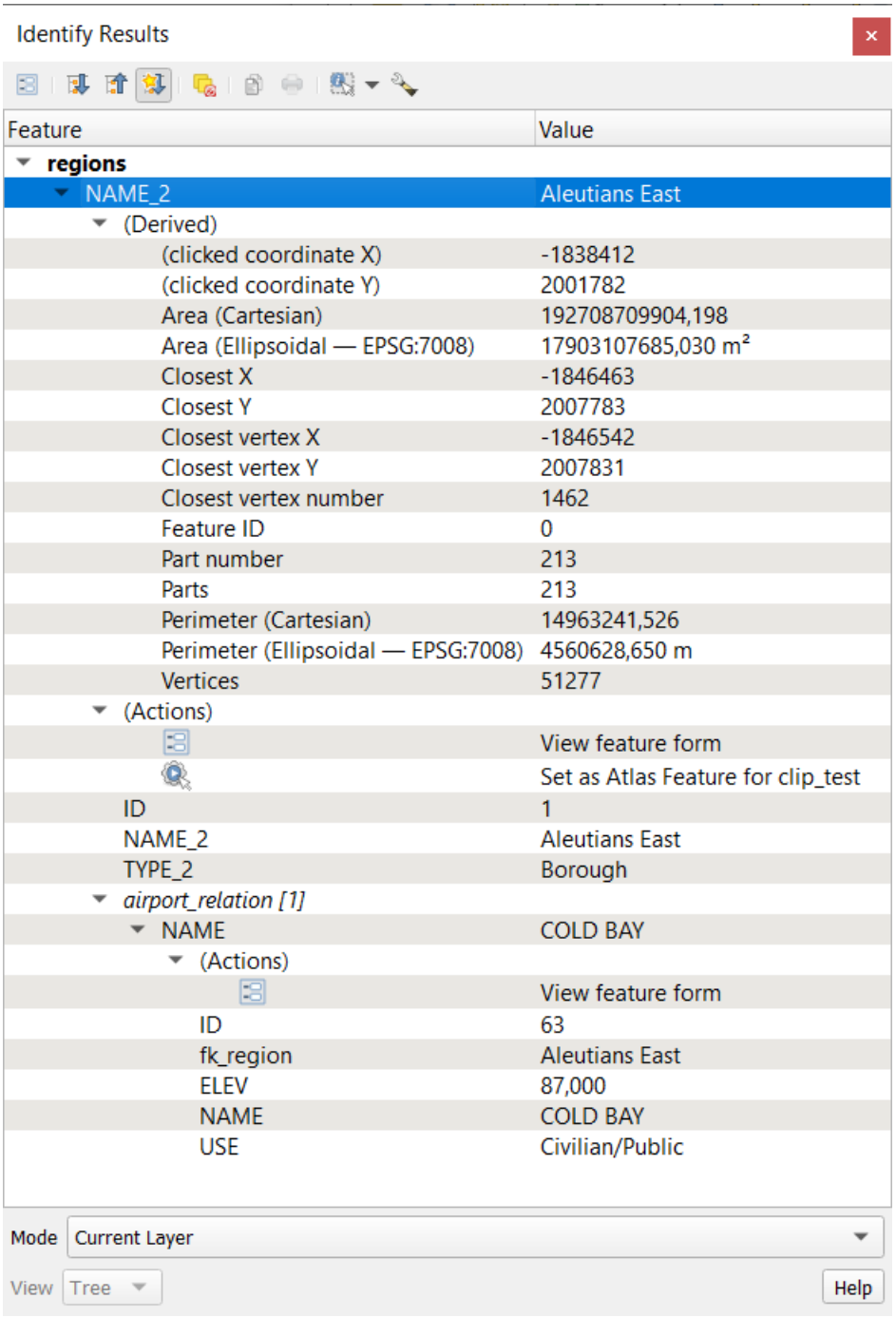


Figura 8.11: Dialogo de resultados de la identificación

De abajo a arriba:

- La *Vista* controla el aspecto general del cuadro de diálogo y el formato de los resultados; puede establecerse como:

  - **Tree**: this is the default view, and returns the results in a tree-structure where the first item is the name of the layer and its children are its identified element(s).
  - **Tabla**: disponible sólo para capas basadas en ráster, permite mostrar los resultados como una tabla cuyas columnas son `Layer`, `FID`, `Atribute` y `Value`.
  - o **Gráfico**: disponible sólo para capas basadas en ráster.
- La etiqueta *Modo* le ayuda a seleccionar las capas de las que pueden obtenerse resultados. Estas capas deben establecerse visibles, mostrando los datos en el lienzo del mapa, y establecerse *identificable* desde las **:menuselección:`Propiedades del proyecto --> Fuentes de datos --> Capacidades de las capas`**. Los modos disponibles son:
  - **Capa actual**: sólo la(s) capa(s) seleccionada(s) en el panel *Capas* devuelve(n) resultados. Si se selecciona un grupo, los resultados se obtienen de sus capas hoja.
  - **De arriba abajo, deténgase al principio**: los resultados se obtienen a partir de la capa de la prestación o píxel situado más arriba bajo el ratón.
  - **Arriba abajo**: los resultados proceden de las capas con objeto espacial o píxel bajo el ratón.
  - **Selección de capa**: abre un menú contextual en el que el usuario selecciona la capa desde la que identificar los objetos espaciales. Si sólo hay un objeto espacial bajo el ratón, los resultados se muestran automáticamente.
- In the upper part, the information widget: when you identify a data in the map canvas, this is the place where the *Identify Results* dialog will list details about the clicked (or hovered over, depending on the tool in use) items. Their formatting relies on the *selected view*.

  The information displayed by the identify tool will depend on the type of layer you have selected, whether it is:
  - a *vector layer* (including vector tiles or point cloud data),
  - a *raster layer*,
  - a *mesh layer*.
- When in a tree view, the upper part also displays a handful of tools above the results:
  -  Formulario abierto de la entidad actual
  -  Expandir árbol
  -  Colapsar árbol
  -  Expandir nuevos resultados por defecto para definir si la siguiente información de la entidad identificada debe contraerse o expandirse
  -  Limpiar Resultados
  -  Copiar el objeto espacial identificado al portapapeles, adecuado para pegar en una hoja de cálculo.
  -  Imprimir respuesta HTML seleccionada: un formato basado en texto de los resultados para imprimir en papel o guardar como archivo `.PDF`.
  - las *herramientas de identificación interactiva*: un menú desplegable con herramientas para seleccionar en el lienzo del mapa objetos espaciales o píxeles que identificar
  - En  Identificar configuración, puede activar si desea:
    *  *Abrir formulario automáticamente para resultados de características individuales*: Si se marca, cada vez que se identifique una característica, se abrirá un formulario mostrando sus atributos. Es una forma práctica de editar rápidamente los atributos de un objeto espacial.
    *  *Ocultar atributos derivados de los resultados* para mostrar sólo los campos realmente definidos en la capa

    - *Ocultar valores NULL de los resultados*
    - *Show Relations* to show relations in the vector layer *Identify results tree*
  - Ayuda para acceder a la documentación actual

Usando la herramienta de identificación de entidades

En su presentación predeterminada (*Vista: Árbol*), el panel *Identificar resultados* ofrece varias herramientas para interactuar con las capas a consultar. Una combinación inteligente de estas herramientas con el *selector de capas objetivo* puede mejorar considerablemente las operaciones de identificación:

- Identificar objeto(s) espacial(es) con un solo clic o haciendo clic y arrastrando
  - un solo clic o hacer clic y arrastrar: se devuelven los objetos espaciales superpuestos en las capas de destino
  - clic con el botón derecho: los objetos espaciales superpuestos de las capas de destino aparecen en el menú contextual, agrupadas por capas. A continuación, puede elegir:
    - mostrar el resultado de un objeto espacial concreto,
    - mostrar el resultado para todas los objetos espaciales de una capa específica,
    - para las capas vectoriales, también es posible abrir su tabla de atributos filtrada a los objetos espaciales devueltos
    - o mostrar todos los objetos espaciales devueltos.
- Identificar objetos espaciales al pasar el ratón por encima: al pasar el ratón por encima del lienzo del mapa, los elementos sobre los que se pase el ratón en las capas de destino se resaltarán y aparecerán en el panel de resultados.
- Identificar objetos espaciales por polígono: devuelve elementos superpuestos a un polígono dibujado o seleccionado.
  - Dibuje un polígono (clic izquierdo para añadir un punto, clic derecho para cerrar el polígono) y todos los objetos espaciales superpuestos de las capas de destino se resaltarán y aparecerán en el panel de resultados.
  - Haga clic con el botón derecho y obtendrá la lista de todos los objetos espaciales polígono visibles en el proyecto bajo el clic. Elija una entrada y QGIS devolverá todas los objetos espaciales de las capas de destino que se solapan con el polígono seleccionado.
- Identificar objetos espaciales a mano alzada: devuelve elementos superpuestos a un polígono dibujado a mano alzada. Dibuje un polígono (haga clic con el botón izquierdo para empezar, mueva el puntero para dar forma al área y haga clic con el botón derecho para cerrar el polígono). Todas las prestaciones superpuestas de las capas de destino se resaltan y se devuelven en el panel de resultados.
- Identificar objetos espaciales por radio Devuelve elementos superpuestos a un círculo dibujado. Dibuje un círculo (haga click izquierdo para indicar el punto central, mueva el puntero para dar forma al área o introduzca el radio en el cuadro de texto emergente y haga click izquierdo o pulse `Enter` para validar el círculo). Todos los objetos superpuestos de las capas objetivo se resaltan y se muestran en el panel de resultados.

Truco

Filtre las capas para consultar con la herramienta Identificar entidades

En *Proyecto* ► *Propiedades…* ► *Fuentes de Datos* ► *Capacidades de Capa*, desmarque la columna *Identificable* junto a una capa para evitar que sea consultada cuando utilice la herramienta Identificar objetos espaciales. Esta es una forma práctica de devolver objetos espaciales sólo de las capas que sean de su interés.

Menú contextual de resultados

Other functions can be found in the context menu of the identified item. For example, depending on the layer type, you can:

- interact with the individual identified item:
  - *View feature form* or *Edit feature form*, for vector layers
  - *Zoom to feature*: Zooms the map canvas to the extent of the identified feature or pixel
  - *Identify feature*: Restarts and recenters the search results on a *referenced or referencing vector feature* of the identified feature. Only available if  *Show Relations* is checked in Identify Settings.
  - *Copy feature*: Copies the vector geometry and attributes, or the pixel geometry
  - *Toggle feature selection*: Adds or removes identified feature to/from the selection
  - *Copy attribute value*: Copies the value in the *Value* column of the tree item you click on
  - *Copy feature attributes*: Copies the attribute names and values of the identified item
  - *Select features by attribute value*: Selects all vector features in the layer that match the selected attribute value
- control the rendering of the layers, the results panel or the map canvas:
  - *Clear result*: Removes results in the window
  - *Clear highlights*: Removes items highlighted on the map
  - *Highlight all*: Highlights in the map canvas all the items identified in the results dialog, regardless of their layer
  - *Highlight layer*: Highlights in the map canvas all the items identified in the layer of the selected entry
  - *Activate layer*: Selects the layer of the identified item in the *Layers* panel
  - *Layer properties*: Opens the layer properties window of the identified item
  - *Expand all* or *Collapse all* the results tree

8.5 Guardar y compartir Propiedades

8.5.1 Administrando Estilos Personalizados

Cuando se agrega una capa vectorial al lienzo del mapa, QGIS utiliza de forma predeterminada un símbolo/color aleatorio para representar sus entidades. Sin embargo, puede establecer un símbolo predeterminado en *Proyecto -> Propiedades … -> Estilos predeterminados* que se aplicará a cada capa recién agregada de acuerdo con su tipo de geometría.

Sin embargo, la mayoría de las veces preferiría tener un estilo personalizado y más complejo que se pueda aplicar automática o manualmente a las capas (con menos esfuerzo). Puede lograr esto utilizando el menú *Estilo* en la parte inferior del cuadro de diálogo Propiedades de Capa. Este menú le proporciona funciones para crear, cargar y administrar estilos.

Un estilo almacena diversa información establecida en el diálogo de propiedades de la capa o del proyecto para renderizar o interactuar con la capa. Esto incluye:

Tabla 8.4: Componentes del estilo almacenado y sus referencias

Categoría	Vector	Raster
Configuración de Capas	*Capacidades de las capas*	
	Propiedades a mostrar	*Propiedades a mostrar*
Simbología	*Propiedades de simbología*	*Propiedades de simbología* ([1])
Simbología 3D	*Propiedades de visor 3D*	N/D
Etiquetas	*Propiedades de etiquetas*	N/D
Campos	*Propiedades de campos* y restricciones	N/D
Formulario de atributos	*Formulario de propiedades de campo*	N/D
Acciones	*Propiedades de acciones*	N/D
Consejos sobre mapas	*Propiedades a mostrar*	*Propiedades a mostrar*
Diagramas	*Propiedades de diagrama*	N/D
Configuración de la tabla de atributos	*Trabajar con la tabla de atributos*	
Representación	*Propiedades de representación*	*Propiedades de representación* ([1])
Propiedades personalizadas	*Propiedades de información*	*Propiedades de información*
Opciones de geometría	*Propiedades de Digitalizado*	N/D
Relaciones	*Establecer relaciones entre varias capas*	N/D
Propiedades Temporales	*Propiedades Temporales*	*Propiedades Temporales*
Configuración de leyenda	*Propiedades de la leyenda*	N/D
Propiedades de la elevación	*Propiedades de elevación*	*Propiedades de elevación*
Notas	*Notas de capa*	

Por defecto, el estilo aplicado a una capa cargada se denomina `predeterminado`. Una vez que tenga el renderizado ideal y apropiado para su capa, puede guardarlo haciendo click en el cuadro combinado *Estilo* y elegir:

- **Cambiar nombre actual**: el estilo activo cambia de nombre y se actualiza con las opciones actuales
- **Agregar**: se crea un nuevo estilo con las opciones actuales. Por defecto, se guardará en el archivo del proyecto QGIS. Consulte a continuación para guardar el estilo en otro archivo o una base de datos.
- **Eliminar**: elimine el estilo no deseado, en caso de que tenga más de un estilo definido para la capa.

En la parte inferior de la lista desplegable Estilo, puede ver los estilos establecidos para la capa con la activa activada.

Tenga en cuenta que cada vez que valida el cuadro de diálogo de propiedades de capa, el estilo activo se actualiza con los cambios que ha realizado.

Puede crear tantos estilos como desee para una capa, pero solo uno puede estar activo a la vez. En combinación con *Map Themes*, esto ofrece una manera rápida y poderosa de administrar proyectos complejos sin la necesidad de duplicar ninguna capa en la leyenda del mapa.

Nota

[1] For raster layers, symbology and rendering items are melted together, meaning that saving/loading one would also save/load items of/from the other.

Dado que cada vez que aplica modificaciones a las propiedades de la capa, los cambios se almacenan en el estilo activo, asegúrese siempre de que está editando el estilo correcto para evitar alterar por error un estilo utilizado en un *map theme*.

Truco

Administrar estilos desde el menú contextual de la capa

Haga clic con el botón derecho del ratón sobre la capa en el panel *Capas* para copiar, pegar, añadir o renombrar estilos de capa. Tanto para capas vectoriales como rasterizadas, puede elegir qué categorías de estilo copiar/pegar. Elija *Todas las categorías de estilo* o una de las categorías disponibles, dependiendo del tipo de capa. Tenga en cuenta que para las capas rasterizadas, las categorías *Simbología* y *Renderizado* siempre se copian juntas.

8.5.2 Almacenar estilos en un archivo o una base de datos

Si bien los estilos creados desde el cuadro combinado *Estilo* se guardan por defecto dentro del proyecto y se pueden copiar y pegar de capa a capa en el proyecto, también es posible guardarlos fuera del proyecto para que puedan cargarse en otro proyecto.

Guardar como archivo de texto

Clickando el  *Estilo* ► *Guardar Estilo*, puede guardar este estilo como un:

- Archivo de estilo de capa QGIS (`.qml`)
- Archivo SLD (`.sld`), solo disponible para capars vectoriales

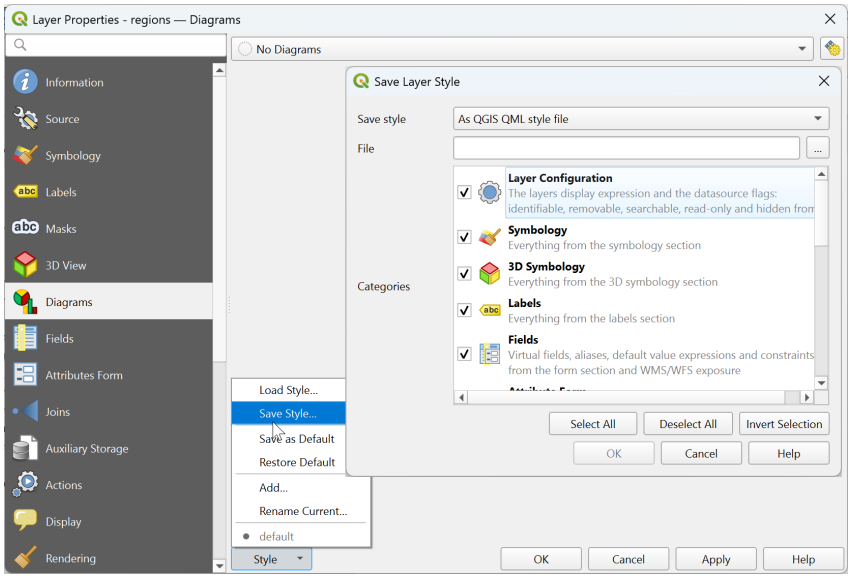


Figura 8.12: Cuadro combinado de opciones de Estilo de capa vectorial

Se utiliza en capas de formato basadas en archivos (`.shp`, :file:` .tab` …), *Guardar como predeterminado* genera un archivo `.qml` para la capa (con el mismo nombre) Los SLD se pueden exportar desde cualquier tipo de renderizador (símbolo único, categorizado, graduado o basado en reglas), pero al importar un SLD, se crea un solo símbolo o un renderizador basado en reglas. Esto significa que los estilos categorizados o graduados se convierten en basados en reglas. Si desea preservar esos renderizadores, debe usar el formato QML. Por otro lado, a veces puede ser muy útil tener esta forma fácil de convertir estilos a reglas.

Guardar en la base de datos

Vector layer styles can also be stored in a database if the layer datasource is a database provider. Supported formats are PostgreSQL, GeoPackage, SpatiaLite, MS SQL Server and Oracle. The layer style is saved inside a table (named `layer_styles`) in the database. Click on *Save Style…* ► *Save in database* then fill in the dialog to define a style name, add a description, a `.ui` file if applicable and to check if the style should be the default style.

Puede guardar varios estilos para una sola tabla en la base de datos. Sin embargo, cada tabla solo puede tener un estilo predeterminado. Los estilos predeterminados se pueden guardar en la base de datos de capas o en `qgis.db`, una base de datos SQLite local en el directorio activo *user profile*.

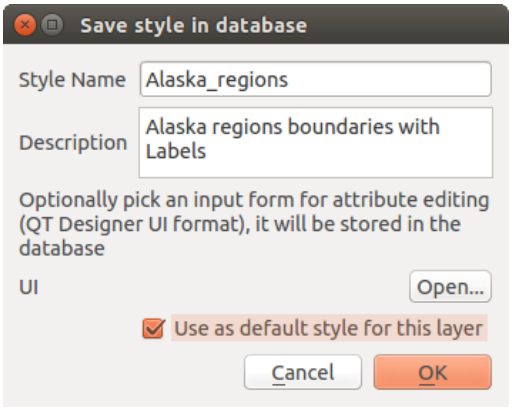


Figura 8.13: Guardar estilo en el Diálogo base de datos

Truco

Compartir archivos de estilo entre bases de datos

Sólo se puede guardar el estilo en una base de datos si la capa procede de dicha base de datos. No puedes mezclar bases de datos (capa en Oracle y estilo en MS SQL Server, por ejemplo). Utiliza en su lugar un archivo de texto plano si quieres que el estilo se comparta entre bases de datos.

Nota

Puede encontrar problemas al restaurar la tabla `layer_styles` desde una copia de seguridad de la base de datos PostgreSQL. Seguir *Copia de seguridad de base de datos y tabla de estilo de capa de QGIS* to fix that.

Cargar estilos

Al cargar una capa en QGIS, si ya existe un estilo predeterminado para esta capa, QGIS carga la capa con este estilo. También *Estilo -> Restaurar Predeterminado* busca y carga ese archivo, reemplazando el estilo actual de la capa.

Estilo ► *Cargar Estilo* le ayuda a aplicar cualquier estilo guardado a una capa. Si bien los estilos de archivo de texto (`.sld` o `.qml`) se pueden aplicar a cualquier capa, sea cual sea su formato, cargar estilos almacenados en una base de datos solo es posible si la capa es de la misma base de datos o El estilo se almacena en la base de datos local QGIS.

El diálogo *Administrador de estilos de base de datos* muestra una lista de estilos relacionados con la capa que se encuentra en la base de datos y todos los demás estilos guardados en ella, con nombre y descripción.

Truco

Compartir rápidamente un estilo de capa dentro del proyecto

También puede compartir estilos de capa dentro de un proyecto sin importar un archivo o estilo de base de datos: haga click con el botón derecho en la capa en *Panel de capas* y, desde el cuadro combinado *Estilos*, copie el estilo de un capa y péguelo en un grupo o una selección de capas: el estilo se aplica a todas las capas que son del mismo tipo (vector vs ráster) que la capa original y, en el caso de capas vectoriales, tienen el mismo tipo de geometría (punto, línea o polígono).

8.5.3 Archivo de definición de capa

Las definiciones de capa se pueden guardar como un `Archivo de definición de capa (.qlr)` usando *Exportar -> Guardar como archivo de definición de capa…`en el menú contextual de las capas activas. Un archivo de definición de capa (:file:*.qlr`) incluye referencias a la fuente de datos de las capas y sus estilos. los archivos `.qlr` se muestran en el Panel del navegador y se pueden usar para agregar las capas (con el estilo guardado) al Panel de capas. También puede arrastrar y soltar archivos `.qlr` desde el administrador de archivos del sistema en el lienzo del mapa.

8.6 Documentando tus datos

Además de mostrar y simbolizar los datos en las capas, QGIS le permite completar:

- **metadata**: information to help people find and understand the dataset, how they can access and use it… these are properties of the datasource and can live out of the QGIS project. Metadata can also be defined for the project file itself.
- **notes**: instrucciones y comentarios sobre la capa en el proyecto actual

8.6.1 Metadatos

In the project or layer properties dialog, the  *Metadata* tab provides you with options to create and edit a metadata report on the element. Information to fill concern:

- La *Identificación* de los datos: atribución básica del conjunto de datos (identificador padre, identificador, título, resumen, idioma…);
- Las *Categorías* a las que pertenecen los datos. Además de las categorías **ISO**, puede agregar categorías personalizadas;
- Las *Palabras clave* para recuperar los datos y conceptos asociados siguiendo un vocabulario basado en estándares;
- El *Acceso* al conjunto de datos (licencias, derechos, tarifas y restricciones);
- La *Extensión* del conjunto de datos, ya sea espacial (CRS, extensión del mapa, altitudes) o temporal;
- El *Contacto* del propietario(s) del conjunto de datos;
- Los *Enlaces* a recursos auxiliares e información relacionada;
- El *Historial* del conjunto de datos.

Se proporciona un resumen de la información completada en la pestaña *Validación* y lo ayuda a identificar posibles problemas relacionados con el formulario. A continuación, puede corregirlos o ignorarlos.

Los metadatos se guardan por defecto en el archivo del proyecto, el desplegable *Metadatos* ofrece opciones para cargar/guardar metadatos desde el archivo `.qmd` y para cargar/guardar metadatos en la ubicación «Predeterminada».

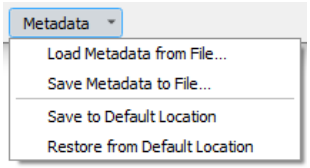


Figura 8.14: Opciones de carga/guardado de metadatos

La ubicación «Predeterminada» utilizada por *Guardar en Ubicación Predeterminada* y *Restaurar desde Ubicación Predeterminada* cambia dependiendo de la fuente de datos subyacente y de su configuración:

- Para fuentes de datos PostgreSQL si la opción de configuración *Permitir guardar/cargar metadatos de capas QGIS en la base de datos* está marcada los metadatos se almacenan dentro de una tabla dedicada en la base de datos.
- Para fuentes de datos GeoPackage *Guardar en Ubicación Predeterminada* siempre guarda los metadatos en las tablas de metadatos internas del GeoPackage.

  Cuando los metadatos se almacenan en tablas internar de PostgreSQL o GeoPackage pueden llegar a estar disponibles para búsquedas y filtrado en el navegador y en el panel *búsqueda de metadatos de capa*.
- Para el resto de fuentes de datos basadas en ficheros *Guardar en ubicación Predeterminada* guarda los metadatos en un fichero `.qmd` junto al fichero.
- En todos los demás casos *Guardar en ubicación Predeterminada* guarda los metadatos en una base de datos local `.sqlite`.

8.6.2 Notas de capa

Las notas de capa le permiten documentar la capa dentro del proyecto actual. Pueden ser lugar para almacenar mensajes importantes para los usuarios del proyecto como listas de tareas, instrucciones, advertencias, …

Desde el menú contextual de la capa en el panel *Layers*, seleccione *Add layer notes…* y llene el cuadro de diálogo abierto con los textos necesarios.

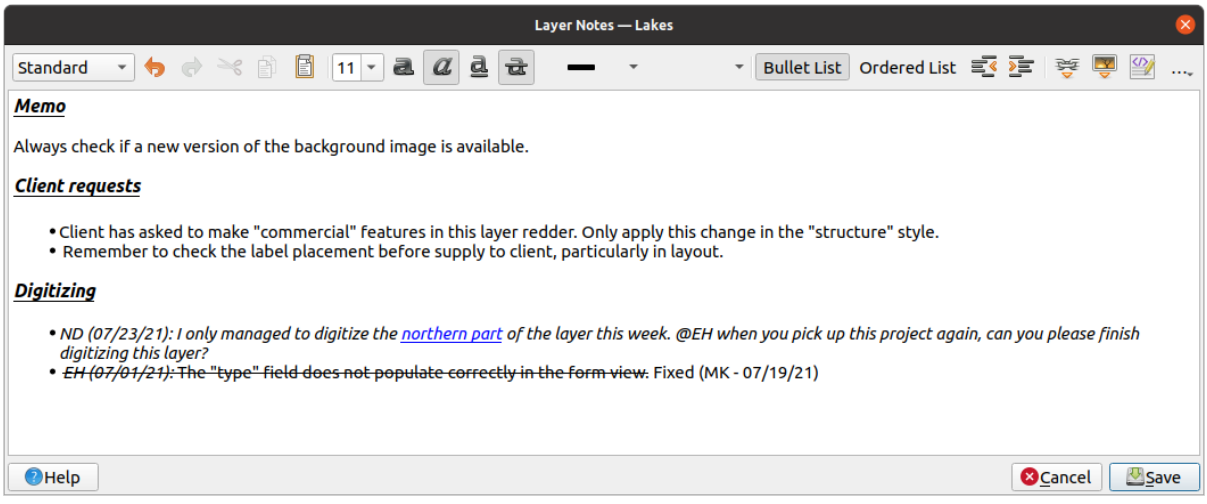


Figura 8.15: Agregar notas a una capa

El cuadro de diálogo *Add layer notes* proporciona un cuadro de texto multilínea basado en html con un conjunto completo de herramientas para:

- manipulación de texto: cortar, copiar, pegar, deshacer, rehacer
- formato de caracteres, aplicado a todo o parte del contenido: tamaño y color de fuente, negrita, cursiva, subrayado, tachado, color de fondo, resaltado de URL

- estructuración de párrafos: viñetas y listas numeradas, sangría, encabezados predefinidos
- inserción de archivos, incluso con arrastrar y soltar
- edición con codificación HTML

Desde el menú desplegable … en el extremo derecho de la barra de herramientas, puede:

- *Remover todo el formato*
- *Eliminar formato de caracteres*
- *Borrar todo el contenido*

En el panel *Layers*, a una capa con una nota se le asigna el  icono que, al pasar el mouse, muestra la nota. Haga clic en el icono para editar la nota. También puede hacer clic derecho en la capa y *Edit layer note…* o *Remove layer note*.

Nota

Las notas son parte del *layer style* y se pueden guardar en el archivo `.qml` o `.qlr`. También se pueden transferir de una capa a otra mientras se copia y pega el estilo de capa.

8.7 Almacenando valores en variables

En QGIS, puede usar variables para almacenar valores recurrentes útiles (por ejemplo, el título del proyecto o el nombre completo del usuario) que se pueden usar en expresiones. Las variables se pueden definir en el nivel global de la aplicación, el nivel del proyecto, el nivel de la capa, el nivel del modelador de procesamiento, el nivel del diseño y el nivel del elemento del diseño. Al igual que las reglas en cascada de CSS, las variables se pueden sobrescribir; por ejemplo, una variable de nivel de proyecto sobrescribirá cualquier variable de nivel global de aplicación configurada con el mismo nombre. Puede usar estas variables para crear cadenas de texto u otras expresiones personalizadas usando el carácter `@` antes del nombre de la variable. Por ejemplo, en el diseño de impresión creando una etiqueta con este contenido:

```
This map was made using QGIS [% @qgis_version %]. The project file for this
map is: [% @project_path %]
```

Representará la etiqueta como ésta:

```
This map was made using QGIS 3.4.4-Madeira. The project file for this map is:
/gis/qgis-user-conference-2019.qgs
```

Además de *preset read-only variables*, puede definir sus propias variables personalizadas para cualquiera de los niveles mencionados anteriormente. Puedes gestionar:

- **variables globales** desde el menú *Ajustes* ► *Opciones*
- **variables del proyecto** desde el diálogo *Propiedades del Proyecto* (Ver *Propiedades del proyecto*)
- **variables de capa vectorial** desde el diálogo *Propiedades de capa* (ver *El Diálogo de las Propiedades del Vector*);
- **variables del modelador** desde el diálogo *Diseñador de modelos* (véase *El diseñador de modelos*);
- **variables de diseño** desde el panel *Diseño* en el Diseñador de Impresión (ver *El Panel de Diseño*);
- y **variables de elementos de diseño** desde el panel *Propiedades de elementos* en el Diseñador de Impresión (ver *Opciones de Elementos comunes de Composición*).

Para diferenciarse de las variables editables, los nombres y valores de las variables de solo lectura se muestran en cursiva. Por otro lado, las variables de nivel superior sobrescritas por las de nivel inferior se tachan.

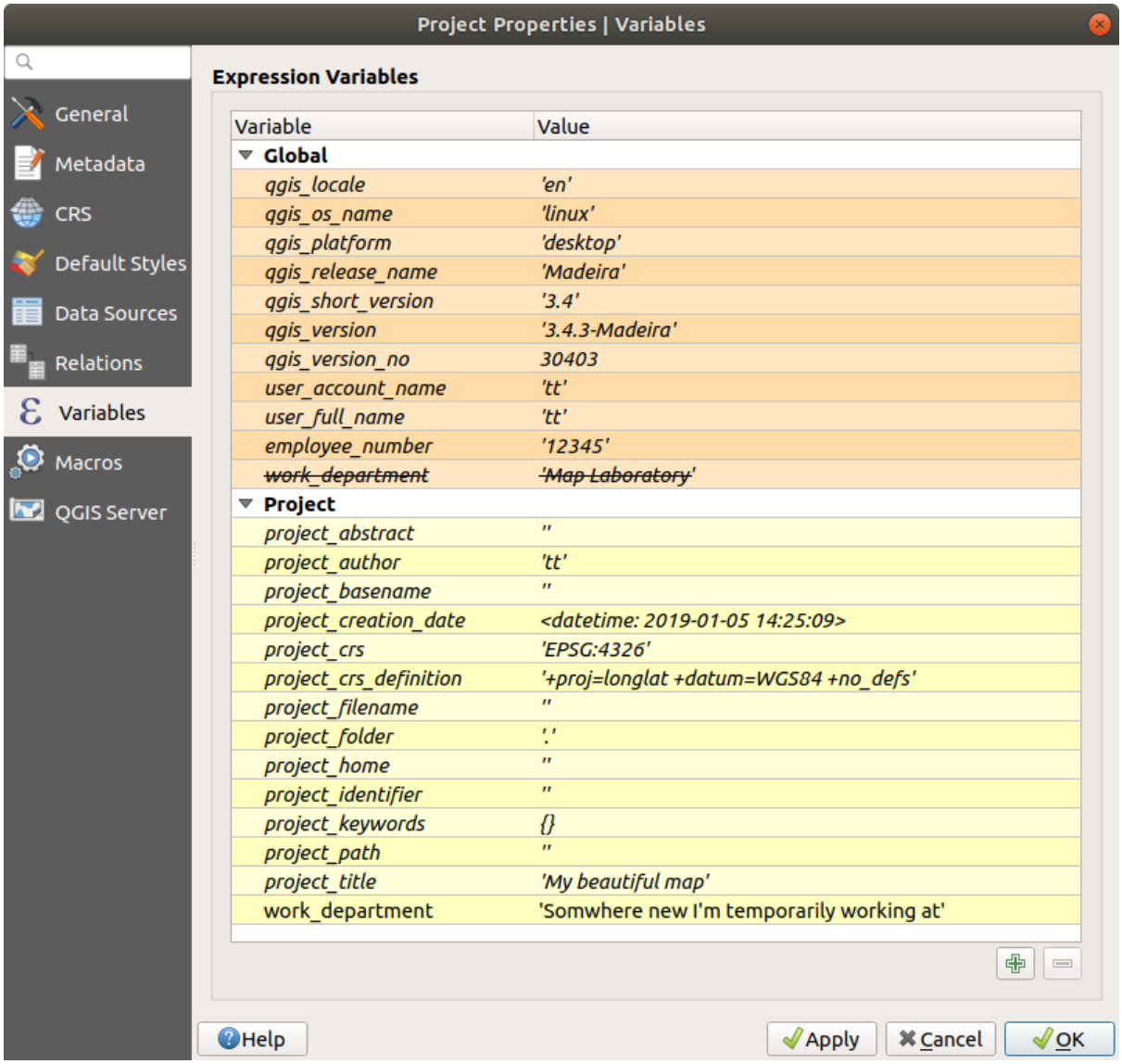


Figura 8.16: Editor de variables a nivel de proyecto

Nota

Puede leer más sobre variables y encontrar algunos ejemplos en las publicaciones del blog de Nyall Dawson's Explorando variables en QGIS 2.12, parte 1, parte 2 y parte 3 .

8.8 Widgets comunes

En QGIS, hay algunas opciones con las que a menudo tendrá que trabajar. Para mayor comodidad, QGIS le proporciona widgets especiales que se presentan a continuación.

8.8.1 Selector de color

El diálogo Color

El diálogo *Seleccionar color* aparecerá cada vez que haga click en  icono para elegir un color. Las características de este cuadro de diálogo dependen del estado de la casilla de verificación del parámetro *Usar cuadros de diálogo del selector de color nativo* en :menuselection:` Configuración -> Opciones … -> General`. Cuando se marca, el diálogo de color utilizado es el nativo del sistema operativo en el que se ejecuta QGIS. De lo contrario, se utiliza el selector de color personalizado QGIS.

El cuadro de diálogo de selección de color personalizado tiene cuatro pestañas diferentes que le permiten seleccionar colores mediante  Rampa de color,  Rueda de colores,  Muestras de color o  Selector de color. Con las dos primeras pestañas, puede buscar todas las combinaciones de colores posibles y aplicar su elección al elemento.

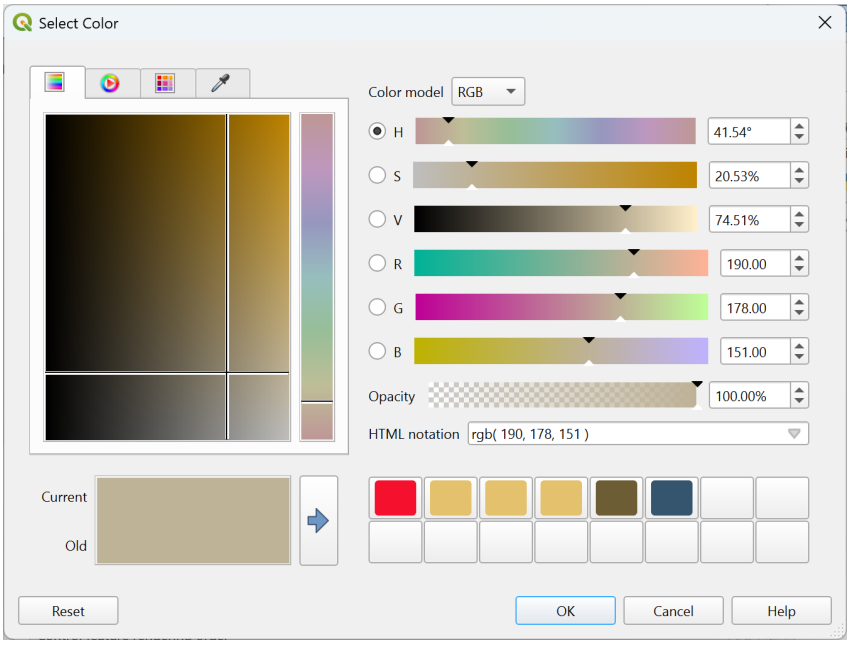


Figura 8.17: Pestaña de selección de rampa de color

En la pestaña  Color swatches, puedes elegir de una lista de paletas de colores (ver *Configuración de colores* para detalles). Todos menos la paleta *Colores recientes* se puede modificar con los botones  Agregar color actual y  Eliminar color seleccionado en la parte inferior del marco.

El botón … junto al cuadro combinado de paleta también ofrece varias opciones para:

- copiar, pegar, importar o exportar colores
- crear, importar o borrar paletas de color
- agregue la paleta personalizada al widget selector de color con el elemento *Mostrar en botones de color* (ver Figura 8.19)

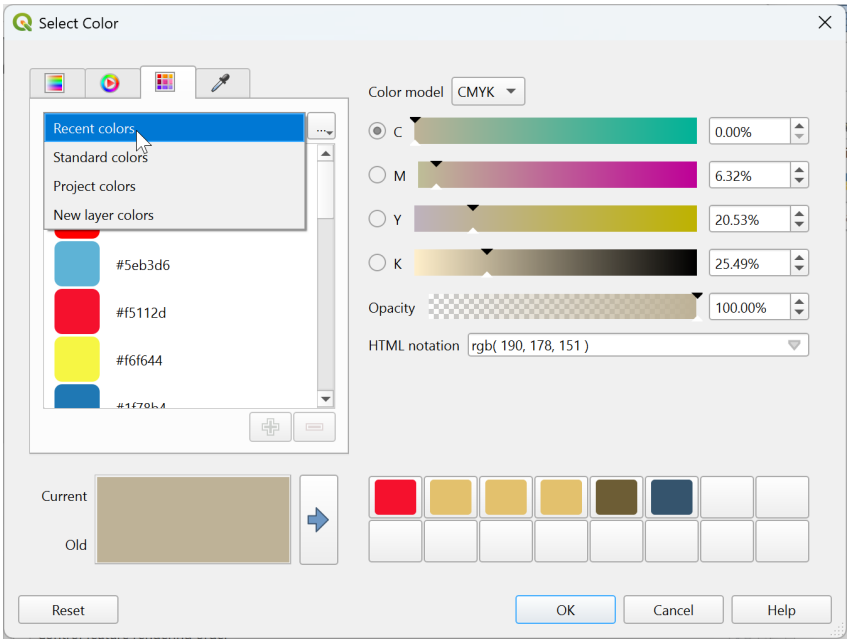


Figura 8.18: Pestaña de muestras del selector de color

Otra opción es usar el Selector de color que le permite muestrear un color debajo del cursor del mouse en cualquier parte de la interfaz de usuario de QGIS o incluso desde otra aplicación: presione la barra espaciadora mientras la pestaña está activa, mueva el mouse sobre el color deseado y haga click en él o presione la barra espaciadora nuevamente. También puede hacer click en el botón *Muestra de Color* para activar el selector.

Sea cual sea el método utilizado, el color seleccionado siempre se describe mediante opciones de un *Modelo de color* que por defecto es el *modelo de color del proyecto*:

- *RGB*: el color se ajusta utilizando deslizadores para los valores `HSV` (Tono, Saturación, Valor) o `RGB` (Rojo, Verde, Azul).
- *CMYK* (Cian, Magenta, Amarillo, Negro)

Modificar un color es tan simple como hacer click en la rueda de color o la rampa o en cualquiera de los controles deslizantes de parámetros de color. Puede ajustar dichos parámetros con el cuadro giratorio al lado o desplazando la rueda del mouse sobre el control deslizante correspondiente. También puede escribir el color en notación HTML. Finalmente, hay un control deslizante *Opacidad* para establecer el nivel de transparencia.

El cuadro de diálogo también proporciona una comparación visual entre el color *Viejo* (aplicado al objeto) y el *Actual* (siendo seleccionado). Usando arrastrar y soltar o presionando el botón Agregar color a tela, cualquiera de estos colores se puede guardar en una ranura para facilitar el acceso.

Truco

Modificación rápida de color

Arrastre y suelte un widget selector de color en otro para aplicar su color.

El atajo al desplegable de color

Haga click en la flecha desplegable a la derecha del botón  de color para mostrar un widget para una rápida selección de color. Este acceso directo proporciona acceso a:

- una rueda de colores para elegir un color
- un control deslizante alfa para cambiar la opacidad del color
- las paletas de colores previamente configuradas en *Mostrar en botones de colores*
- copia el color actual y pégalo en otro widget
- elige un color desde cualquier lugar de la pantalla de tu computadora
- elija un color del cuadro de diálogo selector de color
- arrastre y suelte el color de un widget a otro para una modificación rápida

Truco

Desplace la rueda del ratón sobre un widget selector de color para modificar rápidamente la opacidad del color asociado.

Nota

Cuando el widget de color se establece en un *project color* a través de las propiedades de anulación definidas por datos, las funciones anteriores para cambiar el color no están disponibles. Primero debe *Desvincular color* o *Borrar* la definición.

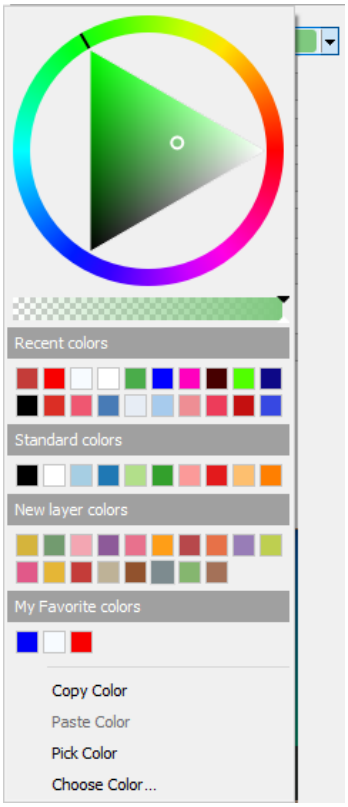


Figura 8.19: Menú seleccionador rápido de color

El atajo de la rampa de color desplegable

Las rampas de color son una forma práctica de aplicar un conjunto de colores a una o varias características. Su creación se describe en la sección *Estableciendo una Rampa de Color*.En cuanto a los colores, presionando el botón |selectColorRamp|de rampa de color abre el cuadro de diálogo correspondiente del tipo de rampa de color que le permite cambiar sus propiedades.

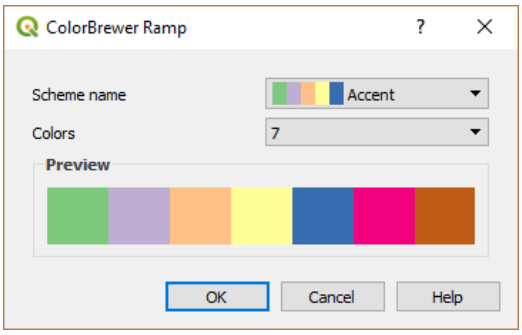


Figura 8.20: Personalizando una rampa colorbrewer

El menú desplegable a la derecha del botón brinda acceso rápido a un conjunto más amplio de rampas de color y opciones:

- *Invertir Rampa de Color*
- *Borrar rampa actual* para desactivar cualquier rampa de color asignada al widget (disponible sólo en algunos contextos)
- *Random Colors*: disponible solo en algunos contextos (p. ej., cuando se usa una rampa de color para una simbología de capa), al marcar esta entrada se crea y aplica una rampa de color con colores aleatorios. También habilita una entrada *Shuffle random colors* para regenerar una nueva rampa de color aleatoria si la actual no es satisfactoria.
- una vista previa de las rampas de color `gradiente` o `catálogo:cpt-city` marcadas como **Favoritos** en el diálogo *Administrador de Estilo*
- *Todas las rampas de color* para acceder a la base de datos de rampas de color compatibles
- *Crear nueva rampa de color…* de cualquier tipo compatible que pueda usarse en el widget actual (tenga en cuenta que esta rampa de color no estará disponible en otro lugar a menos que la guarde en la biblioteca)
- *Editar rampa de color…*, lo mismo que hacer click en el botón de rampa de color completo
- *Guardar rampa de color…*, para guardar la rampa de color actual con sus personalizaciones en la biblioteca de estilos

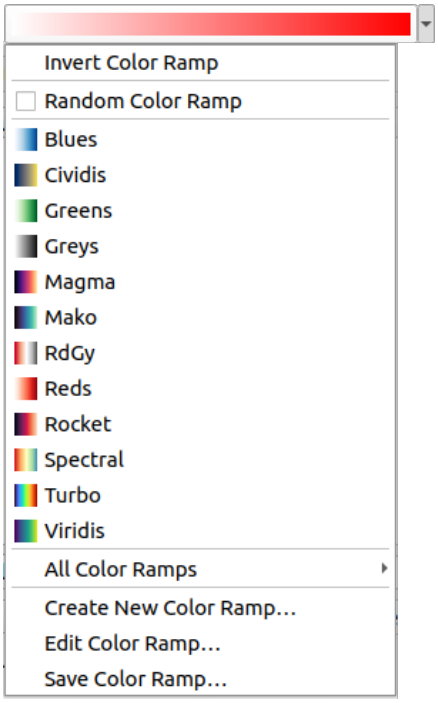


Figura 8.21: Widget de selección rápida de rampa de color

8.8.2 Widget Símbolo

El widget selector:guilabel:*Symbol* es un atajo conveniente cuando desea establecer las propiedades de símbolo de una característica. Al hacer click en la flecha desplegable, se muestran las siguientes opciones de símbolos, junto con las características de *color drop-down widget*:

- *Configurar símbolo…*: lo mismo que presionar el widget selector de símbolos. Abre un diálogo para configurar el *symbol parameters*.
- *Copiar símbolo* del elemento actual
- *Pegar símbolo* en el elemento actual, acelerando la configuración
- *Borrar símbolo actual* para anular cualquier símbolo asignado al widget (disponible sólo en algunos contextos)

Truco

Desplace la rueda del ratón sobre un widget de marcador o símbolo de línea para modificar rápidamente el tamaño del símbolo asociado.

8.8.3 Selector de archivos remoto o incrustado

Junto con el widget de selección de archivos, el botón … a veces mostrará una flecha desplegable. Esto suele estar disponible cuando se utiliza:

- un archivo SVG en un símbolo o una etiqueta
- una imagen rasterizada para personalizar símbolos, etiquetas, texturas o decoraciones

Presionar la flecha le proporcionará un menú para:

- *Seleccionar Archivo…*, para cargar el archivo desde el sistema de archivos. El archivo se identifica a través de la ruta del archivo y QGIS necesita resolver la ruta para mostrar la imagen correspondiente.

- *Desde URL…*, para cargar el archivo desde una URL remota. Como en el caso anterior, la imagen sólo se cargará cuando se recupere correctamente el recurso remoto.
- *Incorporar archivo…*, para incrustar el archivo dentro del proyecto actual, la base de datos de estilos o la plantilla de diseño de impresión. El archivo siempre se muestra como parte del elemento. Esta es una forma conveniente de crear proyectos autónomos con símbolos personalizados que pueden ser fácilmente compartidos entre diferentes usuarios e instalaciones de QGIS.
- *Extraer archivo incrustado…*, para extraer el archivo incrustado del widget y guardarlo en disco.

8.8.4 Selector de escala de visibilidad

El selector de escala de visibilidad proporciona opciones para controlar las escalas a las que un elemento se hará visible en el lienzo del mapa. Fuera del rango de escalas especificado, los elementos no se muestran. Puede aplicarse, por ejemplo, a capas, etiquetas o diagramas, desde su pestaña de propiedades *Renderizado*.

1. Marque la casilla *Visibilidad dependiente de la escala*.
2. Rellene la caja *Mínimo (exclusivo)* con la escala deseada más alejada tecleando el valor o seleccionándolo de las *escalas predefinidas*
3. y/o rellene la casilla *Máximo (inclusive)* con la escala deseada más ampliada

   El botón Set to current canvas scale situado junto a los cuadros de escala establece la escala actual del lienzo del mapa como límite de la visibilidad del rango. Pulse la flecha situada junto al botón para acceder a las escalas de los mapas de los diseños y reutilizarlas para rellenar el cuadro.

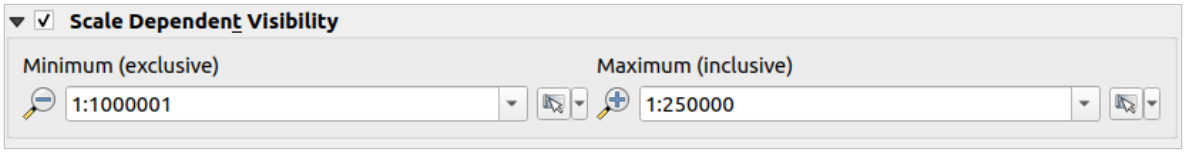


Figura 8.22: Widget selector de escala de visibilidad

8.8.5 Selector de extensión espacial

El widget selector *Extent* es un atajo conveniente cuando desea seleccionar una extensión espacial para asignar a una capa o para limitar las acciones a ejecutar. Dependiendo del contexto, ofrece la posibilidad de elegir entre:

- *Extensión actual de la capa*: por ejemplo, al exportar una capa.
- *Calcular a partir de capa* ►: utiliza la extensión de una capa cargada en el proyecto actual
- Utilizar *Extensión del lienzo del mapa* actual
- *Dibujar sobre lienzo*: un rectángulo cuyas coordenadas se utilizan a continuación
- *Calcular desde Marcador*: usa la extensión de un *marcador* guardado
- *Calcular desde el Diseño de Mapa*: usa la extensión de un *diseño de mapa*
- Introduzca o edite las coordenadas como `xmin, xmax, ymin, ymax`

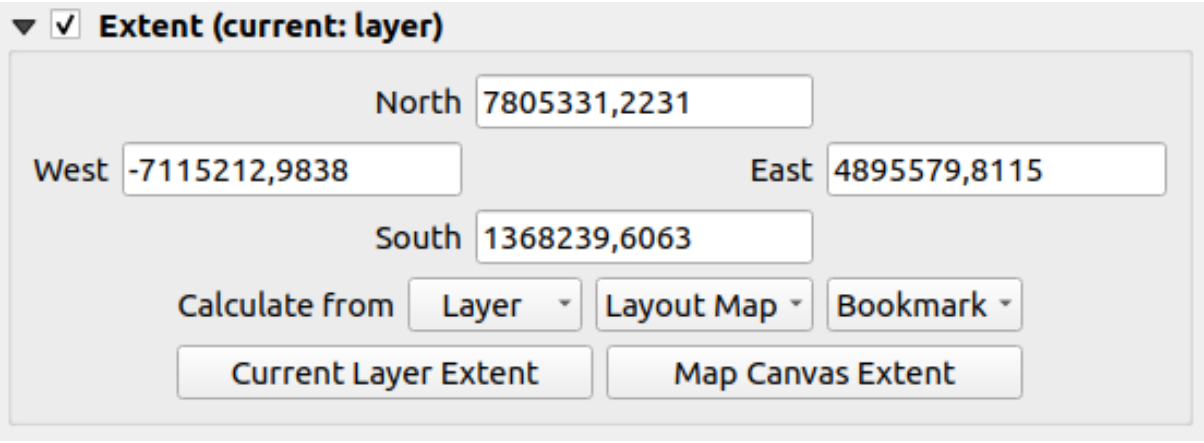


Figura 8.23: Widget selector de extensión

8.8.6 Selector de Fuente

El control selector de *Fuente* es un atajo conveniente cuando desea establecer propiedades de fuente para información textual (etiquetas de entidades, etiquetas de decoración, texto de leyenda de mapa, …). Al hacer clic en la flecha desplegable, se muestran algunas o todas las siguientes opciones:



Figura 8.24: Menú desplegable de selección de fuente

- *Borrar formato de texto actual* para desactivar cualquier formato de texto asignado al widget (disponible sólo en algunos contextos)
- *Tamaño de fuente* en la unidad asociada
- *Fuentes recientes* ► menú con la fuente activa marcada (en la parte superior)
- *Configurar formato…*: igual que pulsar el widget selector de fuentes. Abre el diálogo *Formato de texto*, que proporciona opciones avanzadas de formato como color, opacidad, orientación, notación HTML, búfer, fondo,

sombra, …

- *Copiar formato* del texto
- *Pegar Formato* al texto, acelerando la configuración
- el *control de color* para configuración rápida del color

Truco

Desplace la rueda del ratón sobre un widget selector de fuentes para modificar rápidamente el tamaño de la fuente del texto asociado.

8.8.7 Selector de Unidad

Las propiedades de tamaño de los elementos (etiquetas, símbolos, elementos de diseño, …) en QGIS no están necesariamente vinculadas ni a las unidades del proyecto ni a las unidades de una capa en particular. Para un gran conjunto de propiedades, el menú desplegable:guilabel:*Unidad* le permite ajustar sus valores de acuerdo con la representación que desee (según la resolución de la pantalla, el tamaño del papel o el terreno). Las unidades disponibles son:

- *Milímetros*
- *Puntos*
- *Pixels*
- *Pulgadas*
- *Porcentaje*: permite establecer algunas propiedades como un porcentaje de otra. Por ejemplo, esto es útil para la creación de formatos de texto en los que los componentes (tamaño del búfer, radio de la sombra…) se escalan a medida que cambia el tamaño del texto, en lugar de tener tamaños de búfer/sombra constantes. Así no es necesario ajustar esos tamaños cuando cambia el tamaño del texto.
- *Metros en escala*: Permite establecer siempre el tamaño en metros, independientemente de cuáles sean las unidades subyacentes del mapa (por ejemplo, pueden estar en pulgadas, pies, grados geográficos, …). El tamaño en metros se calcula en base a la configuración actual del elipsoide del proyecto y una proyección de las distancias en metros en el centro de la extensión actual del mapa. Para los mapas en un sistema de coordenadas proyectadas se calcula utilizando unidades proyectadas. Para los mapas en un sistema geográfico (latitud/longitud) el tamaño se aproxima calculando el tamaño en metros utilizando cálculos elipsoidales para la escala vertical del mapa.
- y *Unidades de mapa*: el tamaño se escala según la escala de la vista del mapa. Como esto puede conducir a valores demasiado grandes o demasiado pequeños, use el botón 🔧 al lado de la entrada para restringir el tamaño a un rango de valores basado en:
  - El *Escala mínima* y el :guilabel:` Escala máxima`: el valor se escala en función de la escala de la vista del mapa hasta llegar a cualquiera de estos límites de escala. Fuera del rango de escala, se mantiene el valor en el límite de escala más cercano.
  - y/o El *Tamaño mínimo* y el :guilabel:` Tamaño máximo` en `mm`: El valor se escala en función de la escala de la vista del mapa hasta que alcanza cualquiera de estos límites; Luego se mantiene el tamaño límite.

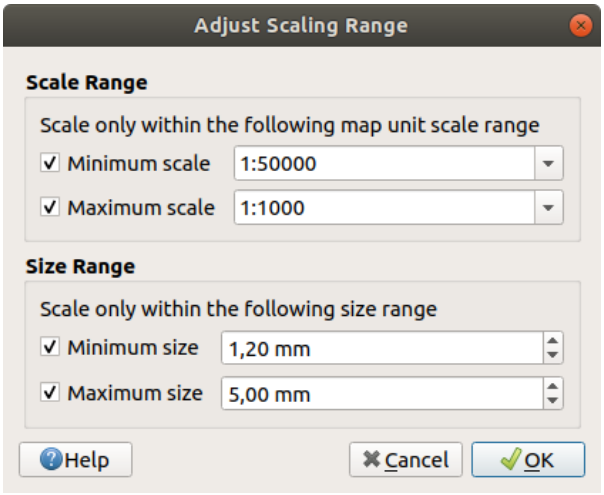


Figura 8.25: Cuadro de diálogo Ajustar rango de escala

8.8.8 Formato de Números

Los formateadores numéricos permiten formatear valores numéricos para su visualización, utilizando una variedad de técnicas de formateo diferentes (por ejemplo, notación científica, valores monetarios, valores porcentuales, formateo personalizado, etc.). Por ejemplo, para definir el texto de una barra o tabla de escala, una leyenda de rampa de color, un gráfico de perfil de elevación, etc.

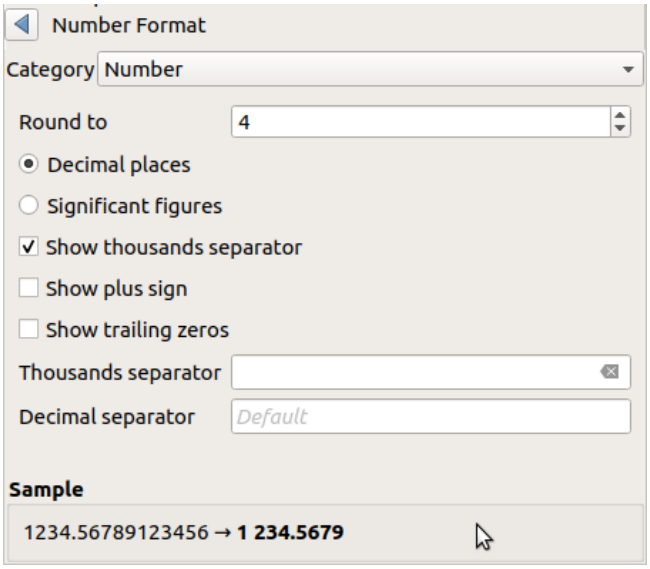


Figura 8.26: Formatting numeric value

Se admiten diferentes categorías de formatos. Para la mayoría de ellos, puede establecer parte o todas las siguientes opciones numéricas:

- *Mostrar separador de miles*
- *Mostrar símbolo mas*
- *Mostrar ceros finales*

Pero también pueden tener sus configuraciones personalizadas. Las categorías proporcionadas son:

- *General*, la categoría por defecto: no tiene configuración y muestra los valores establecidos en las propiedades del widget principal o usando la configuración global.
- *Expresión personalizada*: permite crear una *expresión QGIS* personalizada para formatear números. La expresión puede utilizar la variable `@value` para recuperar el valor a formatear y, a continuación, utilizar cualquier función de expresión estándar de QGIS para darle el formato deseado.
- *Número*
  - El valor puede ser *Redondeado a* un número autodefinido de *posiciones Decimales* o su *cifras significativas*
  - personalice el *separador de Miles* y el *separador Decimal*
- *Rumbo* para una representación textual de una dirección/rumbo usando:
  - *Formato*: posibles rangos de valores son `0 a 180°, con sufijo E/W`, `-180 a +180°` y `0 a 360°`
  - numero de *cifras Decimales*
- *Moneda* para una representación de texto de un valor de moneda.
  - *Prefijo*
  - *Sufijo*
  - numero de *cifras Decimales*
- *Fracción* para una representación fraccional común de un valor decimal (e.g. *1/2* en lugar de *0.5*)
  - ☐ *Usar unicode super/subíndice* a mostrar. Por ejemplo $^{1}/_{2}$ en lugar de 1/2
  - ☐ *Usar códigos Unicode dedicados*
  - personalizar el *separador de Miles*
- *Porcentage* - agrega `%` a los valores, con ajuste de:
  - numero de *cifras Decimales*
  - *Escalado* para indicar si los valores reales ya representan porcentajes (luego se mantendrán como están) o fracciones (luego se convertirán)
- Notación *Científica* de la forma `2.56e+03`. El número de *cifras Decimales* puede ser ajustado.

Se muestra una vista previa en vivo de la configuración debajo de la muestra *Muestra*.

8.8.9 Modos de Mezcla

QGIS ofrece diferentes opciones para efectos de renderizado especiales con estas herramientas que anteriormente solo sabías de los programas gráficos. Los modos de fusión se pueden aplicar en capas y entidades, y también en elementos de diseño de impresión:

- *Normal*: Este es el modo de mezcla estándar, que utiliza el canal alfa del píxel superior para mezclarse con el píxel que tiene debajo. Los colores no se mezclan.
- *Aclarar*: Selecciona el máximo de cada componente de los píxeles de primer plano y de fondo. Tenga en cuenta que los resultados tienden a ser irregulares y duros.
- *Pantalla*: Los píxeles claros del origen se pintan sobre el destino, mientras que los oscuros no. Este modo es muy útil para mezclar la textura de un elemento con la de otro elemento (como usar una sombra de colina para texturizar otra capa).
- *Esquivar*: Aclara y satura los píxeles subyacentes basándose en la luminosidad del píxel superior. Los píxeles superiores más brillantes hacen que aumente la saturación y el brillo de los píxeles subyacentes. Esto funciona mejor si los píxeles superiores no son demasiado brillantes. De lo contrario, el efecto es demasiado extremo.
- *Adición*: Añade los valores de píxel de un elemento al otro. En caso de valores superiores al valor máximo (en el caso de RGB), se muestra el color blanco. Este modo es adecuado para resaltar elementos.

- *Oscurecer*: Conserva los valores más bajos de cada componente de los píxeles de primer plano y de fondo. Al igual que aclarar, los resultados tienden a ser irregulares y duros.
- *Multiplicar*: Los valores de píxel del elemento superior se multiplican por los valores correspondientes del elemento inferior. Los resultados son más oscuros.
- *Quemar*: Los colores más oscuros en el elemento superior hacen que los elementos subyacentes se oscurezcan. Quemar puede usarse para retocar y colorear las capas subyacentes.
- *Superposición*: Combina los modos multiplicar y mezcla de pantalla. Las partes claras se aclaran y las oscuras se oscurecen.
- *Luz suave*: Muy similar a la superposición, pero en lugar de usar multiplicar/pantalla usa quemar/doblar color. Se supone que emula el reflejo de una luz suave sobre una imagen.
- *Luz dura*: La luz dura también es muy similar al modo de superposición. Se supone que emula la proyección de una luz muy intensa sobre una imagen.
- *Diferencia*: Resta el píxel superior del inferior, o al revés, para obtener siempre un valor positivo. La mezcla con negro no produce ningún cambio, ya que la diferencia con todos los colores es cero.
- *Sustraer*: Resta los valores de píxel de un elemento del otro. En caso de valores negativos, se muestra en negro.

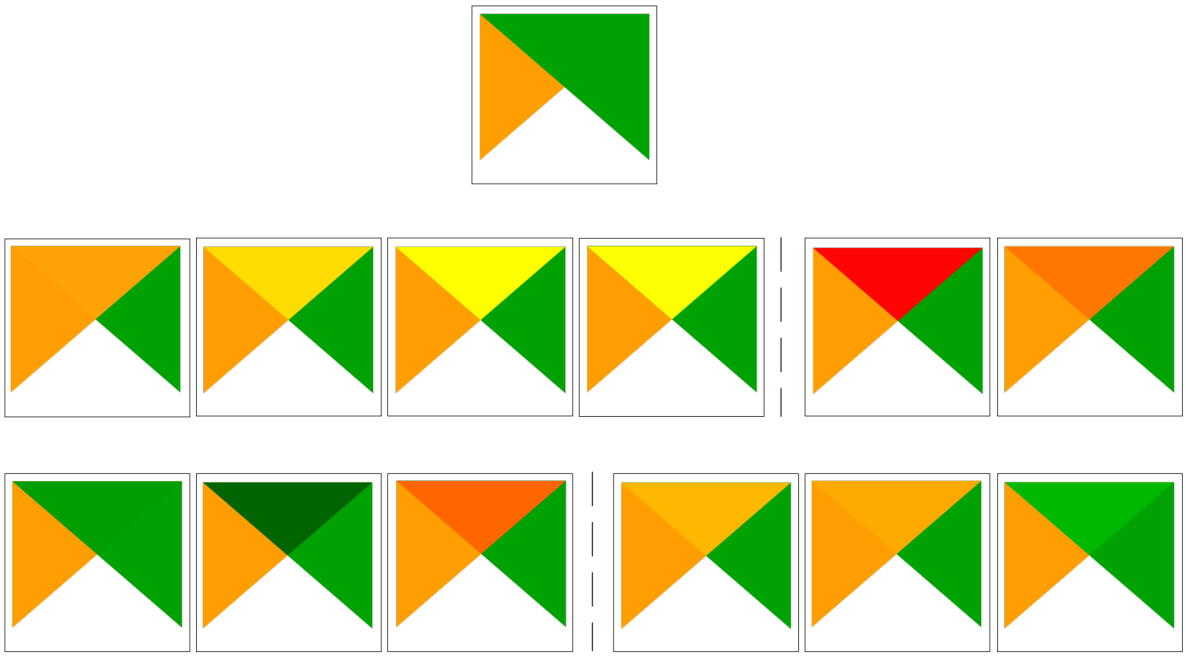


Figura 8.27: Ejemplos de modos de fusión aplicados a un elemento verde sobre uno naranja
De arriba a abajo, de izquierda a derecha: Normal – Aclarar, Trama, Esquivar, Suma – Diferencia, Resta – Oscurecer, Multiplicar, Quemar – Superponer, Luz suave, Luz dura

Cuando una capa forma parte de un grupo que *renderiza las capas como un grupo*, Existen modos de fusión adicionales para el renderizado. Proporcionan métodos para recortar el renderizado del contenido de una capa por el contenido de una segunda capa «máscara».

- *Enmascarado Por Debajo*: La salida es el píxel superior, donde la opacidad se reduce en la del píxel inferior.
- *Máscara inferior*: La salida es el píxel inferior, donde la opacidad se reduce en la del píxel superior.
- *Inverso Enmascarado Por Debajo*: La salida es el píxel superior, donde la opacidad se reduce en el inverso del píxel inferior.
- *Máscara Inversa Abajo*: La salida es el píxel inferior, donde la opacidad se reduce en el inverso del píxel superior.

- *Pintar dentro de abajo*: El píxel superior se mezcla sobre el píxel inferior, con la opacidad del píxel superior reducida por la opacidad del píxel inferior.
- *Pintar por debajo del interior*: El píxel inferior se mezcla encima del píxel superior, con la opacidad del píxel inferior reducida por la opacidad del píxel superior.

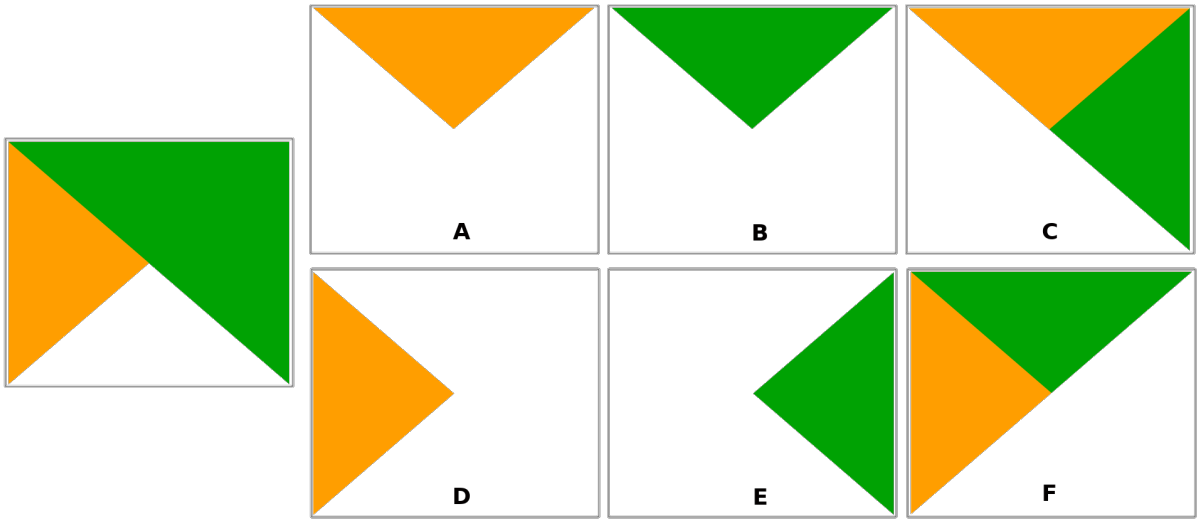


Figura 8.28: Ejemplos de modo de recorte de fusión aplicado a la capa verde superior de un grupo
A: Enmascarado Por Abajo **B**: Enmascarar Por Debajo **C**: Pintar Por Dentro **D**: Máscara Inversa Abajo **E**: Inverso Enmascarado Por Debajo **F**: Pintar dentro de abajo

8.8.10 Configuración de anulación definida por datos

Junto a muchas opciones en el cuadro de diálogo de propiedades de capa vectorial o en la configuración del diseño de impresión, encontrará un icono Anulación de datos definidos. Utilizando *expressions* basado en atributos de capa o configuraciones de elementos, funciones preconstruidas o personalizadas y *variables*, Esta herramienta le permite establecer valores dinámicos para los parámetros. Cuando está habilitado, el valor devuelto por este widget se aplica al parámetro independientemente de su valor normal (casilla de verificación, cuadro de texto, control deslizante …).

El widget de anulación de datos definido

Al hacer click en el icono Anulación de datos definidos muestra las siguientes entradas:

- *Descripción …* que indica si la opción está habilitada, qué entrada se espera, el tipo de entrada válido y la definición actual. Al pasar el cursor sobre el widget también aparece esta información.
- *Almacenar datos en el proyecto*: un botón que permite almacenar la propiedad utilizando el mecanismo *Propiedades de almacenamiento auxiliar*.
- *Tipo de campo*: una entrada para seleccionar de los campos de la capa que coinciden con el tipo de entrada válido.
- *Color*: cuando el widget está vinculado a una propiedad de color, este menú da acceso a los colores definidos como parte del actual *project's colors* scheme.
- Un menú *Variable*: para acceder a los disponibles definidos por el usuario *variables*
- El botón *Editar …* para crear o editar la expresión a aplicar, usando el diálogo *Generador de cadenas de expresión*. Para ayudarlo a completar correctamente la expresión, se proporciona un recordatorio del formato de salida esperado en el cuadro de diálogo.

- Los botones *Pegar* y :guilabel:` Copiar`.
- botón de *Limpiar* para eliminar la configuración.
- Para propiedades numéricas y de color,:guilabel:*Assistente…* para reescalar cómo se aplican los datos de la característica a la propiedad (más detalles *below*)

Truco

Use el botón derecho para (des)activar la anulación de datos

Cuando la opción de anulación definida por datos se configura correctamente, el icono es amarillo o . Si está roto, el icono es rojo o .

Puede habilitar o deshabilitar un botón configurado anulación definida por datos simplemente haciendo click en el widget con el botón derecho del mouse.

Truco

Utilice el botón central del ratón para crear o editar la expresión a aplicar.

Puede abrir directamente el diálogo *Constructor de cadenas de expresión* para crear o editar la expresión a aplicar simplemente haciendo clic con el botón central del ratón en el widget Sobreescritura definida por datos.

Usar la interfaz de asistente definida por datos

When the Data-defined override button is associated with a size, a width, a rotation, an opacity or a color property, it has an *Assistant…* option that helps you change how the data is applied to the parameter for each feature. Basically, QGIS will scale a range of input values over a range of output values, and render each feature with their corresponding ouput value.

The assistant allows you to:

- Define the *Input* data, i.e.:
  - *Source*: el atributo a representar, usando un campo o un *expression*
  - the range of values to represent: you can manually enter the values or use the Fetch value range from layer button to fill these fields automatically with the minimum and maximum values returned by the *Source* expression applied to your data. Features whose values are out of the given range will be represented with the property of the closest limit of the output range.
- *Apply transform curve*: by default, output values are applied to input features following a specific scaling method (see below for setting). You can override this logic: enable the transform option, click on the graphic to add break point(s) and drag the point(s) to apply a custom distribution.
- Defina los valores *Salida*: las opciones varían según el parámetro a definir. Puede establecer globalmente:
  - for color: a linear interpolation of source values is done over a given *color ramp* in order to assign each symbol the correct color. A color to use for symbols with NULL values can also be given.
  - for rotation: a linear interpolation is done over a range of minimum and maximum angles in order to assign each symbol the correct rotation. An angle to use for symbols with NULL values can also be given.
  - for opacity: a polynomial interpolation is done over a range of minimum and maximum opacity values. The *Exponent* field dictates the way source values are mapped to the output range. Power values greater than `1` will cause the output values distribution to start slowly before accelerating as the source values

approach their maximum. Smaller exponents (less than `1`) will do the opposite. An opacity value to use for symbols with NULL values can also be given.

- for width (line symbols): an interpolation is done over a range of minimum and maximum values in order to assign each symbol the correct width. The *Scale method* field controls how the interpolation is done and what the output value represents.
  - **Linear**: linearly interpolates the width of the symbol between the *Size from* and the *Size to* when the *Source* expression value varies from *value from* to *value to*. Symbols width are proportional.
  - **Exponential**: Also interpolates the width, but allows to control the exact power used for scaling through the *Exponent* field. Power values greater than `1` will cause the output values distribution to start slowly before accelerating as the source values approach their maximum. Smaller exponents (less than `1`) will do the opposite.

  The width to use for symbols with NULL values can also be given.
- for size (point symbols): an interpolation is done over a range of minimum and maximum values in order to assign each symbol the correct size «diameter». The *Scale method* field controls how the interpolation is done and what the output value represents.
  - **Radius**: linearly interpolates the diameter of the symbol between the *Size from* and the *Size to* when the *Source* expression value varies from *value from* to *value to*. Symbols diameter are proportional.
  - **Surface**: scales the symbol diameter according to the square root (power of `0.5`) of the *Source* expression value to linearly interpolate the area of the symbol between the *Size from* and the *Size to* when the *Source* expression value varies from *value from* to *value to*. Symbols area are proportional. This method reflects human perception of size variation better than the **radius** one.
  - **Flannery**: Applies a compensation factor (power of `0.57`) to the area interpolation, in order to mitigate the eyes" perceptual underestimation of proportional areas. Maybe relevant only for circular symbols.
  - **Exponential**: Also interpolates the area, but allows to control the exact power used for scaling through the *Exponent* field. Power values greater than `1` will cause the output values distribution to start slowly before accelerating as the source values approach their maximum. Smaller exponents (less than `1`) will do the opposite.

  The size to use for symbols with NULL values can also be given.

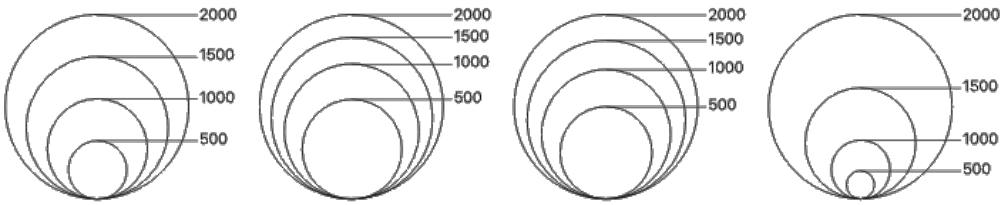


Figura 8.29: Scaling values using different methods (from left to right): Radius - Surface - Flannery - Exponential of 2

Cuando es compatible con la propiedad, se muestra una vista previa de actualización en vivo en el lado derecho del cuadro de diálogo para ayudarlo a controlar la escala del valor.

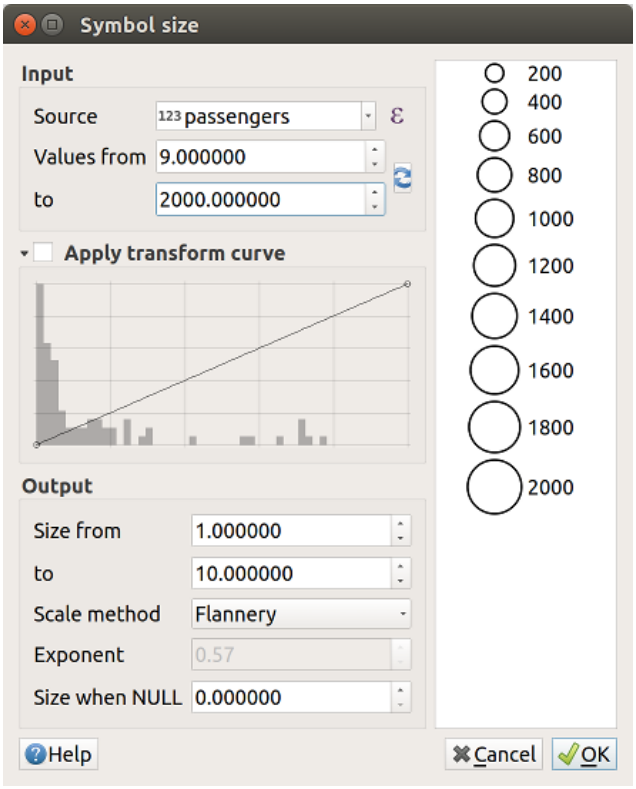


Figura 8.30: Tamaño de función de escala basado en el valor del campo de pasajeros

The values presented in the varying size assistant above will set the size “Data-defined override” with the editable formula below:

```
coalesce(scale_polynomial("passengers", 9, 2000, 1, 10, 0.57), 0)
```

8.8.11 Autenticación

Many data providers in QGIS (such as databases, web services or file-based services) require credentials in order to connect. The authentication widget in QGIS helps users supply these credentials securely, and it offers two ways to do so:

- entering basic credentials directly,
- or selecting an existing authentication configuration stored in the encrypted authentication database.

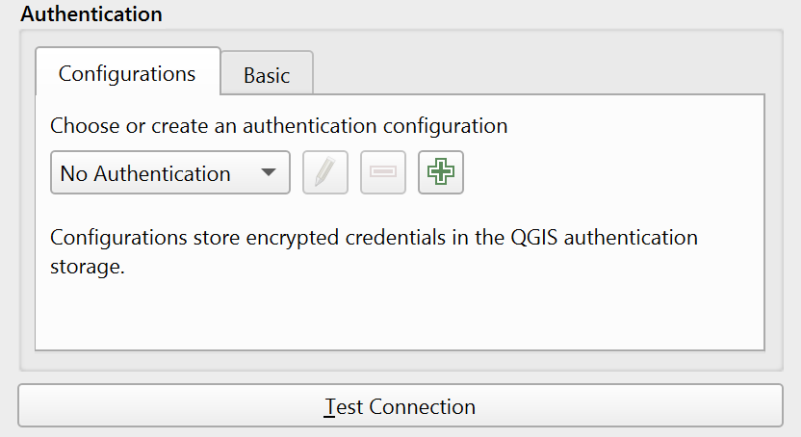


Figura 8.31: Authentication widget

Basic

When setting authentication credentials, the *Basic* tab allows you to set:

- *User name*: User name used to connect to the data source.
- *Password*: Password used to connect to the data source.

Convert to configuration will save the user name and password in a new authentication configuration.

You can save any or both of the `User name` and `Password` parameters, in which case they will be used by default each time you need to connect to this database or service. If not saved, you'll be prompted to supply the credentials to connect to the database or service in next QGIS sessions. You can cancel the prompt, in which case the connection will fail. Click the *Ignore for session* button to avoid being prompted again during the current QGIS session. It is also possible to *Ignore for 10 seconds*. This will cancel the prompt but you need to wait 10 seconds before you can attempt to connect again. The connection parameters you entered are stored in a temporary internal cache and returned whenever a username/password for the same database or service is requested, until you end the current QGIS session.

Advertencia

QGIS User Settings and Security

In the *Authentication* tab, saving **username** and **password** will keep unprotected credentials in the connection configuration. Those **credentials will be visible** if, for instance, you share the project file with someone. Therefore, it is advisable to save your credentials in an *Authentication configuration* instead (*Configurations* tab - See *Sistema de autenticación* for more details) or in a service connection file (see *PostgreSQL Service connection file* for example).

Configurations

QGIS can store credentials under *authentication configurations*. Each configuration safely saves your login details in a portable encrypted database, so you can use them again without typing them every time. These configurations can be safely referenced by their ID tokens in projects and settings.

The authentication configuration selector allows users to choose from existing authentication configurations or create new ones. It provides options to:

- Select an existing authentication configuration from a drop-down list.
- Create a new authentication configuration

-  Delete selected configuration
-  Edit selected configuration

For more information on the authentication system, see *Sistema de autenticación*.

CAPÍTULO 9

Sube de nivel con Expresiones

9.1 Expresiones

En base a datos de capa y funciones predefinidas o definidas por el usuario, **Expresiones** ofrece una manera poderosa de manipular el valor de un atributo, la geometría y las variables con el objetivo de cambiar dinámicamente el estilo de geometría, el contenido o posición de la etiqueta, el valor para el diagrama, la altura de un elemento de diseño, seleccionar algunas funciones, crear campo virtual, …

Nota

Puede encontrar una lista de las funciones y variables predeterminadas para escribir expresiones en *Lista de funciones*, con información detallada y ejemplos.

9.1.1 El constructor de cadena de Expresión

El diálogo principal para construir expresiones, el *Expression string builder* está disponible desde muchas partes en QGIS y, en particular, se puede acceder mediante:

- clickando el botón ;
- *selecting features* con la herramienta Select By Expression…;
- *editing attributes* con p.ej. la herramienta Field calculator;
- manipular parámetros de simbología, etiqueta o elemento de diseño con la herramienta Data defined override (ver *Configuración de anulación definida por datos*);
- construyendo una *geometry generator* symbol layer;
- haciendo algún *geoprocessing*.

El cuadro de diálogo del generador de expresiones ofrece acceso a:

- pestaña *Expresión* que, gracias a una lista de *funciones predefinidas*, ayuda a escribir y comprobar la expresión a utilizar;
- *Function Editor tab* lo que ayuda a ampliar la lista de funciones mediante la creación de funciones personalizadas.

La Interfaz

La pestaña *Expresión* proporciona la interfaz principal para escribir expresiones usando funciones, campos de capa y valores. Contiene los siguientes controles:

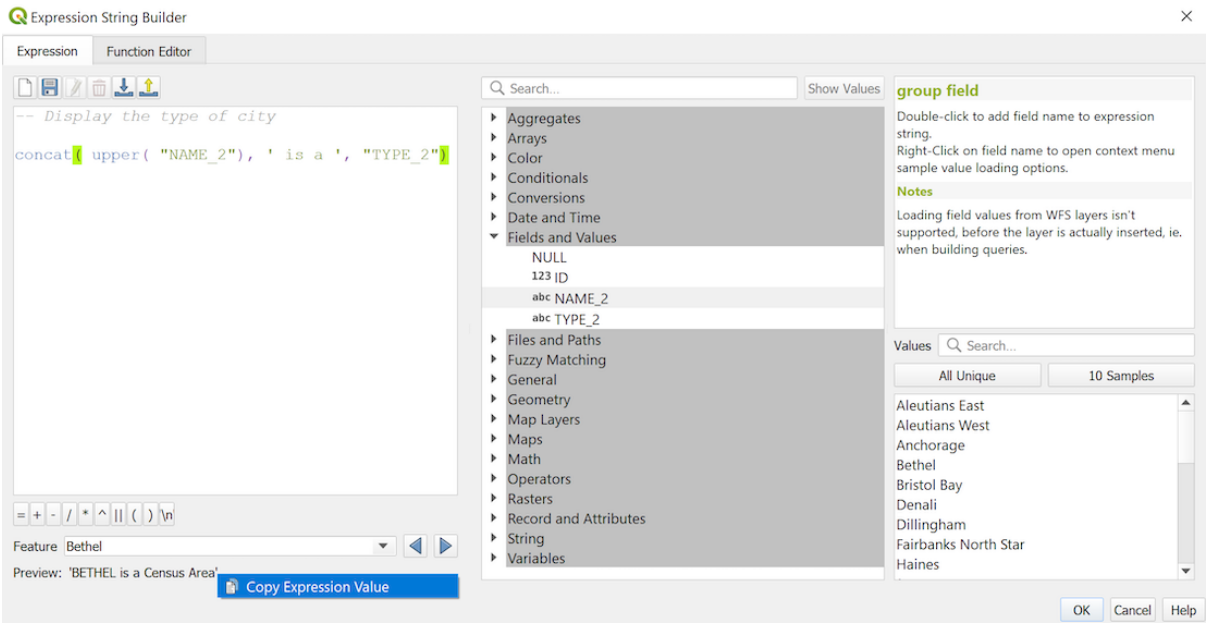


Figura 9.1: La pestaña Expresión

- Un área de editor de expresiones para escribir o pegar expresiones. Está disponible el autcompletado para acelerar la escritura de la expresión:
  - Las variables correspondientes, los nombres de las funciones y los nombres de campo para el texto de entrada se muestran a continuación: use las flechas `Up` y `Down` para examinar los elementos y presione `Tab` para insertar en la expresión o simplemente haga clic en el elemento deseado.
  - Los parámetros de la función son mostrados mientras los rellena.

  QGIS también verifica la corrección de la expresión y resalta todos los errores usando:
  - *Underline*: para funciones desconocidas, argumentos incorrectos o inválidos;
  - *Marker*: para cualquier otro error (por ejemplo, paréntesis faltante, carácter inesperado) en una sola ubicación.

> **Truco**
>
> **Documenta tu expresion con comentarios**
>
> Cuando se utiliza una expresión compleja, es una buena práctica agregar texto como un comentario de varias líneas o comentarios entrelíneas para ayudarlo a recordar.
>
> ```
> /*
> Labels each region with its highest (in altitude) airport(s)
> and altitude, eg 'AMBLER : 264m' for the 'Northwest Artic' region
> */
> ```

```
with_variable(
  'airport_alti', -- stores the highest altitude of the region
  aggregate(
    'airports',
    'max',
    "ELEV", -- the field containing the altitude
    -- and limit the airports to the region they are within
    filter := within( @geometry, geometry( @parent ) )
  ),
    aggregate( -- finds airports at the same altitude in the region
      'airports',
      'concatenate',
      "NAME",
      filter := within( @geometry, geometry( @parent ) )
        and "ELEV" = @airport_alti
    )
    || ' : ' || @airport_alti || 'm'
    -- using || allows regions without airports to be skipped
)
```

- Sobre el editor de expresión, un conjunto de herramientas le ayuda:
  -  Limpiar el editor de expresión
  - Crear y gestionar *expresiones de usuario*
- Bajo el editor de expresión, encontrará:
  - Un conjunto de operadores básicos para ayudarle a construir la expresión
  - Una indicación del formato de salida esperado cuando se definen los datos de las propiedades de las características
  - Una *Vista previa de salida* de la expresión (hasta 60 caracteres), evaluada en la primera característica de la Capa por defecto. Para ver el texto de la vista previa de salida de más de 60 caracteres, puede situar el cursor sobre el texto para que aparezca una ventana emergente con toda la vista previa de salida. Para copiar el texto de la vista previa de salida en el portapapeles, haga clic con el botón derecho en el texto de la vista previa de salida y seleccione **|editarCopiar|** *Copiar valor de expresión*.

    Puede examinar y evaluar otras características de la capa utilizando el cuadro combinado *Feature* (los valores se toman de la propiedad de la capa *nombre para mostrar*).

    En caso de error, se lo indica y puede acceder a los detalles con el hiperenlace proporcionado.
- Un selector de funciones muestra la lista de funciones, variables, campos … organizados en grupos. Hay un cuadro de búsqueda disponible para filtrar la lista y encontrar rápidamente una función o campo en particular. Al hacer doble clic en un elemento, se agrega al editor de expresiones.
- Un panel de ayuda para cada elemento seleccionado en el selector de función.

> **Truco**
>
> Press¡iona `Ctrl+Click` cuando se pasa el nombre de una función en una expresión para mostrar automáticamente su ayuda en el cuadro de diálogo.

El control de valores de un campo que se muestra cuando se selecciona un campo en el selector de funciones ayuda a obtener atributos de objetos:

- Buscar un determinado valor de campo
- Mostrar la lista de valores *Todos únicos* o *10 Muestras*. También dsiponible con click derecho.

Cuando el campo está mapeado con otra capa o un conjunto de valores, es decir, si el *widget de campo* es de tipo *RelationReference*, *ValueRelation* or *ValueMap*, es posible enumerar todos los valores del campo mapeado (de la capa, tabla o lista referenciada). Además, puede filtrar esta lista a *Mostrar solo los valores en uso* en el campo actual.

DOble Click en un valor de campo en el control lo agrega al editor de expresión.

Truco

El panel derecho, mostrando funciones de ayuda o valores de campo, puede ser colapsado (invisible) en el diálogo. Presione el botón guilabel:*Show Values* o *Show Help* para restaurarlo.

Escribir una expresión

Las expresiones de QGIS son usadas para seleccionar objetos o establecer valores. Escribir una expresión en QGIS debe cumplir algunas reglas:

1. **El cuadro de diálogo define el contexto**: si está acostumbrado a SQL, probablemente conozca consultas del tipo *seleccione entidades de la capa donde condición* o *actualice el campo del conjunto de capas = nuevo_valor donde condición*. Una expresión de QGIS también necesita toda esta información, pero la herramienta que usa para abrir el cuadro de diálogo del generador de expresiones proporciona partes de ella. Por ejemplo, dando una capa (`edificios`) con un campo (`altura`):

   - presionando la herramienta Seleccionar por expresión quiere decir que quiere «seleccionar objetos de buildings». La **condición** es la única información que necesita proporcionar en el control de texto de expresión, p.ej. tipo `"height" > 20` para seleccionar buildings mas altos que 20.
   - con esta selección hecha, presionando el botón Field calculator y eligiendo «altura» como *Actualizar campo existente*, ya proporciona el comando «actualizar edificios establecer altura = ??? donde altura> 20». Los únicos bits restantes que debe proporcionar en este caso son el **nuevo valor**, por ejemplo, simplemente ingrese `50` en el cuadro de texto del editor de expresiones para establecer la altura de los edificios seleccionados previamente.

2. **Preste atención a las comillas**: las comillas simples devuelven un literal, por lo que un texto colocado entre comillas simples (`'145'`) se interpreta como una cadena. Las comillas dobles te darán el valor de ese texto, así que úsalas para los campos (`"myfield"`). Los campos también se pueden utilizar sin comillas (`myfield`). Sin comillas para números (`3,16`).

Nota

Las funciones normalmente toman como argumento una cadena para el nombre de campo. Haga:

```
attribute( @atlas_feature, 'height' ) -- returns the value stored in the
↪"height" attribute of the current atlas feature
```

Y no:

```
attribute( @atlas_feature, "height" ) -- fetches the value of the attribute␣
↪named "height" (e.g. 100), and use that value as a field
                                      -- from which to return the atlas␣
↪feature value. Probably wrong as a field named "100" may not exist.
```

Truco

Usar parámetros nombrados para facilitar la lectura de la expresión

Algunas funciones requieren la configuración de muchos parámetros. El motor de expresión admite el uso de parámetros con nombre. Esto significa que en lugar de escribir la expresión críptica `clamp(1, 2, 9)`, puede usar ``clamp(min:=1, value:=2, max:=9) “”. Esto también permite cambiar los argumentos, p. Ej. ``clamp(value:=2, max:=9, min:=1) “”. El uso de parámetros con nombre ayuda a aclarar a qué se refieren los argumentos de una función de expresión, lo cual es útil cuando intentas interpretar una expresión más adelante.

Algunos caso de uso de expresiones

- De la calculadora de Campo, calcular un campo «pop_density» usando los campos existentes «total_pop» y «area_km2»:

```
"total_pop" / "area_km2"
```

- Etiquetar o categorizar entidades basandose en sus áreas:

```
CASE WHEN $area > 10 000 THEN 'Larger' ELSE 'Smaller' END
```

- Actualice el campo «density_level» con categorías de acuerdo con los valores «pop_density»

```
CASE WHEN "pop_density" < 50 THEN 'Low population density'
     WHEN "pop_density" >= 50 and "pop_density" < 150 THEN 'Medium population␣
↪density'
     WHEN "pop_density" >= 150 THEN 'High population density'
END
```

- Aplique un estilo categorizado a todas las características de acuerdo a si el precio promedio de su casa es menor o mayor a 10000€ por metro cuadrado

```
"price_m2" > 10000
```

- Usando la herramienta «Select By Expression…», seleccione todas las características que representan áreas de “High population density” and cuyo precio promedio de la vivienda es superior a 10000€ por metro cuadrado:

```
"density_level" = 'High population density' and "price_m2" > 10000
```

  La expresión previa puede además ser usada para definir que entidades etiquetar o mostrar en el mapa.

- Seleccione los elementos que se solapan con una zona natural en la capa «lands»:

```
overlay_intersects( layer:='lands', filter:="zone_type"='Natural' )
```

- Cuenta para cada objeto espacial el número de edificios que contienen:

```
array_length( overlay_contains( layer:='buildings', expression:=@id ) )
```

- Crear un símbolo diferente (tipo) para la capa, usando el generador de geometría:

```
point_on_surface( @geometry )
```

- Dando una entidad de punto, generar una línea cerrada (usando `make_line`) alrededor de su geometría:

```
make_line(
  -- using an array of points placed around the original
  array_foreach(
    -- list of angles for placing the projected points (every 90°)
    array:=generate_series( 0, 360, 90 ),
    -- translate the point 20 units in the given direction (angle)
    expression:=project( @geometry, distance:=20, azimuth:=radians( @element )␣
↪)
```

(continúe en la próxima página)

(proviene de la página anterior)

```
  )
)
```

- En una etiqueta de diseño de impresión, muestre el nombre de las entidades de los «aeropuertos» que se encuentran dentro del elemento «Mapa 1» del diseño:

```
with_variable( 'extent',
               map_get( item_variables( 'Map 1' ), 'map_extent' ),
               aggregate( 'airports', 'concatenate', "NAME",
                          intersects( @geometry, @extent ), ' ,'
                        )
             )
```

Guardando expresiones

Uso de Add current expression to user expressions sobre el marco del editor de expresiones, puede guardar expresiones importantes a las que desea tener acceso rápido. Están disponibles en el grupo **Expresiones de usuario** en el panel central. Se guardan en el *user profile* (`<userprofile>/QGIS/QGIS3.ini` file) y están disponibles en todos los diálogos de expresión dentro de todos los proyectos del perfil de usuario actual.

Un conjunto de herramientas sobre el marco del editor de expresión le ayuda a administrar las expresiones de usuario:

- Agregar la expresión actual a las expresiones de usuario: almacenar la expresión en el perfil de usuario. Se puede agregar una etiqueta y un texto de ayuda para una fácil identificación.
- Editar expresión seleccionada de expresiones de usuario, así como su ayuda y su etiqueta
- Eliminar la expresión seleccionada de las expresiones del usuario
- Importar expresiones de usuario de un archivo `.json` en la carpeta del perfil del usuario actual
- Exportar expresiones de usuario como un archivo``.json``; todas las expresiones de usuario en el archivo del perfil de usuario `QGIS3.ini` son compartidos

9.1.2 Editor de Funciones

Con la pestaña *Function Editor*, puedes escribir tus propias funciones en lenguaje Python. Esto proporciona una forma práctica y cómoda de abordar necesidades particulares que no estarían cubiertas por las funciones predefinidas.

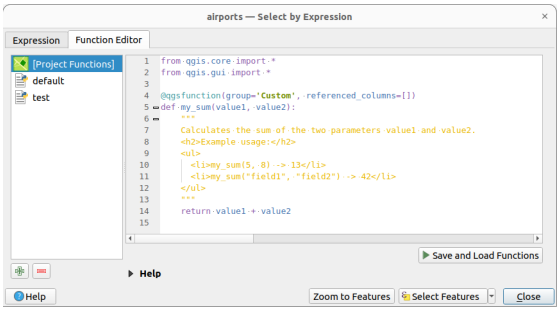


Figura 9.2: La pestaña Editor de funciones

Para crear una nueva función:

1. Presiona el botón New File.

2. Seleccione dónde desea guardar la nueva función. Puede elegir entre guardarla en un *archivo de funciones* o en *funciones del proyecto*.

   - Si selecciona *Archivo de función*, deberá introducir un nombre para el archivo Python `.py`, que se basa en una plantilla de QGIS y se almacena en la carpeta `/python/expressions` bajo el directorio perfil de usuario activo. <user_profiles>`, por lo que estará disponible para todos los proyectos que abras con ese perfil.

     Las funciones Python personalizadas almacenadas como archivos de función se cargan automáticamente cada vez que se inicia un nuevo proyecto en el perfil de usuario. Tenga en cuenta que para compartir dichas funciones Python, es necesario compartir el archivo `.py` en la carpeta `/python/expressions`.

   - Si selecciona *Funciones del proyecto*, la nueva función se almacena en el archivo del proyecto y solo estará disponible para ese proyecto.

     Las funciones Python personalizadas almacenadas como funciones de proyecto se pueden compartir fácilmente compartiendo el archivo de proyecto en el que se guardaron. Al abrir un proyecto con funciones de proyecto, QGIS puede cargarlas o ignorarlas dependiendo de si *el código Python incrustado está habilitado.*.

   Se añade un nuevo elemento en el panel izquierdo de la pestaña *Editor de función*.

3. El panel derecho muestra el contenido del archivo: una plantilla de script de Python.

4. Uncomment the code block and start editing it according to your needs. You can rely on code autocompletion, code formatting and typing helpers, as defined in *Configuración de Python*. Online API documentation for Qt and PyQGIS objects is also accessible pressing the `F1` key.

   Encontrará en el widget *Ayuda* información general para configurar el decorador @qgsfunction.

5. Presione el botón  *Save and Load Functions*. La función que escribió se agrega al árbol de funciones en la pestaña *Expression*, por defecto bajo el grupo``Custom``.

6. Disfruta tu nueva función.

7. Si la función requiere mejoras, habilite la pestaña *Function Editor*, haga los cambios y presione de nuevo el botón  *Save and Load Functions* para hacerlos accesibles en el archivo, por lo tanto en cualquier pestaña de expresión.

Para borrar una función personalizada:

1. Active la pestaña *Editor de Función*
2. Seleccione la función de la lista
3. Pulse el símbolo  Eliminar función seleccionada. La función se elimina de la lista y, dependiendo del almacenamiento, se elimina el archivo `.py` correspondiente de la carpeta de perfil de usuario o se eliminan las funciones Python del archivo de proyecto.

> **Nota**
>
> Cuando se cierra un archivo de proyecto con funciones de proyecto, las funciones de proyecto correspondientes se descargan y dejan de estar disponibles en la sesión de QGIS.

Ejemplo

Aquí hay un breve ejemplo de como crear su propia función `my_sum` que operará con dos valores.

```
from qgis.core import *
from qgis.gui import *

@qgsfunction(group='Custom', referenced_columns=[])
def my_sum(value1, value2):
    """
```

(continúe en la próxima página)

(proviene de la página anterior)

```
    Calculates the sum of the two parameters value1 and value2.
    <h2>Example usage:</h2>
    <ul>
      <li>my_sum(5, 8) -> 13</li>
      <li>my_sum("field1", "field2") -> 42</li>
    </ul>
    """
    return value1 + value2
```

El decorador `@qgsfunction` acepta los siguientes argumentos:

- El argumento `group` indica el grupo en el que debe aparecer la función en el cuadro de diálogo Expresión.
- `usesgeometry`: Establezca este valor en `True` si la expresión requiere acceso a la geometría de las entidades. Por defecto: `False`.
- `handlesnull`: Establezca este valor en `True` si la expresión tiene un tratamiento específico para los valores NULL. Si se establece en `False` (valor predeterminado), el resultado siempre será NULL si alguno de los parámetros es NULL.
- `referenced_columns`: Un array con los nombres de los atributos necesarios para ejecutar la función. Por defecto, el valor es `[QgsFeatureRequest.ALL_ATTRIBUTES]`. Especificar un subconjunto de campos o una lista vacía mejorará el rendimiento.
- `params_as_list`: Si se establece en `True`, los parámetros de la función se pasarán como una lista. Si se establece en `False` (valor predeterminado), la función recibirá los parámetros como argumentos individuales.

La función en sí puede recibir cualquier número y tipo de parámetros que se desee. Estos parámetros deben especificarse antes de los siguientes argumentos opcionales:

- `feature`: el objeto espacial actual
- `parent`: el objeto `QgsExpression`.
- `context`: Si se encuentra un argumento llamado `context` en la última posición, esta variable contendrá un objeto `QgsExpressionContext`, que da acceso a diversa información adicional como variables de expresión. Por ejemplo `context.variable( 'layer_id' )`

Si esos argumentos están presentes en la definición de la función, se pasarán automáticamente a la función, sin necesidad de especificarlos en la expresión.

La función de ejemplo anterior se puede usar en expresiones:

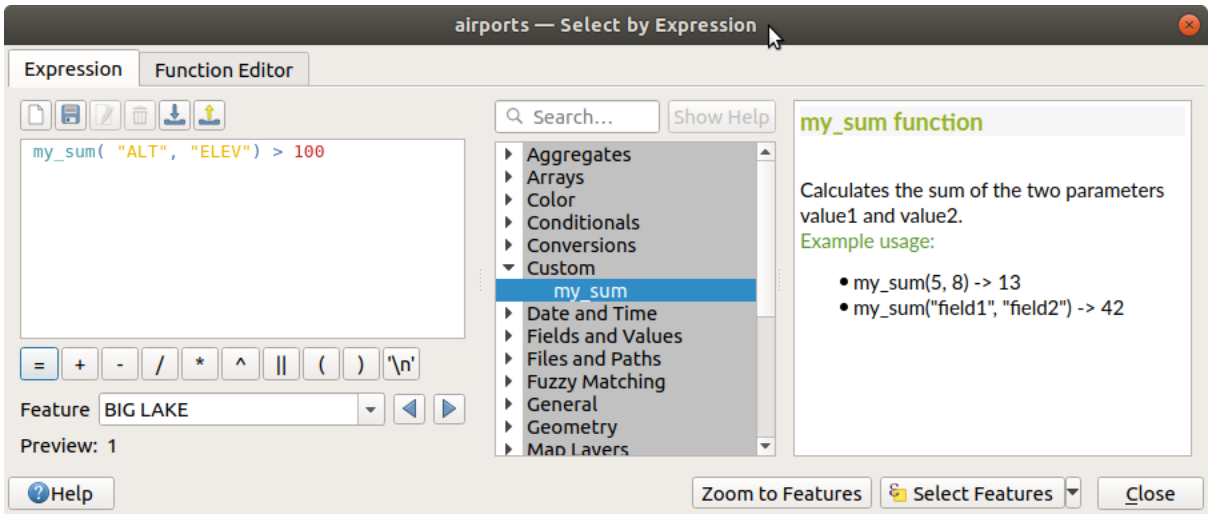


Figura 9.3: Función personalizada agregada a la pestaña Expresión

Puede encontrar más ejemplos y argumentos adicionales para funciones personalizadas en el módulo `qgsfunction`. Para obtener más información sobre la creación de código en Python, consulte la guía PyQGIS-Developer-Cookbook.

9.2 Lista de funciones

Las funciones, operadores y variables disponibles en QGIS están listadas abajo, agrupadas por categorías.

9.2.1 Agregar Funciones

Este grupo contiene funciones que agregan valores sobre campos y capas.

aggregate

Devuelve un valor agregado calculado usando objetos espaciales de otra capa.

Sintaxis	aggregate(layer, aggregate, expression, [filter], [concatenator=""], [order_by]) [] marca argumentos opcionales
Argumentos	• **layer** - una cadena, que representa un nombre capa o ID de capa • **aggregate** - una cadena que corresponde al agregado a calcular. Las opciones válida son: – count – count_distinct – count_missing – mínimo o min – máximo o max – sum – mean – median – stdev – stdevsample – range – minority – majority – q1: primer cuartil – q3: tercer cuartil – iqr: rango intercuartil – min_length: longitud mínima de cadena – max_length: longitud máxima de cadena – concatenate: unir cadenas con un concatenador – concatenate_unique: unir cadenas únicas con un concatenador – collect: crear una geometría multiparte agregada – array_agg: crear una colección de valores agregados • **expresión** - sub expresión o nombre de campo a agregar • **filter** - expresión de filtro opcional para limitar los objetos espaciales usados para calcular el agregado. Los campos y la geometría son de los objetos espaciales en la capa unida. Los objetos fuente pueden ser accedidos con la variable @parent. • **concatenador** - cadena opcional para unir valores en los agregados "concatenate" y "concatenate_unique" • **order_by** - expresión de filtro opcional para ordenar los objetos espaciales usados para calcular el agregado. Los campos y la geometría son de los objetos en la capa unida. De forma predeterminada, los objetos espaciales serán regresados en un orden no especificado.
Ejemplos	• `aggregate(layer:='rail_stations',aggregate:='sum', expression:="passengers")` → suma de todos los valores en el campo pasajeros en la capa rail_stations • `aggregate('rail_stations','sum', "passengers"/7)` → calcula un promedio diario de «passengers» dividiendo el campo «passengers» por 7 antes de sumar agregadamente los valores • `aggregate(layer:='rail_stations',aggregate:='sum', expression:="passengers",filter:="class">3)` → suma agregadamente todos los valores del campo «passengers» de objetos donde el atributo «class» son mayores que 3 solamente • `aggregate(layer:='rail_stations',aggregate:='concatenate', expression:="name", concatenator:=',')` → lista separada por coma del campo nombre para todos los objetos espaciales en la capa rail_stations • `agregate(layer:='countries', aggregate:='max', expression:="code", filter:=intersects( @geometry, geometry(@parent) ) )` → El código de país de un país intersecado en la capa "countries" • `aggregate(layer:='rail_stations',aggregate:='sum', expression:="passengers",filter:=contains( @atlas_geometry, @geometry )` → suma de todos los valores del campo passengers en rail_stations dentro de la característica atlas actual. • `aggregate(layer:='rail_stations', aggregate:='collect', expression:=centroid(@geometry), group_by:="region" = attribute(@parent,'name') )` → agrega geometrías centroides de las rail_stations de la misma región que el objeto espacial actual.

array_agg

Devuelve una matriz de valores agregados de un campo o expresión.

Sintaxis	array_agg(expression, [group_by], [filter], [order_by]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado • **order_by** - expresión opcional que se usa para ordenar las características que se usan para calcular el agregado. De forma predeterminada, las funciones se devolverán en un orden no especificado.
Ejemplos	• `array_agg("name",group_by:="state")` → lista de valores de nombre, agrupados por campo de estado

collect

Devuelve la geometría multiparte de geometrías agregadas desde una expresión

Sintaxis	collect(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - expresión de geometría para agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `collect( @geometry )` → geometría multiparte de geometrías agregadas. • `collect( centroid(@geometry), group_by:="region", filter:= "use" = 'civilian' )` → centroides agregados de los objetos espaciales civiles en función de su valor de región.

concatenate

Devuelve todas las cadenas agregadas de un campo o expresión unidos por un delimitador.

Sintaxis	concatenate(expression, [group_by], [filter], [concatenator], [order_by]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado • **concatenator** - cadena opcional para usar para unir valores. Vacío por defecto. • **order_by** - expresión opcional que se usa para ordenar las características que se usan para calcular el agregado. De forma predeterminada, las funciones se devolverán en un orden no especificado.
Ejemplos	• `concatenate("town_name",group_by:="state",concatenator:=', ')` → lista separada por coma de town_names, agrupado por state field

concatenate_unique

cadena opcional para usar para unir valores. Vacío por defecto.

Sintaxis	concatenate_unique(expression, [group_by], [filter], [concatenator], [order_by]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado • **concatenator** - cadena opcional para usar para unir valores. Vacío por defecto. • **order_by** - expresión opcional que se usa para ordenar las características que se usan para calcular el agregado. De forma predeterminada, las funciones se devolverán en un orden no especificado.
Ejemplos	• `concatenate_unique("town_name",group_by:="state", concatenator:=',')` → lista separada por coma de town_names único, agrupado por state field

count

Devuelve el recuento de objetos coincidentes.

Sintaxis	count(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `count("stations",group_by:="state")` → recuento de estaciones, agrupadas por campo estatal

count_distinct

Devuelve el recuento de valores distintos.

Sintaxis	count_distinct(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `count_distinct("stations",group_by:="state")` → recuento de valores de estaciones distintas, agrupados por campo de estado

count_missing

Devuelve el recuento de valores perdidos (NULL).

Sintaxis	count_missing(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `count_missing("stations",group_by:="state")` → recuento de valores de estación perdidos (NULL), agrupados por campo de estado

iqr

Devuelve el rango intercuartil calculado de un campo o expresión.

Sintaxis	iqr(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `iqr("population",group_by:="state")` → rango intercuartil del valor de la población, agrupado por campo de estado

majority

Devuelve la mayoría agregada de valores (valor que ocurre con más frecuencia) de un campo o expresión.

Sintaxis	majority(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `majority("class",group_by:="state")` → valor de clase más común, agrupado por campo de estado

max_length

Devuelve la longitud máxima de cadenas de un campo o expresión.

Sintaxis	max_length(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `max_length("town_name",group_by:="state")` → longitud máxima de town_name, agrupada por campo de estado

maximum

Devuelve el valor máximo agregado de un campo o expresión.

Sintaxis	maximum(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `maximum("population",group_by:="state")` → valor máximo de población, agrupado por campo de estado

mean

Devuelve el valor medio agregado de un campo o expresión.

Sintaxis	mean(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `mean("population",group_by:="state")` → valor medio de la población, agrupado por campo estatal

median

Devuelve el valor medio agregado de un campo o expresión.

Sintaxis	median(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `median("population",group_by:="state")` → valor medio de la población, agrupado por campo de estado

min_length

Devuelve la longitud mínima de cadenas de un campo o expresión.

Sintaxis	min_length(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `min_length("town_name",group_by:="state")` → longitud mínima de town_name, agrupada por campo de estado

minimum

Devuelve el valor mínimo agregado de un campo o expresión.

Sintaxis	minimum(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `minimum("population",group_by:="state")` → valor mínimo de población, agrupado por campo estatal

minority

Devuelve la minoría agregada de valores (valor que ocurre menos) de un campo o expresión.

Sintaxis	minority(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `minority("class",group_by:="state")` → valor de clase menos frecuente, agrupado por campo de estado

q1

Devuelve el primer cuartil calculado de un campo o expresión.

Sintaxis	q1(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `q1("population",group_by:="state")` → primer cuartil del valor de la población, agrupado por campo de estado

q3

Devuelve el tercer cuartil calculado de un campo o expresión.

Sintaxis	q3(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `q3("population",group_by:="state")` → tercer cuartil del valor de la población, agrupado por campo de estado

range

Devuelve el rango agregado de valores (máximo - mínimo) de un campo o expresión.

Sintaxis	range(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `range("population",group_by:="state")` → rango de valores de población, agrupados por campo de estado

relation_aggregate

Devuelve un valor agregado calculado utilizando todas las entidades secundarias coincidentes de una relación de capa.

Sintaxis	relation_aggregate(relation, aggregate, expression, [concatenator=""], [order_by]) [] marca argumentos opcionales
Argumentos	• **relación**: una cadena que representa un ID de relación • **aggregate** - una cadena que corresponde al agregado a calcular. Las opciones válida son: – count – count_distinct – count_missing – mínimo o min – máximo o max – sum – mean – median – stdev – stdevsample – range – minority – majority – q1: primer cuartil – q3: tercer cuartil – iqr: rango intercuartil – min_length: longitud mínima de cadena – max_length: longitud máxima de cadena – concatenate: unir cadenas con un concatenador – concatenate_unique: unir cadenas únicas con un concatenador – collect: crear una geometría multiparte agregada – array_agg: crear una colección de valores agregados • **expresión** - sub expresión o nombre de campo a agregar • **concatenator** - cadena opcional a usar para unir valores para el agregado "concatenate" • ** order_by **: expresión opcional para ordenar las entidades utilizadas para calcular el agregado. Los campos y la geometría son de las entidades de la capa unida. De forma predeterminada, las funciones se devolverán en un orden no especificado.
Ejemplos	• `relation_aggregate(relation:='my_relation', aggregate:='mean',expression:="passengers")` → valor medio de todas las entidades secundarias coincidentes utilizando la relación "my_relation" • `relation_aggregate('my_relation','sum', "passengers"/7)` → suma del campo de pasajeros dividido por 7 para todas las entidades secundarias coincidentes que utilizan la relación "my_relation" • `relation_aggregate('my_relation','concatenate', "towns", concatenator:=',')` → lista separada por comas del campo de las ciudades para todas las entidades secundarias coincidentes utilizando la relación "my_relation" • `relation_aggregate('my_relation','array_agg', "id")` → matriz del campo de identificación de todas las entidades secundarias coincidentes utilizando la relación "my_relation"

Otras lecturas: *Establecer relaciones entre varias capas*

stdev

Devuelve el valor de desviación estándar agregado de un campo o expresión.

Sintaxis	stdev(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `stdev("population",group_by:="state")` → desviación estándar del valor de la población, agrupada por campo de estado

sum

Devuelve el valor sumado agregado de un campo o expresión.

Sintaxis	sum(expression, [group_by], [filter]) [] marca argumentos opcionales
Argumentos	• **expression** - sub expresión de campo a agregar • **group_by** - expresión opcional para usar para agrupar cálculos agregados • **filter** - expresión opcional para usar para filtrar las entidades que se usan para calcular el agregado
Ejemplos	• `sum("population",group_by:="state")` → Valor de la población sumada, agrupada por campo de estado

9.2.2 Funciones Arreglo

Este grupo contiene funciones para crear y manipular matrices (también conocidas como estructuras de datos de lista). El orden de los valores dentro de la matriz es importante, a diferencia de *"map" data structure*, donde el orden de los pares clave-valor es irrelevante y los valores se identifican por sus claves.

array

Devuelve una matriz que contiene todos los valores pasados como parámetro.

Sintaxis	array(value1, value2, ...)
Argumentos	• **value** - un valor
Ejemplos	• `array(2,10)` → [2, 10] • `array(2,10)[0]` → 2

array_all

Devuelve TRUE si un arreglo contiene todos los valores de un arreglo dado.

Sintaxis	array_all(array_a, array_b)
Argumentos	• **array_a** - una colección • **array_b** - la colección de valores a buscar
Ejemplos	• `array_all(array(1,2,3),array(2,3))` → TRUE • `array_all(array(1,2,3),array(1,2,4))` → FALSE

array_append

Devuelve una matriz con el valor dado agregado al final.

Sintaxis	array_append(array, value)
Argumentos	• **array** - un arreglo • **value** - el valor a agregar
Ejemplos	• `array_append(array(1,2,3),4)` → [1, 2, 3, 4]

array_cat

Devuelve una matriz que contiene todas las matrices dadas concatenadas.

Sintaxis	array_cat(array1, array2, …)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_cat(array(1,2),array(2,3))` → [1, 2, 2, 3]

array_contains

Devuelve TRUE si un array contiene el valor dado.

Sintaxis	array_contains(array, value)
Argumentos	• **array** - un arreglo • **value** - el valor a buscar
Ejemplos	• `array_contains(array(1,2,3),2)` → TRUE

array_count

Cuenta el número de ocurrencias de un valor dado en una matriz.

Sintaxis	array_count(array, value)
Argumentos	• **array** - un arreglo • **valor** - el valor a contar
Ejemplos	• `array_count(array('a', 'b', 'c', 'b'), 'b')` → 2

array_distinct

Devuelve una matriz que contiene valores distintos de la matriz dada.

Sintaxis	array_distinct(array)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_distinct(array(1,2,3,2,1))` → [1, 2, 3]

array_filter

Devuelve una matriz con solo los elementos para los que la expresión se evalúa como verdadera.

Sintaxis	array_filter(array, expression, [limit=0]) [] marca argumentos opcionales
Argumentos	• **array** - un arreglo • **expression**: una expresión para evaluar en cada elemento. La variable @ *elemento* será reemplazada por el valor actual. • **limit** - número máximo de elementos a devolver. Utilice 0 para devolver todos los valores.
Ejemplos	• `array_filter(array(1,2,3),@element < 3)` → [1, 2] • `array_filter(array(1,2,3),@element < 3, 1)` → [1]

array_find

Devuelve el índice más bajo (0 para el primero) de un valor dentro de una matriz. Devuelve -1 si no se encuentra el valor.

Sintaxis	array_find(array, value)
Argumentos	• **array** - un arreglo • **value** - el valor a buscar
Ejemplos	• `array_find(array('a', 'b', 'c'), 'b')` → 1 • `array_find(array('a', 'b', 'c', 'b'), 'b')` → 1

array_first

Devuelve el primer valor de una matriz.

Sintaxis	array_first(array)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_first(array('a','b','c'))` → “a”

array_foreach

Devuelve una matriz con la expresión dada evaluada en cada elemento.

Sintaxis	array_foreach(array, expression)
Argumentos	• **array** - un arreglo • **expresión**: una expresión que se evaluará en cada elemento. La variable *@element* se sustituirá por el valor actual y la variable *@counter* por el índice actual (empezando por 0).
Ejemplos	• `array_foreach(array('a','b','c'),upper(@element))` → [“A”, “B”, “C”] • `array_foreach(array(1,2,3),@element + 10)` → [11, 12, 13] • `array_foreach(array(1,2,3),@element + @counter)` → [1, 3, 5]

array_get

Devuelve el valor N (0 para el primero) o el último valor -N (-1 para el último) de una matriz.

Sintaxis	array_get(array, pos)
Argumentos	• **array** - un arreglo • **pos** - el índice para obtener (basado en 0)
Ejemplos	• `array_get(array('a','b','c'),1)` → “b” • `array_get(array('a','b','c'),-1)` → “c”

Consejo

También puede utilizar el *index operator ([])* para obtener un valor de una matriz.

array_insert

Devuelve una matriz con el valor dado agregado en la posición dada.

Sintaxis	array_insert(array, pos, value)
Argumentos	• **array** - un arreglo • **pos** - la posición donde agregar (basada en 0) • **value** - el valor a agregar
Ejemplos	• `array_insert(array(1,2,3),1,100)` → [1, 100, 2, 3]

array_intersect

Devuelve TRUE si al menos un elemento de array1 existe en array2.

Sintaxis	array_intersect(array1, array2)
Argumentos	• **array1** - un arreglo • **array2** - otro arreglo
Ejemplos	• `array_intersect(array(1,2,3,4),array(4,0,2,5))` → TRUE

array_last

Devuelve el último valor de una matriz.

Sintaxis	array_last(array)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_last(array('a','b','c'))` → “c”

array_length

Devuelve el número de elementos de una matriz.

Sintaxis	array_length(array)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_length(array(1,2,3))` → 3

array_majority

Devuelve los valores más comunes en una matriz.

Sintaxis	array_majority(array, [option="all"]) [] marca argumentos opcionales
Argumentos	• **array** - un arreglo • **option="all"** - una cadena que especifica el manejo de los valores devueltos. Las opciones válidas son: – all: Por defecto, todos los valores más comunes se devuelven en una matriz. – any: Devuelve uno de los valores más comunes. – median: Devuelve la mediana de los valores más comunes. Los valores no aritméticos se ignoran. – real_majority: Devuelve el valor que ocurre más de la mitad del tamaño de la matriz.
Ejemplos	• `array_majority(array(0,1,42,42,43), 'all')` → [42] • `array_majority(array(0,1,42,42,43,1), 'all')` → [42, 1] • `array_majority(array(0,1,42,42,43,1), 'any')` → 1 or 42 • `array_majority(array(0,1,1,2,2), 'median')` → 1.5 • `array_majority(array(0,1,42,42,43), 'real_majority')` → NULL • `array_majority(array(0,1,42,42,43,42), 'real_majority')` → NULL • `array_majority(array(0,1,42,42,43,42,42), 'real_majority')` → 42

array_max

Devuelve el valor máximo de una matriz.

Sintaxis	array_max(array)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_max(array(0,42,4,2))` → 42

array_mean

Devuelve la media de los valores aritméticos de una matriz. Los valores no numéricos de la matriz se ignoran.

Sintaxis	array_mean(array)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_mean(array(0,1,7,66.6,135.4))` → 42 • `array_mean(array(0,84,'a','b','c'))` → 42

array_median

Devuelve la mediana de los valores aritméticos de una matriz. Los valores no aritméticos de la matriz se ignoran.

Sintaxis	array_median(array)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_median(array(0,1,42,42,43))` → 42 • `array_median(array(0,1,2,42,'a','b'))` → 1.5

array_min

Devuelve el valor mínimo de una matriz.

Sintaxis	array_min(array)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_min(array(43,42,54))` → 42

array_minority

Devuelve los valores menos comunes en una matriz.

Sintaxis	array_minority(array, [option=”all”]) [] marca argumentos opcionales
Argumentos	• **array** - un arreglo • **option=”all”** - una cadena que especifica el manejo de los valores devueltos. Las opciones válidas son: – all: Por defecto, todos los valores menos comunes se devuelven en una matriz. – any: Devuelve uno de los valores menos comunes. – median: Devuelve la mediana de los valores menos comunes. Los valores no aritméticos se ignoran. – real_minority: Devuelve valores que ocurren en menos de la mitad del tamaño de la matriz.
Ejemplos	• `array_minority(array(0,42,42), 'all')` → [0] • `array_minority(array(0,1,42,42), 'all')` → [0, 1] • `array_minority(array(0,1,42,42,43,1), 'any')` → 0 or 43 • `array_minority(array(1,2,3,3), 'median')` → 1.5 • `array_minority(array(0,1,42,42,43), 'real_minority')` → [42, 43, 0, 1] • `array_minority(array(0,1,42,42,43,42), 'real_minority')` → [42, 43, 0, 1] • `array_minority(array(0,1,42,42,43,42,42), 'real_minority')` → [43, 0, 1]

array_prepend

Devuelve una matriz con el valor dado añadido al principio.

Sintaxis	array_prepend(array, value)
Argumentos	• **array** - un arreglo • **value** - el valor a agregar
Ejemplos	• `array_prepend(array(1,2,3),0)` → [0, 1, 2, 3]

array_prioritize

Devuelve una matriz ordenada según el orden especificado en otra matriz. Los valores que están presentes en la primera matriz pero que faltan en la segunda matriz se agregarán al final del resultado.

Sintaxis	array_prioritize(array, array_prioritize)
Argumentos	• **array** - un arreglo • **array_prioritize** - una matriz con valores ordenados por prioridad
Ejemplos	• `array_prioritize(array(1, 8, 2, 5), array(5, 4, 2, 1, 3, 8))` → [5, 2, 1, 8] • `array_prioritize(array(5, 4, 2, 1, 3, 8), array(1, 8, 6, 5))` → [1, 8, 5, 4, 2, 3]

array_remove_all

Devuelve una matriz con todas las entradas del valor dado borradas.

Sintaxis	array_remove_all(array, value)
Argumentos	• **array** - un arreglo • **value** - los valores a quitar
Ejemplos	• `array_remove_all(array('a','b','c','b'),'b')` → [“a”, “c”]

array_remove_at

Devuelve una matriz con el elemento en el índice dado eliminado. Admite índice positivo (0 para el primer elemento) y negativo (el último valor -N, -1 para el último elemento).

Sintaxis	array_remove_at(array, pos)
Argumentos	• **array** - un arreglo • **pos** - a posición para quitar (basada en 0)
Ejemplos	• `array_remove_at(array(1, 2, 3), 1)` → [1, 3] • `array_remove_at(array(1, 2, 3), -1)` → [1, 2]

array_replace

Devuelve una matriz con el valor proporcionado, la matriz o el mapa de valores reemplazado.

Valor & variante de matriz

Devuelve una matriz con el valor proporcionado o la matriz de valores reemplazada por otro valor o una matriz de valores.

Sintaxis	array_replace(array, before, after)
Argumentos	• **array** - la colección de entrada • **before** - el valor o matriz de valores a reemplazar • **after** - el valor o matriz de valores para usar como reemplazo
Ejemplos	• `array_replace(array('QGIS','SHOULD','ROCK'),'SHOULD', 'DOES')` → [“QGIS”, “DOES”, “ROCK”] • `array_replace(array(3,2,1),array(1,2,3),array(7,8,9))` → [9, 8, 7] • `array_replace(array('Q','G','I','S'),array('Q','S'),'-')` → [“-”, “G”, “I”, “-”]

Map variant

Devuelve una matriz con las claves de mapa proporcionadas reemplazadas por sus valores emparejados.

Sintaxis	array_replace(array, map)
Argumentos	• **array** - la colección de entrada • **map** - el mapa que contiene claves y valores
Ejemplos	• `array_replace(array('APP', 'SHOULD', 'ROCK'),map('APP', 'QGIS','SHOULD','DOES'))` → [“QGIS”, “DOES”, “ROCK”]

Más información: *replace*, *regexp_replace*

array_reverse

Devuelve la matriz dada con los valores de la matriz en orden inverso.

Sintaxis	array_reverse(array)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_reverse(array(2,4,0,10))` → [10, 0, 4, 2]

array_slice

Devuelve una parte de la matriz. El segmento está definido por los argumentos start_pos y end_pos.

Sintaxis	array_slice(array, start_pos, end_pos)
Argumentos	• **array** - un arreglo • **start_pos** - el índice de la posición inicial del segmento (basado en 0). El índice start_pos se incluye en el segmento. Si usa un start_pos negativo, el índice se cuenta desde el final de la lista (basado en -1). • **end_pos** - el índice de la posición final del segmento (basado en 0). El índice end_pos se incluye en el segmento. Si usa un end_pos negativo, el índice se cuenta desde el final de la lista (basado en -1).
Ejemplos	• `array_slice(array(1,2,3,4,5),0,3)` → [1, 2, 3, 4] • `array_slice(array(1,2,3,4,5),0,-1)` → [1, 2, 3, 4, 5] • `array_slice(array(1,2,3,4,5),-5,-1)` → [1, 2, 3, 4, 5] • `array_slice(array(1,2,3,4,5),0,0)` → [1] • `array_slice(array(1,2,3,4,5),-2,-1)` → [4, 5] • `array_slice(array(1,2,3,4,5),-1,-1)` → [5] • `array_slice(array('Dufour','Valmiera','Chugiak', 'Brighton'),1,2)` → [“Valmiera”, “Chugiak”] • `array_slice(array('Dufour','Valmiera','Chugiak', 'Brighton'),-2,-1)` → [“Chugiak”, “Brighton”]

array_sort

Devuelve la matriz proporcionada con sus elementos ordenados.

Sintaxis	array_sort(array, [ascending=true]) [] marca argumentos opcionales
Argumentos	• **array** - un arreglo • **ascending** - establezca este parámetro en falso para ordenar la matriz en orden descendente
Ejemplos	• `array_sort(array(3,2,1))` → [1, 2, 3]

array_sum

Devuelve la suma de los valores aritméticos de una matriz. Los valores no numéricos de la matriz se ignoran.

Sintaxis	array_sum(array)
Argumentos	• **array** - un arreglo
Ejemplos	• `array_sum(array(0,1,39.4,1.6,'a'))` → 42.0

array_to_string

Concatena los elementos de la matriz en una cadena separada por un delimitador y utiliza una cadena opcional para valores vacíos.

Sintaxis	array_to_string(array, [delimiter=","], [empty_value=""]) [] marca argumentos opcionales
Argumentos	• **array** - la colección de entrada • **delimiter** - el delimitador de cadena utilizado para separar elementos de matriz concatenados • **empty_value** - la cadena opcional para usar como reemplazo de coincidencias vacías (longitud cero)
Ejemplos	• `array_to_string(array('1','2','3'))` → "1,2,3" • `array_to_string(array(1,2,3),'-')` → "1-2-3" • `array_to_string(array('1','','3'),',','0')` → "1,0,3"

Más información: *string_to_array*

generate_series

Crea una matriz que contiene una secuencia de números.

Sintaxis	generate_series(start, stop, [step=1]) [] marca argumentos opcionales
Argumentos	• **start** - primer valor de la secuencia • **stop** - valor que finaliza la secuencia una vez alcanzada • **step** - valor utilizado como incremento entre valores
Ejemplos	• `generate_series(1,5)` → [1, 2, 3, 4, 5] • `generate_series(5,1,-1)` → [5, 4, 3, 2, 1]

geometries_to_array

Divide una geometría en geometrías más simples en una matriz.

Sintaxis	geometries_to_array(geometry)
Argumentos	• **geometry** - la geometría de entrada
Ejemplos	• `geometries_to_array(geom_from_wkt("MultiPoint (1 2, 5 21)"))` → Una matriz que contiene «Point (1 2)» y «Point (5 21)». • `geometries_to_array(geom_from_wkt('GeometryCollection (Polygon ((5 8, 4 1, 3 2, 5 8)),LineString (3 2, 4 2))'))` → una matriz de geometrías poligonales y lineales • `geom_to_wkt(geometries_to_array(geom_from_wkt('GeometryCollection (Polygon ((5 8, 4 1, 3 2, 5 8)),LineString (3 2, 4 2))'))[0])` → "Polygon ((5 8, 4 1, 3 2, 5 8))" • `geometries_to_array(geom_from_wkt('MULTIPOLYGON(((5 5,0 0, 0 10,5 5)),((5 5,10 10,10 0,5 5))'))` → una matriz de dos geometrías poligonales

regexp_matches

Devuelve una matriz de todas las cadenas capturadas mediante la captura de grupos, en el orden en que los propios grupos aparecen en la expresión regular proporcionada contra una cadena.

Sintaxis	regexp_matches(string, regex, [empty_value=""]) [] marca argumentos opcionales
Argumentos	• **string** - la cadena para capturar grupos contra la expresión regular • **regex** - la expresión regular utilizada para capturar grupos • **empty_value** - la cadena opcional para usar como reemplazo de coincidencias vacías (longitud cero)
Ejemplos	• `regexp_matches('QGIS=>rocks','(.*)=>(.*)')` → ["QGIS", "rocks"] • `regexp_matches('key=>','(.*)=>(.*)','empty value')` → ["key", "empty value"]

Más información: *substr*, *regexp_substr*

string_to_array

Divide la cadena en una matriz utilizando el delimitador proporcionado y la cadena opcional para valores vacíos.

Sintaxis	string_to_array(string, [delimiter=","], [empty_value=""]) [] marca argumentos opcionales
Argumentos	• **string** - la cadena entrante • **delimiter** - el delimitador de cadena utilizado para dividir la cadena de entrada • **empty_value** - la cadena opcional para usar como reemplazo de coincidencias vacías (longitud cero)
Ejemplos	• `string_to_array('1,2,3',',')` → ["1", "2", "3"] • `string_to_array('1,,3',',','0')` → ["1", "0", "3"]

Más información: *array_to_string*

9.2.3 Funciones de Color

Este grupo contiene funciones para manipular colores.

color_cmyk

Devuelve una representación en forma de cadena de un color en base a sus componentes cian, magenta, amarillo y negro

Sintaxis	color_cmyk(cyan, magenta, yellow, black)
Argumentos	• **cyan** - componente cian del color, como porcentaje de valor entero de 0 a 100 • **magenta** - componente magenta del color, como un valor entero porcentual de 0 a 100 • **yellow** - componente amarillo del color, como un valor entero porcentual de 0 a 100 • **black** - componente negro del color, como un valor entero porcentual de 0 a 100
Ejemplos	• `color_cmyk(100,50,0,10)` → "0,115,230"

color_cmyka

Devuelve una representación de cadena de un color basada en sus componentes cian, magenta, amarillo, negro y alfa (transparencia)

Sintaxis	color_cmyka(cyan, magenta, yellow, black, alpha)
Argumentos	• **cyan** - componente cian del color, como porcentaje de valor entero de 0 a 100 • **magenta** - componente magenta del color, como un valor entero porcentual de 0 a 100 • **yellow** - componente amarillo del color, como un valor entero porcentual de 0 a 100 • **black** - componente negro del color, como un valor entero porcentual de 0 a 100 • **alpha** - componente alfa como un valor entero de 0 (completamente transparente) a 255 (opaco).
Ejemplos	• `color_cmyka(100,50,0,10,200)` → "0,115,230,200"

color_cmykf

Devuelve un objeto de color basado en sus componentes cian, magenta, amarillo, negro y alfa.

Sintaxis	color_cmykf(cyan, magenta, yellow, black, [alpha=1.0]) [] marca argumentos opcionales
Argumentos	• **cyan**: componente cian como valor flotante entre 0,0 y 1,0. • **magenta**: componente magenta como valor flotante entre 0,0 y 1,0. • **yellow**: componente amarillo como valor flotante entre 0,0 y 1,0. • **black** - componente negro como valor flotante entre 0,0 y 1,0. • **alpha** - componente alfa como valor flotante entre 0,0 y 1,0.
Ejemplos	• `color_cmykf(1,0.9,0.81,0.62)` → CMYKA: 1.00,0.90,0.81,0.62,1.00

color_grayscale_average

Aplica un filtro de escala de grises a un color y lo devuelve. El tipo devuelto es el mismo que el argumento de color, es decir, una representación de cadena de color o un objeto de color.

Sintaxis	color_grayscale_average(color)
Argumentos	• **color**: una representación de cadena de color o un objeto de color.
Ejemplos	• `color_grayscale_average('255,100,50')` → "135,135,135,255" • `color_grayscale_average(color_cmykf(0.6,0.5,0.1,0.8))` → CMYKA: 0.40,0.40,0.40,0.80,1.00

color_hsl

Devuelve una representación de cadena de un color en función de sus atributos de tono, saturación y luminosidad.

Sintaxis	color_hsl(hue, saturation, lightness)
Argumentos	• **hue** - matiz del color, como un valor entero de 0 a 360 • **saturation** - porcentaje de saturación del color como un valor entero de 0 a 100 • **lightness** - porcentaje de luminosidad del color como un valor entero de 0 a 100
Ejemplos	• `color_hsl(100,50,70)` → "166,217,140"

color_hsla

Devuelve una representación en forma de cadena de un color en base a sus atributos de matiz, saturación, luminosidad y canal alfa (transparencia).

Sintaxis	color_hsla(hue, saturation, lightness, alpha)
Argumentos	• **hue** - matiz del color, como un valor entero de 0 a 360 • **saturation** - porcentaje de saturación del color como un valor entero de 0 a 100 • **lightness** - porcentaje de luminosidad del color como un valor entero de 0 a 100 • **alpha** - componente alfa como un valor entero de 0 (completamente transparente) a 255 (opaco).
Ejemplos	• `color_hsla(100,50,70,200)` → "166,217,140,200"

color_hslf

Devuelve un objeto de color basado en sus atributos de tono, saturación y luminosidad.

Sintaxis	color_hslf(hue, saturation, lightness, [alpha=1.0]) [] marca argumentos opcionales
Argumentos	• **hue** - tono del color, como valor flotante entre 0,0 y 1,0 • **saturation** - saturación del color como valor flotante entre 0,0 y 1,0. • **lightness** - luminosidad del color como valor flotante entre 0,0 y 1,0. • **alpha** - componente alfa como valor flotante entre 0,0 y 1,0.
Ejemplos	• `color_hslf(0.3,0.52,0.7)` → HSLA: 0.30,0.52,0.70,1.00

color_hsv

Devuelve una representación de cadena de un color en función de sus atributos de matiz, saturación y valor.

Sintaxis	color_hsv(hue, saturation, value)
Argumentos	• **hue** - matiz del color, como un valor entero de 0 a 360 • **saturation** - porcentaje de saturación del color como un valor entero de 0 a 100 • **value** - porcentaje de valor del color como un número entero de 0 a 100
Ejemplos	• `color_hsv(40,100,100)` → "255,170,0"

color_hsva

Devuelve una representación de cadena de un color en función de sus atributos de tono, saturación, valor y alfa (transparencia).

Sintaxis	color_hsva(hue, saturation, value, alpha)
Argumentos	• **hue** - matiz del color, como un valor entero de 0 a 360 • **saturation** - porcentaje de saturación del color como un valor entero de 0 a 100 • **value** - porcentaje de valor del color como un número entero de 0 a 100 • **alpha** - componente alfa como un valor entero de 0 (completamente transparente) a 255 (opaco)
Ejemplos	• `color_hsva(40,100,100,200)` → "255,170,0,200"

color_hsvf

Devuelve un objeto de color basado en sus atributos de tono, saturación y valor.

Sintaxis	color_hsvf(hue, saturation, value, [alpha=1.0]) [] marca argumentos opcionales
Argumentos	• **hue** - tono del color, como valor flotante entre 0,0 y 1,0 • **saturation** - saturación del color como valor flotante entre 0,0 y 1,0. • **value** - valor del color como un valor flotante entre 0,0 y 1,0 • **alpha** - componente alfa como valor flotante entre 0,0 y 1,0.
Ejemplos	• `color_hsvf(0.4,1,0.6)` → HSVA: 0.40,1.00,0.60,1.00

color_mix

Devuelve un color que mezcla los valores de rojo, verde, azul y alfa de dos colores proporcionados según una proporción determinada. El tipo devuelto es el mismo que el de los argumentos de color, es decir, una representación de color en forma de cadena o un objeto de color.

Sintaxis	color_mix(color1, color2, ratio)
Argumentos	• **color1** - una cadena de color o un objeto de color • **color2** - una cadena de color o un objeto de color • **ratio** - una proporción
Ejemplos	• `color_mix_rgb('0,0,0','255,255,255',0.5)` → "127,127,127,255" • `color_mix(color_cmykf(0.9,0.9,0.9,0.9),color_cmykf(0.1,0.1,0.1,0.1),0.5)` → CMYKA: 0.50,0.50,0.50,0.50,1.00

color_mix_rgb

Devuelve una cadena que representa un color que mezcla los valores rojo, verde, azul y alfa de dos colores proporcionados según una proporción determinada.

Sintaxis	color_mix_rgb(color1, color2, ratio)
Argumentos	• **color1** - una cadena color • **color2** - una cadena color • **ratio** - una proporción
Ejemplos	• `color_mix_rgb('0,0,0','255,255,255',0.5)` → "127,127,127,255"

color_part

Devuelve un componente específico de una cadena de color o un objeto de color, por ejemplo, el componente rojo o el componente alfa.

Sintaxis	color_part(color, component)
Argumentos	• **color**: una cadena de color o un objeto de color. • **component** - una cadena correspondiente al componente de color a devolver. Las opciones válidas son: – red: Componente rojo RGB (0-255) – green: Componente verde RGB (0-255) – blue: Componente azul RGB (0-255) – alpha: valor alfa (transparencia) (0-255) – hue: Tono HSV (0-360) – saturation: Saturación de HSV (0-100) – value: Valor HSV (0-100) – hsl_hue: Tono HSL (0-360) – hsl_saturation: saturación HSL (0-100) – lightness: luminosidad HSL (0-100) – cyan: CMYK componente cyan (0-100) – magenta: CMYK componente magenta (0-100) – yellow: CMYK componente amarillo (0-100) – black: CMYK componente negro (0-100)
Ejemplos	• `color_part('200,10,30','green')` → 10 • `to_int(color_part(color_cmykf(0.1,0.2,0.3,0.9),'black'))` → 90

color_rgb

Devuelve una representación de cadena de un color según sus componentes rojo, verde y azul.

Sintaxis	color_rgb(red, green, blue)
Argumentos	• **red** - componente rojo como un valor entero de 0 a 255 • **green** - componente verde como un valor entero de 0 a 255 • **blue** - componente azul como un valor entero de 0 a 255
Ejemplos	• `color_rgb(255,127,0)` → "255,127,0"

color_rgba

Devuelve una representación de cadena de un color basada en sus componentes rojo, verde, azul y alfa (transparencia).

Sintaxis	color_rgba(red, green, blue, alpha)
Argumentos	• **red** - componente rojo como un valor entero de 0 a 255 • **green** - componente verde como un valor entero de 0 a 255 • **blue** - componente azul como un valor entero de 0 a 255 • **alpha** - componente alfa como un valor entero de 0 (completamente transparente) a 255 (opaco).
Ejemplos	• `color_rgba(255,127,0,200)` → "255,127,0,200"

color_rgbf

Devuelve un objeto de color basado en sus componentes rojo, verde, azul y alfa.

Sintaxis	color_rgbf(red, green, blue, [alpha=1.0]) [] marca argumentos opcionales
Argumentos	• **red** - componente rojo como valor flotante entre 0,0 y 1,0. • **green** - componente verde como valor flotante entre 0,0 y 1,0. • **blue** - componente azul como valor flotante entre 0,0 y 1,0. • **alpha** - componente alfa como valor flotante entre 0,0 y 1,0.
Ejemplos	• `color_rgbf(1.0,0.5,0)` → RGBA: 1.00,0.50,0.00,1.00

create_ramp

Devuelve una rampa de degradado de un mapa de cadenas de colores y pasos.

Sintaxis	create_ramp(map, [discrete=false]) [] marca argumentos opcionales
Argumentos	• **map** - un mapa de cadenas de colores y pasos • **discrete** - establece este parámetro en verdadero para crear una rampa de color discreta
Ejemplos	• `ramp_color(create_ramp(map(0,'0,0,0',1,'255,0,0')),1)` → "255,0,0,255"

darker

Devuelve un color más oscuro (o más claro). El tipo devuelto es el mismo que los argumentos de color, es decir, una representación de cadena de color o un objeto de color.

Sintaxis	darker(color, factor)
Argumentos	• **color**: una cadena de color o un objeto de color. • **factor** - un número entero correspondiente al factor de oscurecimiento: – si el factor es mayor que 100, esta función devuelve un color más oscuro (p. ej., establecer el factor en 200 devuelve un color que es la mitad del brillo); – si el factor es menor que 100, el color de retorno es más claro, pero se recomienda usar la función más claro () para este propósito; – si el factor es 0 o negativo, el valor de retorno no se especifica.
Ejemplos	• `darker('200,10,30', 200)` → "100,5,15,255"

Otras lecturas: *lighter*

lighter

Devuelve un color más claro (o más oscuro). El tipo devuelto es el mismo que los argumentos de color, es decir, una representación de cadena de color o un objeto de color.

Sintaxis	lighter(color, factor)
Argumentos	• **color**: una cadena de color o un objeto de color. • **factor** - un número entero correspondiente al factor de iluminación: – si el factor es mayor que 100, esta función devuelve un color más claro (p. ej., establecer el factor en 150 devuelve un color que es 50% más brillante); – si el factor es menor que 100, el color de retorno es más oscuro, pero se recomienda usar la función darker() para este propósito; – si el factor es 0 o negativo, el valor de retorno no se especifica.
Ejemplos	• `lighter('200,10,30', 200)` → "255,158,168,255"

Otras lecturas: *darker*

project_color

Devuelve un color del esquema de colores del proyecto.

Sintaxis	project_color(name)
Argumentos	• **name** - un nombre de color
Ejemplos	• `project_color('Logo color')` → "20,140,50"

Otras lecturas: *setting project colors*

project_color_object

Devuelve un color del esquema de colores del proyecto. A diferencia de project_color, que devuelve una representación de color en forma de cadena, project_color_object devuelve un objeto de color.

Sintaxis	project_color_object(name)
Argumentos	• **name** - un nombre de color
Ejemplos	• `project_color_object('Logo color')` → RGBA: 0.08,0.55,0.20,1.00

ramp_color

Devuelve una cadena que representa un color de una rampa de color.

Variante de rampa guardada

Devuelve una cadena que representa un color de una rampa guardada

Sintaxis	ramp_color(ramp_name, value)
Argumentos	• **ramp_name** - el nombre de la rampa de color como una cadena, por ejemplo, "Spectral" • **value** - la posición en la rampa para seleccionar el color como un número real entre 0 y 1
Ejemplos	• `ramp_color('Spectral',0.3)` → "253,190,115,255"

Nota

Las rampas de colores disponibles varían entre las instalaciones de QGIS. Es posible que esta función no dé los resultados esperados si mueve su proyecto QGIS entre instalaciones.

Expression-created ramp variant

Devuelve una cadena que representa un color de una rampa creada por expresión

Sintaxis	ramp_color(ramp, value)
Argumentos	• **ramp** - the color ramp • **value** - la posición en la rampa para seleccionar el color como un número real entre 0 y 1
Ejemplos	• `ramp_color(create_ramp(map(0,'0,0,0',1,'255,0,0')),1)` → "255,0,0,255"

Otras lecturas: *Estableciendo una Rampa de Color*, *El atajo de la rampa de color desplegable*

ramp_color_object

Devuelve un objeto de color a partir de una rampa de color. A diferencia de ramp_color, que devuelve una representación de color en forma de cadena, ramp_color_object devuelve un objeto de color.

Variante de rampa guardada

Devuelve un objeto de color a partir de una rampa guardada.

Sintaxis	ramp_color_object(ramp_name, value)
Argumentos	• **ramp_name** - el nombre de la rampa de color como una cadena, por ejemplo, "Spectral" • **value** - la posición en la rampa para seleccionar el color como un número real entre 0 y 1
Ejemplos	• `ramp_color_object('Spectral',0.3)` → RGBA: 0.99,0.75,0.45,1.00

Nota

Las rampas de colores disponibles varían entre las instalaciones de QGIS. Es posible que esta función no dé los resultados esperados si mueve su proyecto QGIS entre instalaciones.

Expression-created ramp variant

Devuelve un objeto de color a partir de una rampa creada mediante una expresión.

Sintaxis	ramp_color_object(ramp, value)
Argumentos	• **ramp** - the color ramp • **value** - la posición en la rampa para seleccionar el color como un número real entre 0 y 1
Ejemplos	• `ramp_color_object(create_ramp(map(0,color_rgbf(0,0,0),1, color_rgbf(1,0,0))),1)` → RGBA: 1.00,0.00,0.00,1.00

set_color_part

Establece un componente de color específico para una cadena de color o un objeto de color, por ejemplo, el componente rojo o el componente alfa.

Sintaxis	set_color_part(color, component, value)
Argumentos	• **color**: una cadena de color o un objeto de color. • **component** - una cadena correspondiente al componente de color a establecer. Las opciones válidas son: – red: Componente rojo RGB (0-255) – green: Componente verde RGB (0-255) – blue: Componente azul RGB (0-255) – alpha: valor alfa (transparencia) (0-255) – hue: Tono HSV (0-360) – saturation: Saturación de HSV (0-100) – value: Valor HSV (0-100) – hsl_hue: Tono HSL (0-360) – hsl_saturation: saturación HSL (0-100) – lightness: luminosidad HSL (0-100) – cyan: CMYK componente cyan (0-100) – magenta: CMYK componente magenta (0-100) – yellow: CMYK componente amarillo (0-100) – black: CMYK componente negro (0-100) • **value** - nuevo valor para el componente de color, respetando los rangos enumerados anteriormente
Ejemplos	• `set_color_part('200,10,30','green',50)` → “200,50,30,255” • `set_color_part(color_cmykf(0.21,0,0.92,0.70),'black',100)` → CMYKA: 0.21,0.00,0.92,1.00,1.00

9.2.4 Funciones Condicionales

Este grupo contiene funciones para manejar comprobaciones condicionales en expresiones.

CASE

CASE se utiliza para evaluar una serie de condiciones y devolver un resultado para la primera condición cumplida. Las condiciones se evalúan secuencialmente y, si una condición es verdadera, la evaluación se detiene y se devuelve el resultado correspondiente. Si ninguna de las condiciones es verdadera, se devuelve el valor de la cláusula ELSE. Además, si no se establece ninguna cláusula ELSE y no se cumple ninguna de las condiciones, se devuelve NULL.

CASE

WHEN *condición* THEN *resultado*

[…n]

[ELSE *resultado*]

END

[ ] marca componentes opcionales

Argumentos	• **WHEN condición** - Una expresión de condición a evaluar • **THEN resultado** - Si *condición* se evalúa como Verdadera entonces *resultado* es evaluado y regresado. • **ELSE resultado** - Si ninguna de la condiciones evaluadas es Verdadera entonces *resultado* es evaluado y regresado.
Ejemplos	• `CASE WHEN "name" IS NULL THEN 'None' END` → Devuelve la cadena "Ninguno" si el campo «nombre» es NULL • `CASE WHEN $area > 10000 THEN 'Big property' WHEN $area > 5000 THEN 'Medium property' ELSE 'Small property' END` → Devuelve la cadena "Propiedad grande" si el área es mayor que 10000, "Propiedad mediana" si el área está entre 5000 y 10000 y "Propiedad pequeña" para otras

coalesce

Devuelve el primer valor no NULO de la lista de expresiones.

Esta función puede aceptar cualquier número de argumentos.

Sintaxis	coalesce(expression1, expression2, …)
Argumentos	• **expression** - cualquier expresión o valor válido, independientemente del tipo.
Ejemplos	• `coalesce(NULL, 2)` → 2 • `coalesce(NULL, 2, 3)` → 2 • `coalesce(7, NULL, 3*2)` → 7 • `coalesce("fieldA", "fallbackField", 'ERROR')` → valor de fieldA si no es NULL; de lo contrario, el valor de «fallbackField» o la cadena "ERROR" si ambos son NULL

if

Prueba una condición y devuelve un resultado diferente según la verificación condicional.

Sintaxis	if(condition, result_when_true, result_when_false)
Argumentos	• **condition** - la condición que debe verificarse • **result_when_true** - el resultado que se devolverá cuando la condición sea verdadera u otro valor que no se convierta en falso. • **result_when_false** - el resultado que se devolverá cuando la condición sea falsa u otro valor que se convierta en falso como 0 o "". NULL también se convertirá en falso.
Ejemplos	• `if( 1+1=2, 'Yes', 'No' )` → "Yes" • `if( 1+1=3, 'Yes', 'No' )` → "No" • `if( 5 > 3, 1, 0)` → 1 • `if( '', 'It is true (not empty)', 'It is false (empty)' )` → "Es falso (vacio)" • `if( ' ', 'It is true (not empty)', 'It is false (empty)' )` → "Si no es verdadero (no vacío)" • `if( 0, 'One', 'Zero' )` → "Zero" • `if( 10, 'One', 'Zero' )` → "One"

nullif

Devuelve un valor NULO si valor1 es igual a valor2; de lo contrario, devuelve value1. Esto se puede usar para sustituir valores condicionalmente por NULL.

Sintaxis	nullif(value1, value2)
Argumentos	• **value1** - El valor que debe usarse o sustituirse por NULL. • **value2** - El valor de control que activará la sustitución de NULL.
Ejemplos	• `nullif('(none)', '(none)')` → NULL • `nullif('text', '(none)')` → "text" • `nullif("name", '')` → NULL, si el nombre es una cadena vacía (o ya es NULL), el nombre en cualquier otro caso.

regexp_match

Devuelve la primera posición coincidente que coincide con una expresión regular dentro de una cadena Unicode, o 0 si no se encuentra la subcadena.

Sintaxis	regexp_match(input_string, regex)
Argumentos	• **input_string** - la cadena para probar con la expresión regular • **regex** - La expresión regular con la que realizar la prueba. Los caracteres de barra invertida deben tener doble escape (e.g., «\\s» para que coincida con un carácter de espacio en blanco o «\\b» para que coincida con el límite de una palabra).
Ejemplos	• `regexp_match('QGIS ROCKS','\\sROCKS')` → 5 • `regexp_match('Budač','udač\\b')` → 2

try

Prueba una expresión y devuelve su valor si no tiene errores. Si la expresión devuelve un error, se devolverá un valor alternativo cuando se proporcione; de lo contrario, la función devolverá NULL.

Sintaxis	try(expression, [alternative]) [] marca argumentos opcionales
Argumentos	• **expression** - la expresión que debería ejecutarse • **alternative** - el resultado que se devolverá si la expresión devuelve un error.
Ejemplos	• `try( to_int( '1' ), 0 )` → 1 • `try( to_int( 'a' ), 0 )` → 0 • `try( to_date( 'invalid_date' ) )` → NULL

9.2.5 Funciones de conversión

Este grupo contiene funciones para convertir un tipo de datos a otro (por ejemplo, cadena de/a entero, binario de/a cadena, cadena a fecha, …).

from_base64

Decodifica una cadena en la codificación Base64 en un valor binario.

Sintaxis	from_base64(string)
Argumentos	• **string** - una cadena a decodificar
Ejemplos	• `from_base64('UUdJUw==')` → “QGIS”

hash

Crea un hash a partir de una cadena con un método determinado. Un byte (8 bits) se representa con dos “”dígitos”” hexadecimales, por lo que “md4” (16 bytes) produce una cadena hexadecimal de 16*2 = 32 caracteres y “keccak_512” (64 bytes) produce un 64 * 2 = Cadena hexadecimal de 128 caracteres.

Sintaxis	hash(string, method)
Argumentos	• **string** - la cadena para hacer hash • **method** - El método hash entre “md4”, “md5”, “sha1”, “sha224”, “sha384”, “sha512”, “sha3_224”, “sha3_256”, “sha3_384”, “sha3_512”, “keccak_224”, “keccak_256”, “keccak_384”, “keccak_512”
Ejemplos	• `hash('QGIS', 'md4')` → “c0fc71c241cdebb6e888cbac0e2b68eb” • `hash('QGIS', 'md5')` → “57470aaa9e22adaefac7f5f342f1c6da” • `hash('QGIS', 'sha1')` → “f87cfb2b74cdd5867db913237024e7001e62b114” • `hash('QGIS', 'sha224')` → “4093a619ada631c770f44bc643ead18fb393b93d6a6af1861fcfece0” • `hash('QGIS', 'sha256')` → “eb045cba7a797aaa06ac58830846e40c8e8c780bc0676d3393605fae50c • `hash('QGIS', 'sha384')` → “91c1de038cc3d09fdd512e99f9dd9922efadc39ed21d3922e69a4305cc2 • `hash('QGIS', 'sha512')` → “c2c092f2ab743bf8edbeb6d028a745f30fc720408465ed369421f0a4e20 • `hash('QGIS', 'sha3_224')` → “467f49a5039e7280d5d42fd433e80d203439e338eaabd701f0d6c17c • `hash('QGIS', 'sha3_256')` → “540f7354b6b8a6e735f2845250f15f4f3ba4f666c55574d9e9354575 • `hash('QGIS', 'sha3_384')` → “96052da1e77679e9a65f60d7ead961b28797782314478638 6eb4364 • `hash('QGIS', 'sha3_512')` → “900d079dc69761da113980253aa8ac0414a8bd6d09879a916228f87 • `hash('QGIS', 'keccak_224')` → “5b0ce6acef8b0a121d4ac4f3eaa8503c799ad4e26a3392d1fb2014 • `hash('QGIS', 'keccak_256')` → “991c520aa6815392de24087f61b2ae0fd56abbfeee4a8ca019c101 • `hash('QGIS', 'keccak_384')` → “c57a3aed9d856fa04e5eeee9b62b6e027cca81ba574116d3cc1f0d

md5

Crea un hash md5 a partir de una cadena.

Sintaxis	md5(string)
Argumentos	• **string** - la cadena para hacer hash
Ejemplos	• `md5('QGIS')` → “57470aaa9e22adaefac7f5f342f1c6da”

sha256

Crea un hash sha256 a partir de una cadena.

Sintaxis	sha256(string)
Argumentos	• **string** - la cadena para hacer hash
Ejemplos	• `sha256('QGIS')` → “eb045cba7a797aaa06ac58830846e40c8e8c780bc0676d3393605fae50c05309”

to_base64

Codifica un valor binario en una cadena, utilizando la codificación Base64.

Sintaxis	to_base64(value)
Argumentos	• **value** - el valor binario a codificar
Ejemplos	• `to_base64('QGIS')` → "UUdJUw=="

to_bool

Converts a given value to a boolean. The function will return false if the value is NULL, an empty string, an empty list, or 0.

Sintaxis	to_bool(value)
Argumentos	• **value** - value to convert to boolean
Ejemplos	• `to_bool('')` → false • `to_bool('123')` → true • `to_bool('false')` → true • `to_bool(0)` → false • `to_bool(1)` → true • `to_bool(null)` → false

to_date

Convierte una cadena en un objeto de fecha. Se puede proporcionar una cadena de formato opcional para analizar la cadena; ver QDate::fromString o la documentación de la función format_date para obtener documentación adicional sobre el formato. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS.

Sintaxis	to_date(string, [format], [language]) [] marca argumentos opcionales
Argumentos	• **string** - cadena representante de un valor de dato • **format** - formato usado para convertir la cadena a fecha • **language** - language (lowercase, two- or three-letter, ISO 639 language code) utilizado para convertir la cadena en una fecha. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS.
Ejemplos	• `to_date('2012-05-04')` → 2012-05-04 • `to_date('June 29, 2019','MMMM d, yyyy')` → 2019-06-29, si la configuración regional actual usa el nombre "junio" para el sexto mes; de lo contrario, se produce un error • `to_date('29 juin, 2019','d MMMM, yyyy','fr')` → 2019-06-29

Más información: *format_date*

to_datetime

Convierte una cadena en un objeto de fecha y hora. Se puede proporcionar una cadena de formato opcional para analizar la cadena; ver QDate::fromString, QTime::fromString o la documentación de la función format_date para obtener documentación adicional sobre el formato. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS.

Sintaxis	to_datetime(string, [format], [language]) [] marca argumentos opcionales
Argumentos	• **string** - cadena que representa un valor de fecha y hora • **format** - formato utilizado para convertir la cadena en una fecha y hora • **language** - language (lowercase, two- or three-letter, ISO 639 language code) utilizado para convertir la cadena en una fecha y hora. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS.
Ejemplos	• `to_datetime('2012-05-04 12:50:00')` → 2012-05-04T12:50:00 • `to_datetime('June 29, 2019 @ 12:34','MMMM d, yyyy @ HH:mm')` → 2019-06-29T12:34, si la configuración regional actual usa el nombre "junio" para el sexto mes; de lo contrario, se produce un error • `to_datetime('29 juin, 2019 @ 12:34','d MMMM, yyyy @ HH:mm', 'fr')` → 2019-06-29T12:34

Más información: *format_date*

to_decimal

Convierte una coordenada en grados, minutos, segundos a su equivalente decimal.

Sintaxis	to_decimal(value)
Argumentos	• **value** - Una cadena de grados, minutos y segundos.
Ejemplos	• `to_decimal('6°21\'16.445')` → 6.3545680555

to_dm

Convierte una coordenada a grados, minutos.

Sintaxis	to_dm(coordinate, axis, precision, [formatting=]) [] marca argumentos opcionales
Argumentos	• **coordinate** - Un valor de latitud o longitud. • **axis** - El eje de la coordenada. Ya sea "x" o "y". • **precision** - Número de decimales. • **formatting** - Designa el tipo de formato. Los valores aceptables son NULL (predeterminado), "alineado" o "sufijo".
Ejemplos	• `to_dm(6.1545681, 'x', 3)` → 6°9.274′ • `to_dm(6.1545681, 'y', 4, 'aligned')` → 6°09.2741′N • `to_dm(6.1545681, 'y', 4, 'suffix')` → 6°9.2741′N

to_dms

Convierte una coordenada a grados, minutos, segundos.

Sintaxis	to_dms(coordinate, axis, precision, [formatting=]) [] marca argumentos opcionales
Argumentos	• **coordinate** - Un valor de latitud o longitud. • **axis** - El eje de la coordenada. Ya sea "x" o "y". • **precision** - Número de decimales. • **formatting** - Designa el tipo de formato. Los valores aceptables son NULL (predeterminado), "alineado" o "sufijo".
Ejemplos	• `to_dms(6.1545681, 'x', 3)` → 6°9′16.445″ • `to_dms(6.1545681, 'y', 4, 'aligned')` → 6°09′16.4452″N • `to_dms(6.1545681, 'y', 4, 'suffix')` → 6°9′16.4452″N

to_int

Convierte una cadena en un número entero. Si un valor no se puede convertir en entero, la expresión no es válida (por ejemplo, "123asd" no es válido).

Sintaxis	to_int(string)
Argumentos	• **string** - cadena a convertir a número entero
Ejemplos	• `to_int('123')` → 123

to_interval

Convierte una cadena en un tipo de intervalo. Puede usarse para tomar días, horas, meses, etc. de una fecha.

Sintaxis	to_interval(string)
Argumentos	• **string** - una cadena que representa un intervalo. Los formatos permitidos incluyen {n} días {n} horas {n} meses.
Ejemplos	• `to_interval('1 day 2 hours')` → interval: 1.08333 days • `to_interval( '0.5 hours' )` → interval: 30 minutes • `to_datetime('2012-05-05 12:00:00') - to_interval('1 day 2 hours')` → 2012-05-04T10:00:00

to_real

Convierte una cadena en un número real. Si un valor no se puede convertir en real, la expresión no es válida (por ejemplo, “123.56asd” no es válido). Los números se redondean después de guardar los cambios si la precisión es menor que el resultado de la conversión.

Sintaxis	to_real(string)
Argumentos	• **string** - cadena para convertir a número real
Ejemplos	• `to_real('123.45')` → 123.45

to_string

Converts a number to string. The conversion is not locale-aware, see “format_number” for a locale-aware alternative.

Sintaxis	to_string(number)
Argumentos	• **number** - Entero o valor real. El número a convertir a cadena.
Ejemplos	• `to_string(1.23)` → “1.23”

Más información: *format_number*

to_time

Convierte una cadena en un objeto de tiempo. Se puede proporcionar una cadena de formato opcional para analizar la cadena; ver QTime::fromString para obtener documentación adicional sobre el formato.

Sintaxis	to_time(string, [format], [language]) [] marca argumentos opcionales
Argumentos	• **string** - cadena representando un valor temporal • **format** - formato utilizado para convertir la cadena en una hora • **language** - idioma (minúsculas, dos o tres letras, código de idioma ISO 639) utilizado para convertir la cadena en una hora
Ejemplos	• `to_time('12:30:01')` → 12:30:01 • `to_time('12:34','HH:mm')` → 12:34:00 • `to_time('12:34','HH:mm','fr')` → 12:34:00

Más información: *format_date*

9.2.6 CRS Functions

This group contains functions to operate on coordinate reference system objects.

crs_from_text

Creates a coordinate reference system from a string definition. The string can represent an authority ID, a WKT definition, or a PROJ string definition of the CRS.

Sintaxis	crs_from_text(definition)
Argumentos	• **definition** - CRS definition
Ejemplos	• `crs_from_text('EPSG:3857')` → crs value "EPSG:3857 - WGS 84 / Pseudo-Mercator" • `crs_from_text('PROJ:+proj=merc +a=6378137 +b=6378137 +lat_ts=0 +lon_0=0 +x_0=0 +y_0=0 +k=1 +units=m +nadgrids=@null +wktext +no_defs +type=crs')` → crs value "EPSG:3857 - WGS 84 / Pseudo-Mercator"

crs_to_authid

Returns the authority ID string for a coordinate reference system.

Sintaxis	crs_to_authid(crs)
Argumentos	• **crs** - crs value
Ejemplos	• `crs_to_authid(crs_from_text('PROJ:+proj=merc +a=6378137 +b=6378137 +lat_ts=0 +lon_0=0 +x_0=0 +y_0=0 +k=1 +units=m +nadgrids=@null +wktext +no_defs +type=crs'))` → "EPSG:3857"

9.2.7 Funciones personalizadas

Este grupo contiene funciones creadas por el usuario. Ver *Editor de Funciones* para mas detalles.

9.2.8 Funciones de Fecha y Hora

Este grupo contiene funciones para manejar datos de fecha y hora. Este grupo comparte varias funciones con las *Funciones de conversión* (to_date, to_time, to_datetime, to_interval) y grupos *Funciones de cadena* (format_date).

Nota

Almacenamiento de fecha, fecha y hora e intervalos en campos

La capacidad de almacenar valores de *fecha*, *hora* y *fecha y hora* directamente en los campos depende del proveedor de la fuente de datos (por ejemplo, Shapefile acepta el formato *fecha*, pero no el formato *fecha y hora* o *hora*). Las siguientes son algunas sugerencias para superar esta limitación:

- *fecha*, *fecha y hora hora* pueden ser convertidos y alamcenados en campos de tipo texto usando la función *format_date()*.
- *Intervalos* pueden ser almacenados en tipo entero o decimaldespues de usar una de las funciones de extracción (p.ej., *day()* para obtener el intervalo expresado en días)

age

Devuelve la diferencia entre dos fechas o fecha-hora.

La diferencia se devuelve como un `Intervalo` y debe usarse con una de las siguientes funciones para extraer información útil:

- `year`
- `month`
- `week`
- `day`
- `hour`
- `minute`
- `second`

Sintaxis	age(datetime1, datetime2)
Argumentos	• **datetime1** - una cadena, fecha o fecha y hora que representa la fecha posterior • **datetime2** - una cadena, fecha o fecha y hora que representa la fecha anterior
Ejemplos	• `day(age('2012-05-12','2012-05-02'))` → 10 • `hour(age('2012-05-12','2012-05-02'))` → 240

datetime_from_epoch

Devuelve una fecha y hora cuya fecha y hora son el número de milisegundos, ms, que han pasado desde 1970-01-01T00:00:00.000, Hora universal coordinada (Qt.UTC) y convertida a Qt.LocalTime.

Sintaxis	datetime_from_epoch(int)
Argumentos	• **int** - número (milisegundos)
Ejemplos	• `datetime_from_epoch(1483225200000)` → 2017-01-01T00:00:00

day

Extrae el día de una fecha o el número de días de un intervalo.

Variante fecha

Extrae el día de una fecha o fecha-hora.

Sintaxis	day(date)
Argumentos	• **date** - un valor de fecha o fecha-hora
Ejemplos	• `day('2012-05-12')` → 12

Variante intervalo

Calcula la longitud en días de un intervalo.

Sintaxis	day(interval)
Argumentos	• **interval** - valor del intervalo a devolver con el número de días
Ejemplos	• `day(to_interval('3 days'))` → 3 • `day(to_interval('3 weeks 2 days'))` → 23 • `day(age('2012-01-01','2010-01-01'))` → 730

day_of_week

Devuelve el día de la semana para una fecha o fecha y hora especificadas. El valor devuelto va de 0 a 6, donde 0 corresponde a un domingo y 6 a un sábado.

Sintaxis	day_of_week(date)
Argumentos	• **date** - valor de fecha o fecha-hora
Ejemplos	• `day_of_week(to_date('2015-09-21'))` → 1

epoch

Devuelve el intervalo en milisegundos entre la época de Unix y un valor de fecha dado.

Sintaxis	epoch(date)
Argumentos	• **date** - un valor de fecha o fecha-hora
Ejemplos	• `epoch(to_date('2017-01-01'))` → 1483203600000

format_date

Formatea un tipo de fecha o una cadena en un formato de cadena personalizado. Utiliza cadenas de formato de fecha/hora Qt. Ver QDateTime::toString.

Sintaxis	format_date(datetime, format, [language]) [] marca argumentos opcionales
Argumentos	• **datetime** - valor de fecha, hora o fecha y hora • **format** - Plantilla de cadena utilizada para formatear la cadena.

Expresión	Salida
d	el día como número sin un cero inicial (1 a 31)
dd	el día como número con un cero inicial (01 a 31)
ddd	el nombre abreviado del día localizado (p. ej., de “Mon” a “Sun”)
dddd	el nombre largo del día localizado (por ejemplo, “Moday” a “Sunday”)
M	El mes como número sin cero inicial (1-12)
MM	el mes como número con un cero por delante (01-12)
MMM	el nombre de mes abreviado localizado (e.g. “Jan” a “Dec”)
MMMM	el nombre de mes largo localizado (e.g. “January” a “December”)
yy	el año como número de dos dígitos (00-99)
yyyy	el año como número de cuatro dígitos

Estas expresiones pueden ser usadas para la parte temporal del formato cadena:

Expresión	Salida
h	la hora sin un cero a la izquierda (0 a 23 o 1 a 12 si se muestra AM / PM)
hh	la hora con un cero a la izquierda (00 a 23 o 01 a 12 si se muestra AM / PM)
H	la hora sin un cero a la izquierda (0 a 23, incluso con pantalla AM / PM)
HH	la hora con un cero a la izquierda (00 a 23, incluso con pantalla AM / PM)
m	el minuto sin un cero a la izquierda (0 a 59)
mm	el minuto con un cero a la izquierda (00 a 59)
s	el segundo sin un cero a la izquierda (0 a 59)
ss	el segundo con un cero a la izquierda (00 a 59)
z	los milisegundos sin ceros finales (0 a 999)
zzz	los milisegundos con ceros finales (000 a 999)
AP o A	interpreta como una hora AM/PM. *AP* deber ser “AM” o “PM”.
ap o a	Interprete como hora AM / PM. *ap* debe ser “am” o “pm”.

- **language** - language (lowercase, two- or three-letter, ISO 639 language code) se utiliza para dar formato a la fecha en una cadena personalizada. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS.

Ejemplos

- `format_date('2012-05-15','dd.MM.yyyy')` → “15.05.2012”
- `format_date('2012-05-15','d MMMM yyyy','fr')` → “15 mai 2012”
- `format_date('2012-05-15','dddd')` → “Tuesday”, si la configuración regional actual es una variante en inglés
- `format_date('2012-05-15 13:54:20','dd.MM.yy')` → “15.05.12”
- `format_date('13:54:20','hh:mm AP')` → “01:54 PM”

hour

Extrae la parte de la hora de una fecha y hora u hora, o el número de horas de un intervalo.

Time variant

Extrae la parte de la hora de una hora o una fecha y hora.

Sintaxis	hour(datetime)
Argumentos	• **datetime**: un valor de hora o fecha y hora
Ejemplos	• `hour( to_datetime('2012-07-22 13:24:57') )` → 13

Variante intervalo

Calcula la duración en horas de un intervalo.

Sintaxis	hour(interval)
Argumentos	• **interval** - valor de intervalor para retornar número de horas desde
Ejemplos	• `hour(to_interval('3 hours'))` → 3 • `hour(age('2012-07-22T13:00:00','2012-07-22T10:00:00'))` → 3 • `hour(age('2012-01-01','2010-01-01'))` → 17520

make_date

Crea un valor fecha a partir de números año, mes y día.

Sintaxis	make_date(year, month, day)
Argumentos	• **year** - Número de año. Los años 1 a 99 se interpretan como están. El año 0 no es válido. • **month** - Número de mes, donde 1 = enero • **day** - Número de día, comenzando con 1 para el primer día del mes
Ejemplos	• `make_date(2020,5,4)` → valor fecha 2020-05-04

make_datetime

Crea un valor de fecha y hora a partir de números de año, mes, día, hora, minuto y segundo.

Sintaxis	make_datetime(year, month, day, hour, minute, second)
Argumentos	• **year** - Número de año. Los años 1 a 99 se interpretan como están. El año 0 no es válido. • **month** - Número de mes, donde 1 = enero • **day** - Número de día, comenzando con 1 para el primer día del mes • **hour** - número horario • **minute** - Minutos • **second** - Segundos (los valores fraccionarios incluyen milisegundos)
Ejemplos	• `make_datetime(2020,5,4,13,45,30.5)` → valor fechahora 2020-05-04 13:45:30.500

make_interval

Crea un valor de intervalo a partir de valores de año, mes, semanas, días, horas, minutos y segundos.

Sintaxis	make_interval([years=0], [months=0], [weeks=0], [days=0], [hours=0], [minutes=0], [seconds=0]) [] marca argumentos opcionales
Argumentos	• **years** - Número de años (supone una duración anual de 365,25 días). • **months** - Número de meses (supone una duración de mes de 30 días) • **weeks** - Número de semanas • **days** - Número de días • **hours** - Número de horas • **minutes** - Número de minutos • **seconds** - Número de segundos
Ejemplos	• `make_interval(hours:=3)` → interval: 3 hours • `make_interval(days:=2, hours:=3)` → interval: 2.125 days • `make_interval(minutes:=0.5, seconds:=5)` → interval: 35 seconds

make_time

Crea un valore temporal a partir de cifras de hora, minuto y segundo.

Sintaxis	make_time(hour, minute, second)
Argumentos	• **hour** - número horario • **minute** - Minutos • **second** - Segundos (los valores fraccionarios incluyen milisegundos)
Ejemplos	• `make_time(13,45,30.5)` → time value 13:45:30.500

minute

Extrae la parte de minutos de una fecha y hora u hora, o el número de minutos de un intervalo.

Time variant

Extrae la parte de los minutos de una hora o una fecha y hora.

Sintaxis	minute(datetime)
Argumentos	• **datetime**: un valor de hora o fecha y hora
Ejemplos	• `minute( to_datetime('2012-07-22 13:24:57') )` → 24

Variante intervalo

Calcula la duración en minutos de un intervalo.

Sintaxis	minute(interval)
Argumentos	• **interval** - valor de intervalo para devolver el número de minutos de
Ejemplos	• `minute(to_interval('3 minutes'))` → 3 • `minute(age('2012-07-22T00:20:00','2012-07-22T00:00:00'))` → 20 • `minute(age('2012-01-01','2010-01-01'))` → 1051200

month

Extrae la parte del mes de una fecha o el número de meses de un intervalo.

Variante fecha

Extrae la parte del mes de una fecha o una fecha y hora.

Sintaxis	month(date)
Argumentos	• **date** - un valor de fecha o fecha-hora
Ejemplos	• `month('2012-05-12')` → 05

Variante intervalo

Calcula la duración en meses de un intervalo.

Sintaxis	month(interval)
Argumentos	• **interval** - valor de intervalo del cual devolver su número de meses
Ejemplos	• `month(to_interval('3 months'))` → 3 • `month(age('2012-01-01','2010-01-01'))` → 4.03333

now

Devuelve la fecha y hora actuales. La función es estática y devolverá resultados consistentes durante la evaluación. La hora devuelta es la hora en que se prepara la expresión.

Sintaxis	now()
Ejemplos	• `now()` → 2012-07-22T13:24:57

second

Extrae la parte de segundos de una fecha y hora u hora, o el número de segundos de un intervalo.

Time variant

Extrae la parte de los segundos de una hora o una fecha y hora.

Sintaxis	second(datetime)
Argumentos	• **datetime**: un valor de hora o fecha y hora
Ejemplos	• `second( to_datetime('2012-07-22 13:24:57') )` → 57

Variante intervalo

Calcula la longitud en segundos de un intervalo.

Sintaxis	second(interval)
Argumentos	• **interval** - valor de intervalo del cual devolver el número de segundos
Ejemplos	• `second(to_interval('3 minutes'))` → 180 • `second(age('2012-07-22T00:20:00','2012-07-22T00:00:00'))` → 1200 • `second(age('2012-01-01','2010-01-01'))` → 63072000

to_date

Convierte una cadena en un objeto de fecha. Se puede proporcionar una cadena de formato opcional para analizar la cadena; ver QDate::fromString o la documentación de la función format_date para obtener documentación adicional sobre el formato. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS.

Sintaxis	to_date(string, [format], [language]) [] marca argumentos opcionales
Argumentos	• **string** - cadena representante de un valor de dato • **format** - formato usado para convertir la cadena a fecha • **language** - language (lowercase, two- or three-letter, ISO 639 language code) utilizado para convertir la cadena en una fecha. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS.
Ejemplos	• `to_date('2012-05-04')` → 2012-05-04 • `to_date('June 29, 2019','MMMM d, yyyy')` → 2019-06-29, si la configuración regional actual usa el nombre "junio" para el sexto mes; de lo contrario, se produce un error • `to_date('29 juin, 2019','d MMMM, yyyy','fr')` → 2019-06-29

Más información: *format_date*

to_datetime

Convierte una cadena en un objeto de fecha y hora. Se puede proporcionar una cadena de formato opcional para analizar la cadena; ver QDate::fromString, QTime::fromString o la documentación de la función format_date para obtener documentación adicional sobre el formato. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS.

Sintaxis	to_datetime(string, [format], [language]) [] marca argumentos opcionales
Argumentos	• **string** - cadena que representa un valor de fecha y hora • **format** - formato utilizado para convertir la cadena en una fecha y hora • **language** - language (lowercase, two- or three-letter, ISO 639 language code) utilizado para convertir la cadena en una fecha y hora. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS.
Ejemplos	• `to_datetime('2012-05-04 12:50:00')` → 2012-05-04T12:50:00 • `to_datetime('June 29, 2019 @ 12:34','MMMM d, yyyy @ HH:mm')` → 2019-06-29T12:34, si la configuración regional actual usa el nombre "junio" para el sexto mes; de lo contrario, se produce un error • `to_datetime('29 juin, 2019 @ 12:34','d MMMM, yyyy @ HH:mm', 'fr')` → 2019-06-29T12:34

Más información: *format_date*

to_interval

Convierte una cadena en un tipo de intervalo. Puede usarse para tomar días, horas, meses, etc. de una fecha.

Sintaxis	to_interval(string)
Argumentos	• **string** - una cadena que representa un intervalo. Los formatos permitidos incluyen {n} días {n} horas {n} meses.
Ejemplos	• `to_interval('1 day 2 hours')` → interval: 1.08333 days • `to_interval( '0.5 hours' )` → interval: 30 minutes • `to_datetime('2012-05-05 12:00:00') - to_interval('1 day 2 hours')` → 2012-05-04T10:00:00

to_time

Convierte una cadena en un objeto de tiempo. Se puede proporcionar una cadena de formato opcional para analizar la cadena; ver QTime::fromString para obtener documentación adicional sobre el formato.

Sintaxis	to_time(string, [format], [language]) [] marca argumentos opcionales
Argumentos	• **string** - cadena representando un valor temporal • **format** - formato utilizado para convertir la cadena en una hora • **language** - idioma (minúsculas, dos o tres letras, código de idioma ISO 639) utilizado para convertir la cadena en una hora
Ejemplos	• `to_time('12:30:01')` → 12:30:01 • `to_time('12:34','HH:mm')` → 12:34:00 • `to_time('12:34','HH:mm','fr')` → 12:34:00

Más información: *format_date*

week

Extrae el número de semana de una fecha o el número de semanas de un intervalo.

Variante fecha

Extrae el número de semana de una fecha o fecha y hora..

Sintaxis	week(date)
Argumentos	• **date** - un valor de fecha o fecha-hora
Ejemplos	• `week('2012-05-12')` → 19

Variante intervalo

Calcula la longitud en semanas de un inervalo.

Sintaxis	week(interval)
Argumentos	• **interval** - valor de intervalo del cual devolver su número de meses
Ejemplos	• `week(to_interval('3 weeks'))` → 3 • `week(age('2012-01-01','2010-01-01'))` → 104.285

year

Extrae la parte del año de una fecha o el número de años de un intervalo.

Variante fecha

Extrae la parte del año de una fecha o una fecha y hora.

Sintaxis	year(date)
Argumentos	• **date** - un valor de fecha o fecha-hora
Ejemplos	• `year('2012-05-12')` → 2012

Variante intervalo

Calcula la duración en años de un intervalo.

Sintaxis	year(interval)
Argumentos	• **interval** - valor de intervalo del cual devolver el número de años
Ejemplos	• `year(to_interval('3 years'))` → 3 • `year(age('2012-01-01','2010-01-01'))` → 1.9986

Algún ejemplo:

Además de estas funciones, restar fechas, fechas y horas utilizando el operador `` -`` (menos) devolverá un intervalo.

Sumar o restar un intervalo a fechas, fechas y horas, utilizando los operadores `` + “” (más) y `` -`` (menos), devolverá una fecha y hora.

- Obtener el número de días hasta el lanzamiento de QGIS 3.0

```
to_date('2017-09-29') - to_date(now())
-- Returns <interval: 203 days>
```

- Lo mismo con la hora:

```
to_datetime('2017-09-29 12:00:00') - now()
-- Returns <interval: 202.49 days>
```

- Obtener la fecha con hora de ahora a 100 días:

```
now() + to_interval('100 days')
-- Returns <datetime: 2017-06-18 01:00:00>
```

9.2.9 Campos y Valores

Contiene una lista de campos de la capa activa y valores especiales. La lista de campos incluye los almacenados en el conjunto de datos, *virtual* y *auxiliar* así como los de *uniones*.

Doble-click en el nombre de un campo para añadirlo a tu expresión. También puede escribir el nombre del campo (preferiblemente entre comillas dobles) o su *alias*.

Para recuperar los valores de los campos para usar en una expresión, seleccione el campo apropiado y, en el widget que se muestra, elija entre *10 Samples* y *All Unique*. Los valores solicitados se muestran y puede usar el cuadro *Search* en la parte superior de la lista para filtrar el resultado. También se puede acceder a los valores de muestra haciendo clic derecho en un campo.

Para añadir un valor a una expresión que esté escribiendo, doble-click en él en la lista. Si el valor es del tipo cadena, debe ir con comillas simples, en otro caso no se necesitan comillas.

NULL

Equivale a un valor NULL.

Sintaxis	NULL
Ejemplos	• `NULL` → un valor NULO

Nota

Para probar NULL use una expresión *IS NULL* o *IS NOT NULL*.

9.2.10 Funciones de Archivos y Rutas

Este grupo contiene funciones que manipulan nombres de archivos y rutas.

base_file_name

Devuelve el nombre base de un archivo sin el directorio o el sufijo del archivo.

Sintaxis	base_file_name(path)
Argumentos	• **path** - una ruta de archivo o un valor de capa de mapa. Si se especifica un valor de capa de mapa, se utilizará la fuente de archivo de la capa.
Ejemplos	• `base_file_name('/home/qgis/data/country_boundaries.shp')` → “country_boundaries”

exif

Recupera valores de etiquetas exif de un archivo de imagen.

Sintaxis	exif(path, [tag]) [] marca argumentos opcionales
Argumentos	• **ruta** - Una ruta de archivo de imagen o un valor de capa de mapa. Si se especifica un valor de capa de mapa, se utilizará el origen de archivo de la capa. • **tag** - La etiqueta a devolver. Si está vacío, se devolverá un mapa con todos los valores de etiquetas exif.
Ejemplos	• `exif('/my/photo.jpg','Exif.Image.Orientation')` → 0

file_exists

Devuelve TRUE si existe una ruta de archivo.

Sintaxis	file_exists(path)
Argumentos	• **path** - una ruta de archivo o un valor de capa de mapa. Si se especifica un valor de capa de mapa, se utilizará la fuente de archivo de la capa.
Ejemplos	• `file_exists('/home/qgis/data/country_boundaries.shp')` → TRUE

file_name

Devuelve el nombre de un archivo (incluyendo la extensión del archivo), excluyendo el directorio.

Sintaxis	file_name(path)
Argumentos	• **path** - una ruta de archivo o un valor de capa de mapa. Si se especifica un valor de capa de mapa, se utilizará la fuente de archivo de la capa.
Ejemplos	• `file_name('/home/qgis/data/country_boundaries.shp')` → "country_boundaries.shp"

file_path

Devuelve el componente de directorio de una ruta de archivo. Esto no incluye el nombre del archivo.

Sintaxis	file_path(path)
Argumentos	• **path** - una ruta de archivo o un valor de capa de mapa. Si se especifica un valor de capa de mapa, se utilizará la fuente de archivo de la capa.
Ejemplos	• `file_path('/home/qgis/data/country_boundaries.shp')` → "/home/qgis/data"

file_size

Devuelve el tamaño (en bytes) de un archivo.

Sintaxis	file_size(path)
Argumentos	• **path** - una ruta de archivo o un valor de capa de mapa. Si se especifica un valor de capa de mapa, se utilizará la fuente de archivo de la capa.
Ejemplos	• `file_size('/home/qgis/data/country_boundaries.geojson')` → 5674

file_suffix

Devuelve el sufijo (extensión) del archivo de una ruta de archivo.

Sintaxis	file_suffix(path)
Argumentos	• **path** - una ruta de archivo o un valor de capa de mapa. Si se especifica un valor de capa de mapa, se utilizará la fuente de archivo de la capa.
Ejemplos	• `file_suffix('/home/qgis/data/country_boundaries.shp')` → "shp"

is_directory

Devuelve TRUE si una ruta corresponde a un directorio.

Sintaxis	is_directory(path)
Argumentos	• **path** - una ruta de archivo o un valor de capa de mapa. Si se especifica un valor de capa de mapa, se utilizará la fuente de archivo de la capa.
Ejemplos	• `is_directory('/home/qgis/data/country_boundaries.shp')` → FALSE • `is_directory('/home/qgis/data/')` → TRUE

is_file

Devuelve TRUE si una ruta corresponde a un archivo.

Sintaxis	is_file(path)
Argumentos	• **path** - una ruta de archivo o un valor de capa de mapa. Si se especifica un valor de capa de mapa, se utilizará la fuente de archivo de la capa.
Ejemplos	• `is_file('/home/qgis/data/country_boundaries.shp')` → TRUE • `is_file('/home/qgis/data/')` → FALSE

9.2.11 Funciones de formulario

Este grupo contiene funciones que operan exclusivamente en el contexto de formulario de atributos. Por ejemplo, en la configuración de *controles de campo*.

current_parent_value

Solo se puede usar en un contexto de formulario incrustado, esta función devuelve el valor actual no guardado de un campo en el formulario principal que se está editando actualmente. Esto diferirá de los valores de atributo reales de la entidad principal para las entidades que se están editando actualmente o que aún no se han agregado a una capa principal. Cuando se usa en una expresión de filtro de widget de relación de valor, esta función debe estar envuelta en un "coalesce()" que puede recuperar la característica principal real de la capa cuando el formulario no se usa en un contexto incrustado.

Sintaxis	current_parent_value(field_name)
Argumentos	• **field_name** - un nombre de campo en el formulario principal actual
Ejemplos	• `current_parent_value( 'FIELD_NAME' )` → El valor actual de un campo "FIELD_NAME" en el formulario principal.

current_value

Devuelve el valor actual, no guardado, de un campo en el formulario o fila de la tabla que se está editando actualmente. Esto diferirá de los valores de atributo reales de la entidad para las entidades que se están editando actualmente o que aún no se han agregado a una capa.

Sintaxis	current_value(field_name)
Argumentos	• **field_name** - un nombre de campo en el formulario actual o fila de la tabla
Ejemplos	• `current_value( 'FIELD_NAME' )` → El valor actual del campo "FIELD_NAME".

9.2.12 Funciones Concordancia aproximada

Este grupo contiene funciones para comparaciones difusas entre valores.

hamming_distance

Devuelve la distancia Hamming entre dos cadenas. Esto equivale al número de caracteres en las posiciones correspondientes dentro de las cadenas de entrada donde los caracteres son diferentes. Las cadenas de entrada deben tener la misma longitud y la comparación distingue entre mayúsculas y minúsculas.

Sintaxis	hamming_distance(string1, string2)
Argumentos	• **string1** - una cadena • **string2** - una cadena
Ejemplos	• `hamming_distance('abc','xec')` → 2 • `hamming_distance('abc','ABc')` → 2 • `hamming_distance(upper('abc'),upper('ABC'))` → 0 • `hamming_distance('abc','abcd')` → NULL

levenshtein

Devuelve la distancia de edición de Levenshtein entre dos cadenas. Esto equivale al número mínimo de ediciones de caracteres (inserciones, eliminaciones o sustituciones) necesarias para cambiar una cadena a otra.

La distancia de Levenshtein es una medida de la similitud entre dos cadenas. Las distancias más pequeñas significan que las cadenas son más similares y las distancias más grandes indican cadenas más diferentes. La distancia distingue entre mayúsculas y minúsculas.

Sintaxis	levenshtein(string1, string2)
Argumentos	• **string1** - una cadena • **string2** - una cadena
Ejemplos	• `levenshtein('kittens','mitten')` → 2 • `levenshtein('Kitten','kitten')` → 1 • `levenshtein(upper('Kitten'),upper('kitten'))` → 0

longest_common_substring

Devuelve la subcadena común más larga entre dos cadenas. Esta subcadena es la cadena más larga que es una subcadena de las dos cadenas de entrada. Por ejemplo, la subcadena común más larga de «ABABC» y «BABCA» es «BABC». La subcadena distingue entre mayúsculas y minúsculas.

Sintaxis	longest_common_substring(string1, string2)
Argumentos	• **string1** - una cadena • **string2** - una cadena
Ejemplos	• `longest_common_substring('ABABC','BABCA')` → "BABC" • `longest_common_substring('abcDeF','abcdef')` → "abc" • `longest_common_substring(upper('abcDeF'),upper('abcdex'))` → "ABCDE"

soundex

Devuelve la representación Soundex de una cadena. Soundex es un algoritmo de coincidencia fonética, por lo que las cadenas con sonidos similares deben estar representadas por el mismo código Soundex.

Sintaxis	soundex(string)
Argumentos	• **string** - una cadena
Ejemplos	• `soundex('robert')` → "R163" • `soundex('rupert')` → "R163" • `soundex('rubin')` → "R150"

9.2.13 Funciones Generales

Este grupo contiene un surtido de funciones generales.

env

Obtiene una variable de entorno y devuelve su contenido como una cadena. Si no se encuentra la variable, se devolverá NULL. Esto es útil para inyectar una configuración específica del sistema, como letras de unidad o prefijos de ruta. La definición de las variables de entorno depende del sistema operativo; consulte con el administrador del sistema o con la documentación del sistema operativo cómo se puede configurar.

Sintaxis	env(name)
Argumentos	• **name** - El nombre de la variable de entorno que se debe recuperar.
Ejemplos	• `env( 'LANG' )` → "en_US.UTF-8" • `env( 'MY_OWN_PREFIX_VAR' )` → "Z:" • `env( 'I_DO_NOT_EXIST' )` → NULL

eval

Evalúa una expresión que es pasada como una cadena. Útil para expandir parámetros dinámicos pasados como variables de contexto o campos.

Sintaxis	eval(expression)
Argumentos	• **expression** - una cadena de expresión
Ejemplos	• `eval('\'nice\'')` → "nice" • `eval(@expression_var)` → [cualquiera que sea el resultado de evaluar @expression_var debe ser…]

eval_template

Evalúa una plantilla que se pasa en una cadena. Útil para expandir parámetros dinámicos pasados como variables de contexto o campos.

Sintaxis	eval_template(template)
Argumentos	• **template** - una cadena plantilla
Ejemplos	• `eval_template('QGIS [% upper(\'rocks\') %]')` → QGIS ROCKS

is_layer_visible

Devuelve TRUE si la capa especificada es visible.

Sintaxis	is_layer_visible(layer)
Argumentos	• **layer** - una cadena, que representa un nombre capa o ID de capa
Ejemplos	• `is_layer_visible('baseraster')` → TRUE

mime_type

Devuelve el tipo mime de los datos binarios.

Sintaxis	mime_type(bytes)
Argumentos	• **bytes** - los datos binarios
Ejemplos	• `mime_type('<html><body></body></html>')` → text/html • `mime_type(from_base64('R0lGODlhAQABAAAAACH5BAEKAAEALAAAAAABAAEAAAIAOw==` → image/gif

var

Devuelve el valor almacenado dentro de una variable especificada.

Sintaxis	var(name)
Argumentos	• **name** - un nombre de variable
Ejemplos	• `var('qgis_version')` → "2.12"

Otras lecturas: Lista de *variables* predeterminadas

with_variable

Esta función establece una variable para cualquier código de expresión que se proporcionará como 3er argumento. Esto solo es útil para expresiones complicadas, donde el mismo valor calculado debe usarse en diferentes lugares.

Sintaxis	with_variable(name, value, expression)
Argumentos	• **name** - el nombre de la variable a establecer • **value** - el valor a establecer • **expression** - la expresión para la cual la variable estará disponible
Ejemplos	• `with_variable('my_sum', 1 + 2 + 3, @my_sum * 2 + @my_sum * 5)` → 42

9.2.14 Funciones de Geometría

Este grupo contiene funciones que operan en objetos geométricos. (p.ej. buffer, transform, $area).

affine_transform

Devuelve la geometría después de una transformación afín. Los cálculos están en el Sistema de Referencia Espacial de esta geometría. Las operaciones se realizan en un orden de escala, rotación y traslación. Si hay un desplazamiento Z o M pero la coordenada no está presente en la geometría, se agregará.

Sintaxis	affine_transform(geometry, delta_x, delta_y, rotation_z, scale_x, scale_y, [delta_z=0], [delta_m=0], [scale_z=1], [scale_m=1]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **delta_x** - traslación del eje x • **delta_y** - traslación del eje y • **rotation_z** - rotación alrededor del eje z en grados en sentido antihorario • **scale_x** - factor de escala del eje x • **scale_y** - factor de escala del eje y • **delta_z** - traslación del eje z • **delta_m** - traslación del eje m • **scale_z** - factor de escala del eje z • **scale_m** - factor de escala del eje m
Ejemplos	• `geom_to_wkt(affine_transform(geom_from_wkt('LINESTRING(1 1, 2 2)'), 2, 2, 0, 1, 1))` → "LineString (3 3, 4 4)" • `geom_to_wkt(affine_transform(geom_from_wkt('POLYGON((0 0, 0 3, 2 2, 0 0))'), 0, 0, -90, 1, 2))` → "Polygon ((0 0, 6 0, 4 -2, 0 0))" • `geom_to_wkt(affine_transform(geom_from_wkt('POINT(3 1)'), 0, 0, 0, 1, 1, 5, 0))` → "PointZ (3 1 5)"

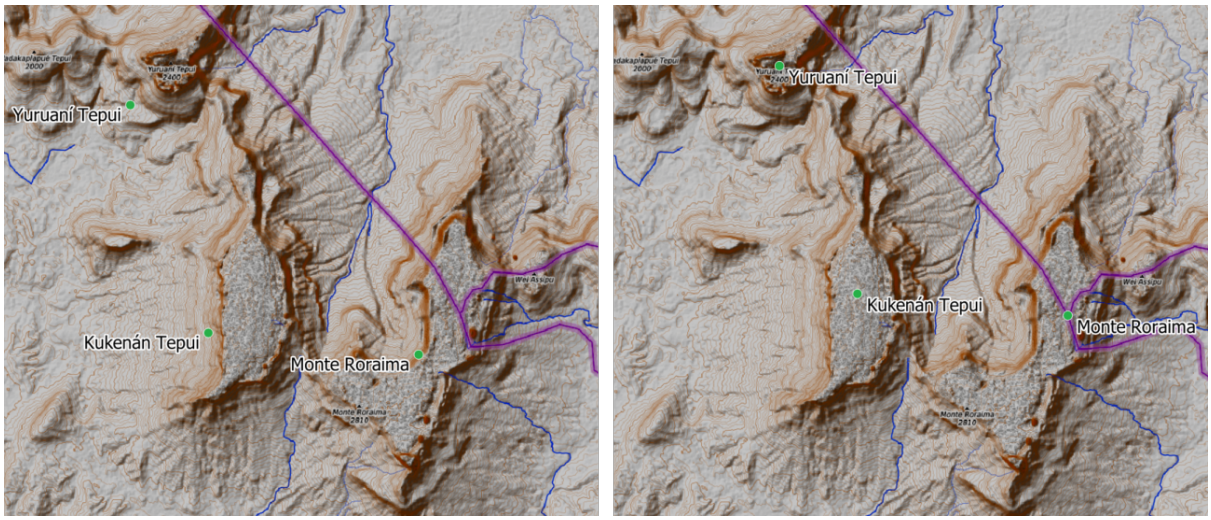


Figura 9.4: Capa de puntos vectoriales (puntos verdes) antes (izquierda) y después (derecha) de una transformación afín (traslación).

angle_at_vertex

Devuelve el ángulo de la bisectriz(ángulo medio) de la geometría para un vértice especificado en una geometría de cadena lineal. Los ángulos están en grados en el sentido de las agujas del reloj desde el norte.

Sintaxis	angle_at_vertex(geometry, vertex)
Argumentos	• **geometry** - una geometría de cadena lineal • **vertex** - índice de vértice, comenzando desde 0; si el valor es negativo, el índice de vértice seleccionado será su recuento total menos el valor absoluto
Ejemplos	• `angle_at_vertex(geometry:=geom_from_wkt('LineString(0 0, 10 0, 10 10)'),vertex:=1)` → 45.0

apply_dash_pattern

Aplica un patrón de guiones a una geometría, devolviendo una geometría MultiLineString que es la geometría de entrada trazada a lo largo de cada línea/anillo con el patrón especificado.

Sintaxis	apply_dash_pattern(geometry, pattern, [start_rule=no_rule], [end_rule=no_rule], [adjustment=both], [pattern_offset=0]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría (acepta (multi)líneas o (multi)polígonos). • **pattern** - patrón de guiones, como una matriz de números que representan la longitud de los guiones y los espacios. Debe contener un número par de elementos. • **start_rule** - regla opcional para limitar el inicio del patrón. Los valores válidos son “no_rule”, “full_dash”, “half_dash”, “full_gap”, “half_gap”. • **end_rule** - regla opcional para limitar el final del patrón. Los valores válidos son “no_rule”, “full_dash”, “half_dash”, “full_gap”, “half_gap”. • **adjustment** - regla opcional para especificar qué parte de los patrones se ajustan a las reglas de patrón deseadas. Los valores válidos son “both”, “dash”, “gap”. • **pattern_offset** - Distancia opcional que especifica una distancia específica a lo largo del patrón en la que comenzar.
Ejemplos	• `geom_to_wkt(apply_dash_pattern(geom_from_wkt('LINESTRING(1 1, 10 1)'), array(3, 1)))` → MultiLineString ((1 1, 4 1),(5 1, 8 1),(9 1, 10 1, 10 1)) • `geom_to_wkt(apply_dash_pattern(geom_from_wkt('LINESTRING(1 1, 10 1)'), array(3, 1), start_rule:='half_dash'))` → MultiLineString ((1 1, 2.5 1),(3.5 1, 6.5 1),(7.5 1, 10 1, 10 1))

$area

Devuelve el área de la entidad actual. El área calculada por esta función respeta tanto la configuración del elipsoide del proyecto actual como la configuración de la unidad de área. Por ejemplo, si se ha configurado un elipsoide para el proyecto, el área calculada será elipsoidal, y si no se establece ningún elipsoide, el área calculada será planimétrica.

Sintaxis	$area
Ejemplos	• `$area` → 42

area

Devuelve el área de un objeto poligonal de geometría. Los cálculos son siempre planimétricos en el Sistema de referencia espacial (SRS) de esta geometría, y las unidades del área devuelta coincidirán con las unidades del SRS. Esto difiere de los cálculos realizados por la función $ area, que realizará cálculos elipsoidales basados en la configuración de unidad de área y elipsoide del proyecto.

Sintaxis	area(geometry)
Argumentos	• **geometry** - objeto poligonal de geometría
Ejemplos	• `area(geom_from_wkt('POLYGON((0 0, 4 0, 4 2, 0 2, 0 0))'))` → 8.0

azimuth

Devuelve el acimut basado en el norte como el ángulo en radianes medido en el sentido de las agujas del reloj desde la vertical en point_a hasta point_b.

Sintaxis	azimuth(point_a, point_b)
Argumentos	• **point_a** - geometría de punto • **point_b** - geometría de punto
Ejemplos	• `degrees( azimuth( make_point(25, 45), make_point(75, 100) ) )` → 42.273689 • `degrees( azimuth( make_point(75, 100), make_point(25,45) ) )` → 222.273689

bearing

Devuelve el rumbo basado en el norte como el ángulo en radianes medido en el sentido de las agujas del reloj en el elipsoide desde la vertical en el punto_a hasta el punto_b.

Sintaxis	bearing(point_a, point_b, [source_crs], [ellipsoid]) [] marca argumentos opcionales
Argumentos	• **point_a** - geometría de punto • **point_b** - geometría de punto • **source_crs** - an optional string or CRS object representing the source CRS of the points. By default the current layer's CRS is used. • **ellipsoid** - an optional string representing the acronym or the authority ID (e.g., "EPSG:7030") of the ellipsoid on which the bearing should be measured. By default the current project's ellipsoid setting is used.
Ejemplos	• `degrees( bearing( make_point(16198544, -4534850), make_point(18736872, -1877769), 'EPSG:3857', 'EPSG:7030') )` → 49.980071 • `degrees( bearing( make_point(18736872, -1877769), make_point(16198544, -4534850), 'EPSG:3857', 'WGS84') )` → 219.282386

boundary

Devuelve el cierre del límite combinatorio de la geometría (es decir, el límite topológico de la geometría). Por ejemplo, una geometría de polígono tendrá un límite que consta de las cadenas de líneas para cada anillo en el polígono. Algunos tipos de geometría no tienen un límite definido, por ejemplo, puntos o colecciones de geometría, y devolverán NULL.

Sintaxis	boundary(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `geom_to_wkt(boundary(geom_from_wkt('Polygon((1 1, 0 0, -1 1, 1 1))')))` → "LineString(1 1,0 0,-1 1,1 1)" • `geom_to_wkt(boundary(geom_from_wkt('LineString(1 1,0 0,-1 1)')))` → "MultiPoint ((1 1),(-1 1))"

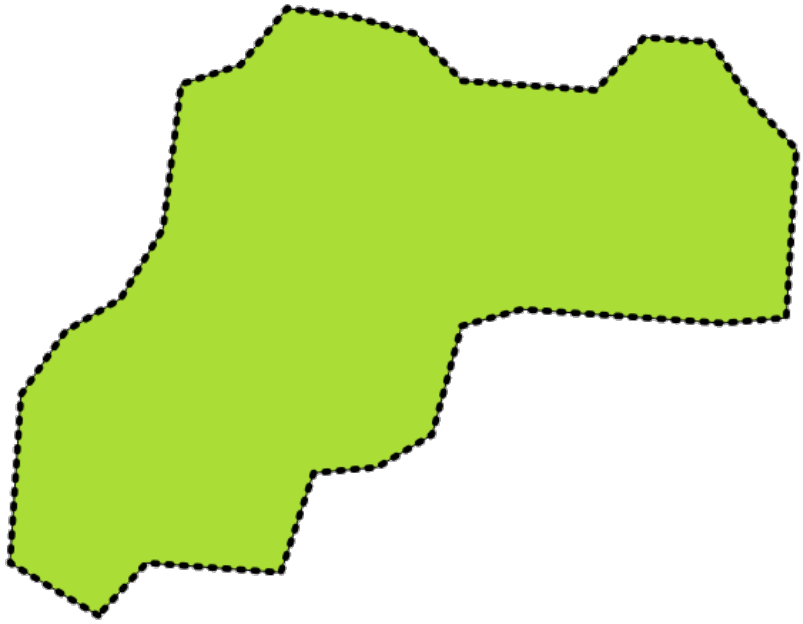


Figura 9.5: Límite (línea negra discontinua) de la capa del polígono de origen

Otras lecturas: algoritmo *Contorno*

bounds

Devuelve una geometría que representa el cuadro delimitador de una geometría de entrada. Los cálculos están en el Sistema de Referencia eSpacial de esta geometría.

Sintaxis	bounds(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `bounds(@geometry)` → cuadro delimitador de la geometría del objeto espacial actual. • `geom_to_wkt(bounds(geom_from_wkt('Polygon((1 1, 0 0, -1 1, 1 1))')))` → "Polygon ((-1 0, 1 0, 1 1, -1 1, -1 0))"

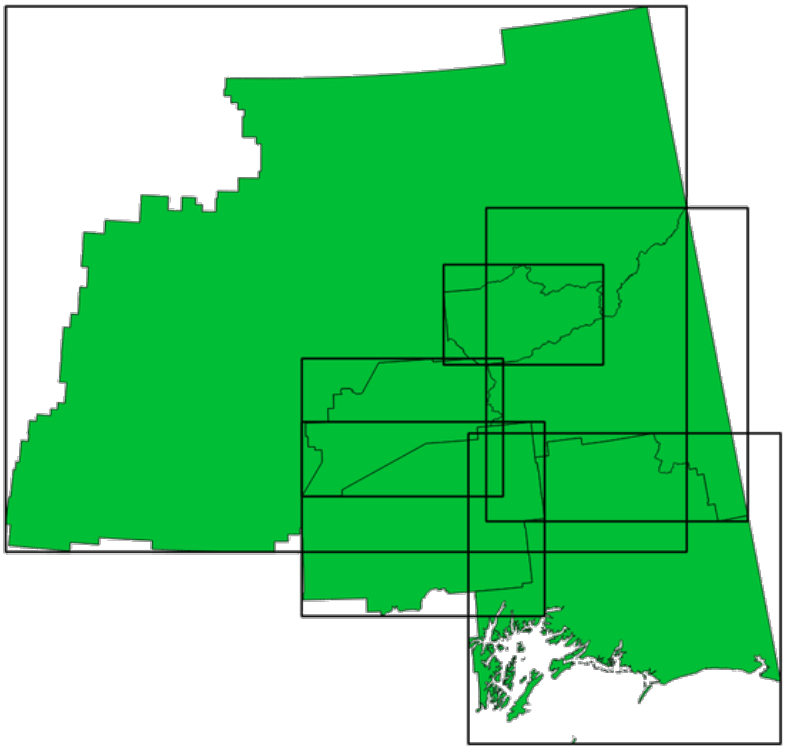


Figura 9.6: Las líneas negras representan los cuadros delimitadores de cada unidad poligonal

Otras lecturas: algoritmo *Recuadros delimitadores*

bounds_height

Devuelve la altura del cuadro delimitador de una geometría. Los cálculos están en el Sistema de Referencia eSpacial de esta geometría.

Sintaxis	bounds_height(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `bounds_height(@geometry)` → altura del cuadro delimitador de la geometría del objeto espacial actual. • `bounds_height(geom_from_wkt('Polygon((1 1, 0 0, -1 1, 1 1))'))` → 1

bounds_width

Devuelve el ancho del cuadro delimitador de una geometría. Los cálculos están en el Sistema de Referencia eSpacial de esta geometría.

Sintaxis	bounds_width(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `bounds_width(@geometry)` → anchura del cuadro delimitador de la geometría del objeto espacial actual. • `bounds_width(geom_from_wkt('Polygon((1 1, 0 0, -1 1, 1 1))'))` → 2

buffer

Devuelve una geometría que representa todos los puntos cuya distancia desde esta geometría es menor o igual que la distancia. Los cálculos están en el Sistema de referencia espacial de esta geometría.

Sintaxis	buffer(geometry, distance, [segments=8], [cap="round"], [join="round"], [miter_limit=2]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **distance** - distancia de búfer en unidades de capa • **segments** - número de segmentos que se utilizarán para representar un cuarto de círculo cuando se utilice un estilo de unión redonda. Un número mayor da como resultado un búfer más suave con más nodos. • **cap** - estilo de tapa final para búfer. Los valores válidos son "round", "flat" or "square" • **join** - estilo de unión para el búfer. Los valores válidos son "round", "bevel" or "miter". • **miter_limit** - límite de distancia de inglete, para usar cuando el estilo de unión se establece en "miter"
Ejemplos	• `buffer(@geometry, 10.5)` → polígono de la geometría del objeto espacial actual con un área de influencia de 10,5 unidades.

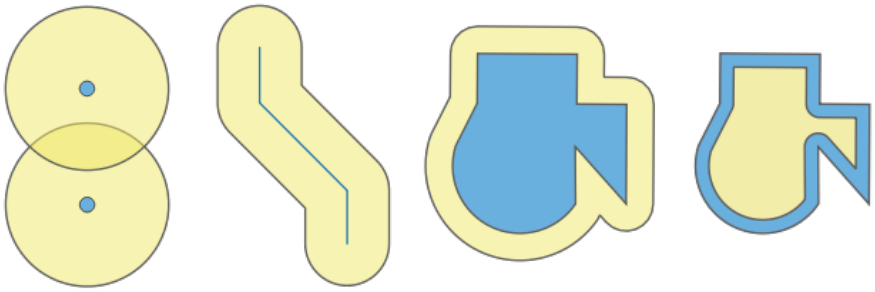


Figura 9.7: Buffer (en amarillo) de puntos, línea, polígono con buffer positivo y polígono con buffer negativo.

Otras lecturas: algoritmo *Buffer*

buffer_by_m

Crea una zona de influencia a lo largo de una geometría de línea donde el diámetro del búfer varía según los valores m en los vértices de la línea.

Sintaxis	buffer_by_m(geometry, [segments=8]) [] marca argumentos opcionales
Argumentos	• **geometry** - geometría de entrada. Debe ser una geometría de (múltiples) líneas con valores m. • **segments** - número de segmentos para aproximar las curvas de un cuarto de círculo en el búfer.
Ejemplos	• `buffer_by_m(geometry:=geom_from_wkt('LINESTRINGM(1 2 0.5, 4 2 0.2)'),segments:=8)` → Un búfer de ancho variable que comienza con un diámetro de 0.5 y termina con un diámetro de 0.2 a lo largo de la geometría de la cadena lineal.

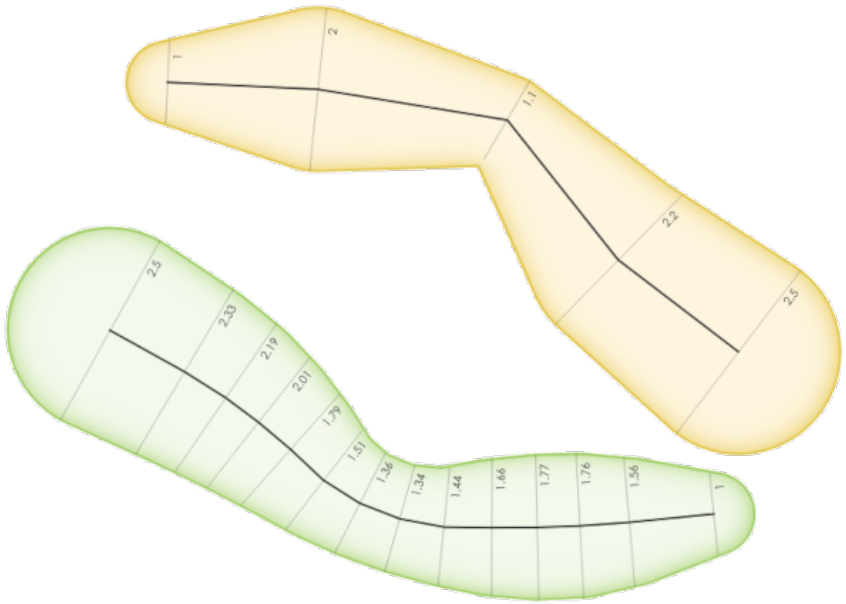


Figura 9.8: Elementos de línea de zona de influencia que utilizan el valor m en los vértices

Otras lecturas: algoritmo *Ancura de buffer variable (por valor de M)*

centroid

Devuelve el centro geométrico de una geometría.

Sintaxis	centroid(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `centroid( @geometry)` → una geometría de puntos

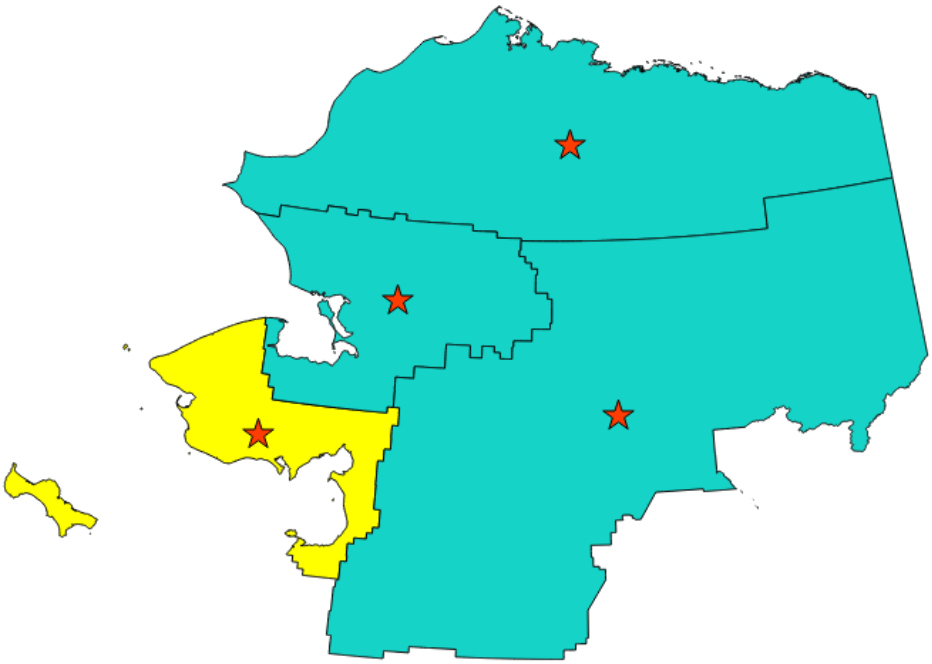


Figura 9.9: Las estrellas rojas representan los centroides de las entidades de la capa entrante.

Otras lecturas: algoritmo *Centroides*

close_line

Devuelve una cadena de línea cerrada de la cadena de línea de entrada agregando el primer punto al final de la línea, si aún no está cerrada. Si la geometría no es una cadena de líneas o una cadena de varias líneas, el resultado será NULL.

Sintaxis	close_line(geometry)
Argumentos	• **geometry** - una geometría de cadena lineal
Ejemplos	• `geom_to_wkt(close_line(geom_from_wkt('LINESTRING(0 0, 1 0, 1 1)')))` → "LineString (0 0, 1 0, 1 1, 0 0)" • `geom_to_wkt(close_line(geom_from_wkt('LINESTRING(0 0, 1 0, 1 1, 0 0)')))` → "LineString (0 0, 1 0, 1 1, 0 0)"

closest_point

Devuelve el punto de geometría1 más cercano a geometría2.

Sintaxis	closest_point(geometry1, geometry2)
Argumentos	• **geometry1** - geometría para la que encontrar el punto más cercano de • **geometry2** - geometría para encontrar el punto más cercano a
Ejemplos	• `geom_to_wkt(closest_point(geom_from_wkt('LINESTRING (20 80, 98 190, 110 180, 50 75 )'),geom_from_wkt('POINT(100 100)')))` → "Point(73.0769 115.384)"

collect_geometries

Recopila un conjunto de geometrías en un objeto de geometría multiparte.

Lista de argumentos variantes

Las partes de geometría son especificadas como argumentos separados a la función.

Sintaxis	collect_geometries(geometry1, geometry2, …)
Argumentos	• **geometry** - una geometría
Ejemplos	• `geom_to_wkt(collect_geometries(make_point(1,2), make_point(3,4), make_point(5,6)))` → "MultiPoint ((1 2),(3 4),(5 6))"

Matriz variante

Partes de geometría son especificadas como una matriz de partes de geometría.

Sintaxis	collect_geometries(array)
Argumentos	• **array** - matriz de objetos de geometría
Ejemplos	• `geom_to_wkt(collect_geometries(array(make_point(1,2), make_point(3,4), make_point(5,6)))` → "MultiPoint ((1 2),(3 4),(5 6))"

Otras lecturas: algoritmo *Coleccionar geometrías*

combine

Devuelve la combinación de dos geometrías.

Sintaxis	combine(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `geom_to_wkt( combine( geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 2 1)' ) ) )` → "MULTILINESTRING((4 4, 2 1), (3 3, 4 4), (4 4, 5 5))" • `geom_to_wkt( combine( geom_from_wkt( 'LINESTRING(3 3, 4 4)' ), geom_from_wkt( 'LINESTRING(3 3, 6 6, 2 1)' ) ) )` → "LINESTRING(3 3, 4 4, 6 6, 2 1)"

concave_hull

Devuelve un polígono posiblemente cóncavo que contiene todos los puntos de la geometría

Sintaxis	concave_hull(geometry, target_percent, [allow_holes=False]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **target_percent** - el porcentaje de área del casco convexo al que intenta aproximarse la solución. Un target_percent de 1 da el mismo resultado que el casco convexo. Un target_percent entre 0 y 0,99 produce un resultado que debería tener un área menor que el casco convexo. • **allow_holes** - argumento opcional que especifica si se permiten agujeros dentro de la geometría de salida. El valor predeterminado es FALSE; establezca TRUE para permitir la inclusión de agujeros en la geometría de salida.
Ejemplos	• `geom_to_wkt(concave_hull(geom_from_wkt('MULTILINESTRING((106 164,30 112,74 70,82 112,130 94,130 62,122 40,156 32,162 76,172 88),(132 178,134 148,128 136,96 128,132 108,150 130,170 142,174 110,156 96,158 90,158 88),(22 64,66 28,94 38,94 68,114 76,112 30,132 10,168 18,178 34,186 52,184 74,190 100,190 122,182 148,178 170,176 184,156 164,146 178,132 186,92 182,56 158,36 150,62 150,76 128,88 118))'), 0.99))` → "Polygon ((30 112, 36 150, 92 182, 132 186, 176 184, 190 122, 190 100, 186 52, 178 34, 168 18, 132 10, 112 30, 66 28, 22 64, 30 112))"

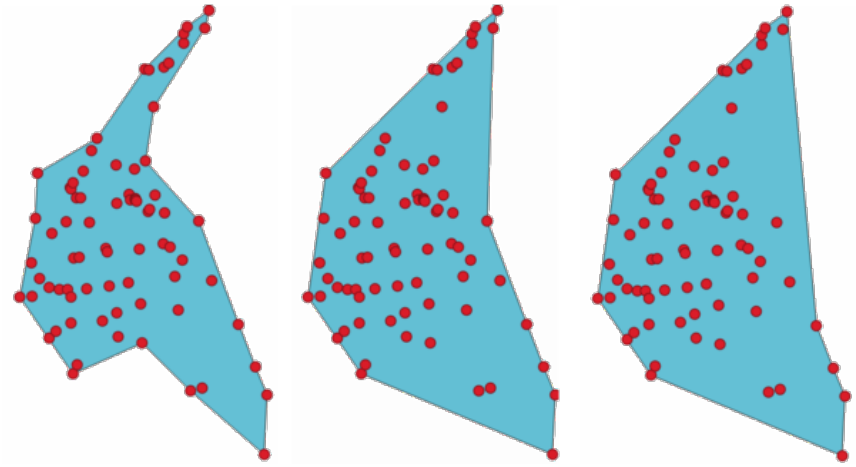


Figura 9.10: Cascos cóncavos con parámetro target_percent creciente

Lectura adicional: algoritmo *convex_hull*, :ref:`qgisconcavehull

contains

Comprueba si una geometría contiene a otra. Devuelve TRUE si y sólo si ningún punto de geometría2 se encuentra en el exterior de geometría1, y al menos un punto del interior de geometría2 se encuentra en el interior de geometría1.

Sintaxis	contains(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `contains( geom_from_wkt( 'POLYGON((0 0, 0 1, 1 1, 1 0, 0 0))' ), geom_from_wkt( 'POINT(0.5 0.5 )' ) )` → TRUE • `contains( geom_from_wkt( 'POLYGON((0 0, 0 1, 1 1, 1 0, 0 0))' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → FALSE

Otras lecturas: *overlay_contains*

convex_hull

Devuelve el casco convexo de una geoemtría. REpresenta la geoemtría convexa mínima que encierra todas las geometrías dentro del conjunto.

Sintaxis	convex_hull(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `geom_to_wkt( convex_hull( geom_from_wkt( 'LINESTRING(3 3, 4 4, 4 10)' ) ) )` → "POLYGON((3 3, 4 10, 4 4, 3 3))"

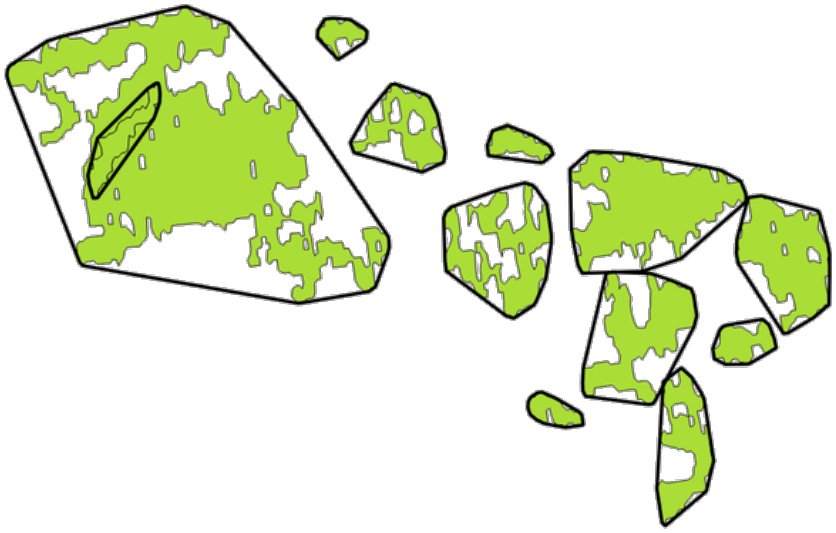


Figura 9.11: Las líneas negras identifican el casco convexo de cada objeto espacial

Lectura adicional: algoritmo *concave_hull*, :ref:`qgisconvexhull

crosses

Comprueba si una geometría cruza a otra. Devuelve TRUE si las geometrías suministradas tienen algunos puntos interiores en común, pero no todos.

Sintaxis	crosses(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `crosses( geom_from_wkt( 'LINESTRING(3 5, 4 4, 5 3)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → TRUE • `crosses( geom_from_wkt( 'POINT(4 5)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → FALSE

Otras lecturas: *overlay_crosses*

densify_by_count

Toma una geometría de capa poligonal o lineal y genera una nueva en la que las geometrías tienen un mayor número de vértices que la original.

Sintaxis	densify_by_count(geometry, vertices)
Argumentos	• **geometry** - una geometría (acepta (multi)líneas o (multi)polígonos). • **vertices** - número de vértices a añadir (por segmento)
Ejemplos	• `geom_to_wkt(densify_by_count(geom_from_wkt('LINESTRING(1 1, 10 1)'), 3))` → LineString (1 1, 3.25 1, 5.5 1, 7.75 1, 10 1)

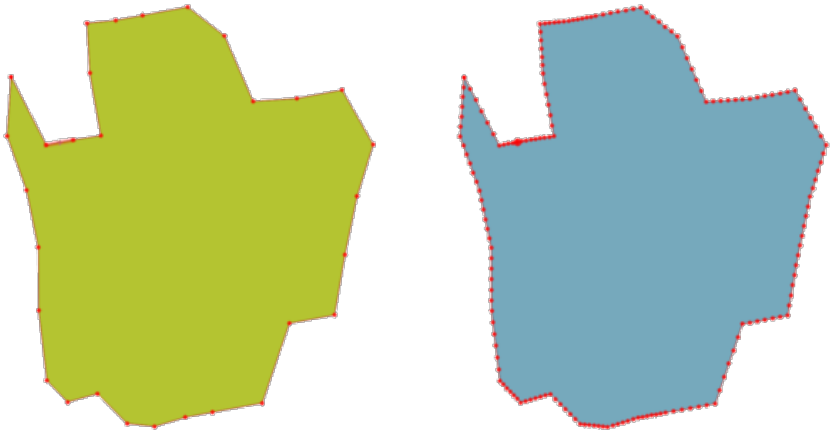


Figura 9.12: Los puntos rojos muestran los vértices antes y despues de densificar

Lectura adicional: algoritmo *Densificar por conteo*

densify_by_distance

Toma una geometría de capa poligonal o lineal y genera una nueva en la que las geometrías se densifican añadiendo vértices adicionales en las aristas que tienen una distancia máxima de la distancia de intervalo especificada.

Sintaxis	densify_by_distance(geometry, distance)
Argumentos	• **geometry** - una geometría (acepta (multi)líneas o (multi)polígonos). • **distance** - intervalo máximo de distancia entre vértices en la geometría de salida
Ejemplos	• `geom_to_wkt(densify_by_distance(geom_from_wkt('LINESTRING(1 1, 10 1)'), 4))` → LineString (1 1, 4 1, 7 1, 10 1)

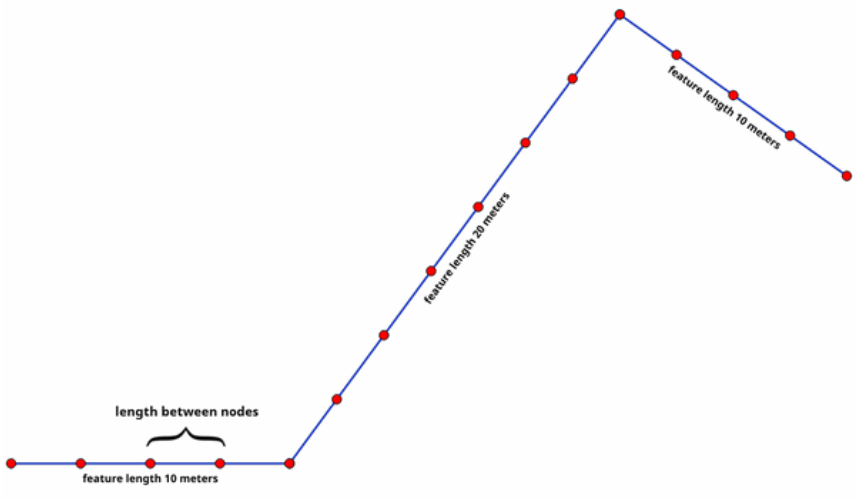


Figura 9.13: Densificar geometría a un intervalo dado

Lectura adicional: algoritmo *Densificar por intervalo*

difference

Devuelve una geometría que representa la parte de geometry1 que no se cruza con geometry2.

Sintaxis	difference(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `geom_to_wkt( difference( geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4)' ) ) )` → "LINESTRING(4 4, 5 5)"

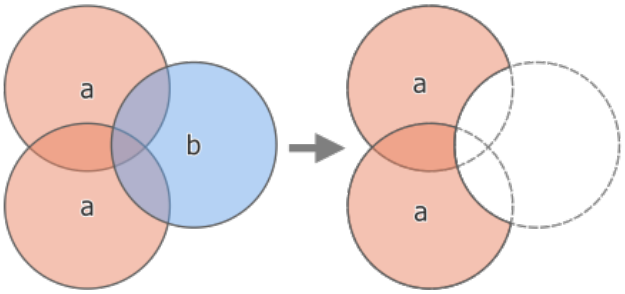


Figura 9.14: Operación de diferencia entre una capa de entrada de dos características «a» y una capa superpuesta de una sola característica «b» (izquierda), lo que da como resultado una nueva capa con las características «a» modificadas (derecha).

Otras lecturas: algoritmo *Diferencia*

disjoint

Comprueba si las geometrías no se cruzan espacialmente. Devuelve TRUE si las geometrías no comparten ningún espacio.

Sintaxis	disjoint(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `disjoint( geom_from_wkt( 'POLYGON((0 0, 0 1, 1 1, 1 0, 0 0 ))' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → TRUE • `disjoint( geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ), geom_from_wkt( 'POINT(4 4)' ))` → FALSE

Otras lecturas: *overlay_disjoint*

distance

Devuelve la distancia mínima (basada en la referencia espacial) entre dos geometrías en unidades proyectadas.

Sintaxis	distance(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `distance( geom_from_wkt( 'POINT(4 4)' ), geom_from_wkt( 'POINT(4 8)' ) )` → 4

distance_to_vertex

Devuelve la distancia a lo largo de la geometría hasta un vértice especificado.

Sintaxis	distance_to_vertex(geometry, vertex)
Argumentos	• **geometry** - una geometría de cadena lineal • **vertex** - índice de vértice, comenzando desde 0; si el valor es negativo, el índice de vértice seleccionado será su recuento total menos el valor absoluto
Ejemplos	• `distance_to_vertex(geometry:=geom_from_wkt('LineString(0 0, 10 0, 10 10)'),vertex:=1)` → 10.0

end_point

Devuelve el último nodo de una geoemtría.

Sintaxis	end_point(geometry)
Argumentos	• **geometry** - objeto geometría
Ejemplos	• `geom_to_wkt(end_point(geom_from_wkt('LINESTRING(4 0, 4 2, 0 2)')))` → “Point (0 2)”

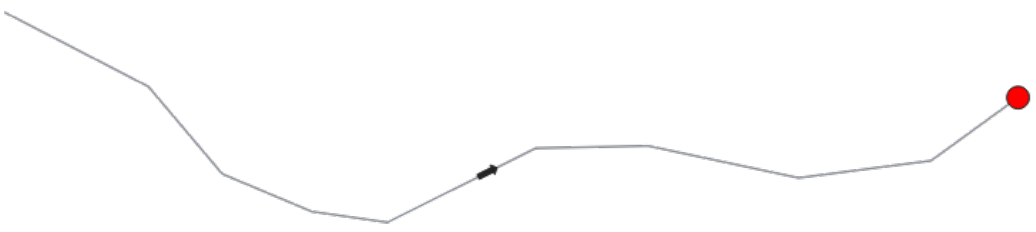


Figura 9.15: Punto final de un objeto espacial línea

Lecturas adicionales: *start_point*, algoritmo :ref:`qgisextractspecificvertices

exif_geotag

Crea una geometría de punto a partir de las etiquetas geográficas exif de un archivo de imagen.

Sintaxis	exif_geotag(path)
Argumentos	• **ruta** - Una ruta de archivo de imagen o un valor de capa de mapa. Si se especifica un valor de capa de mapa, se utilizará el origen de archivo de la capa.
Ejemplos	• `geom_to_wkt(exif_geotag('/my/photo.jpg'))` → "Point (2 4)"

extend

Extiende el inicio y el final de una geometría de cadena lineal en una cantidad especificada. Las líneas se extienden utilizando el rumbo del primer y último segmento de la línea. Para una multilínea, todas las partes están extendidas. Las distancias están en el sistema de referencia espacial de esta geometría.

Sintaxis	extend(geometry, start_distance, end_distance)
Argumentos	• **geometry** - una geometría de cadena lineal(múltiple) • **start_distance** - distancia para extender el inicio de la línea • **end_distance** - distancia para extender el final de la línea.
Ejemplos	• `geom_to_wkt(extend(geom_from_wkt('LineString(0 0, 1 0, 1 1)'),1,2))` → "LineString (-1 0, 1 0, 1 3)" • `geom_to_wkt(extend(geom_from_wkt('MultiLineString((0 0, 1 0, 1 1), (2 2, 0 2, 0 5))'),1,2))` → "MultiLineString ((-1 0, 1 0, 1 3),(3 2, 0 2, 0 7))"

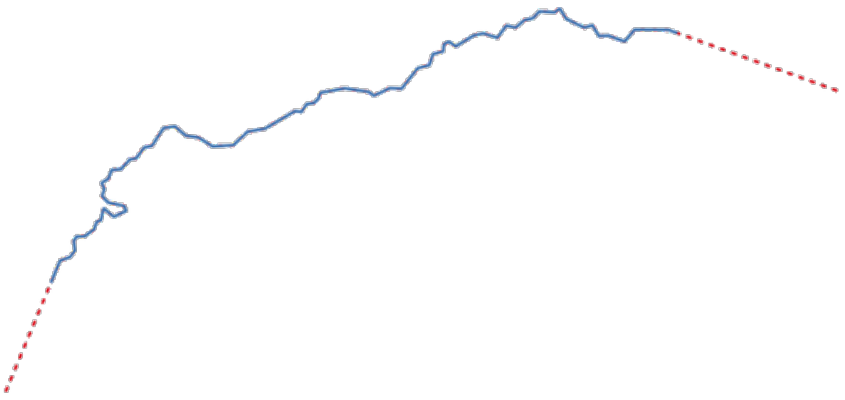


Figura 9.16: Los guiones rojos representan la extensión inicial y final de la capa original

Otras lecturas: algoritmo *Extender líneas*

exterior_ring

Devuelve una cadena lineal que representa el anillo exterior de una geometría poligonal. Si la geometría no es un polígono, el resultado será NULO.

Sintaxis	exterior_ring(geometry)
Argumentos	• **geometry** - una geometría poligonal
Ejemplos	• `geom_to_wkt(exterior_ring(geom_from_wkt('POLYGON((-1 -1, 4 0, 4 2, 0 2, -1 -1),( 0.1 0.1, 0.1 0.2, 0.2 0.2, 0.2, 0.1, 0.1 0.1))')))` → "LineString (-1 -1, 4 0, 4 2, 0 2, -1 -1)"

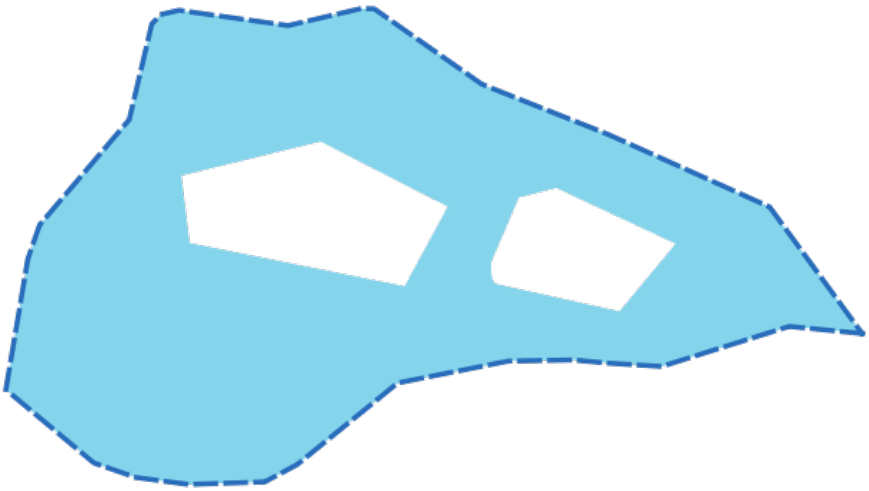


Figura 9.17: La línea discontinua representa el anillo exterior del polígono

Para más información: algoritmo *Contorno*, función de expresión *interior_ring_n*

extrude

Devuelve una versión extruida de la geometría de entrada (Multi-)Curva o (Multi-)Linea con una extensión especificada por x e y.

Sintaxis	extrude(geometry, x, y)
Argumentos	• **geometry** - una curva o geometría lineal • **x** - extensión x, valor numérico • **y** - extensión y, valor numérico
Ejemplos	• `geom_to_wkt(extrude(geom_from_wkt('LineString(1 2, 3 2, 4 3)'), 1, 2))` → "Polygon ((1 2, 3 2, 4 3, 5 5, 4 4, 2 4, 1 2))" • `geom_to_wkt(extrude(geom_from_wkt('MultiLineString((1 2, 3 2), (4 3, 8 3))'), 1, 2))` → "MultiPolygon (((1 2, 3 2, 4 4, 2 4, 1 2)),((4 3, 8 3, 9 5, 5 5, 4 3)))"

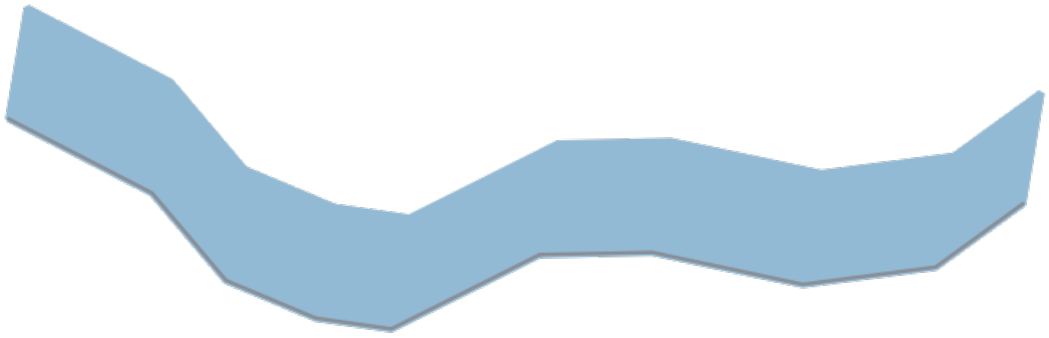


Figura 9.18: Generar un polígono extruyendo una línea con desplazamiento en las direcciones x e y

flip_coordinates

Devuelve una copia de la geometría con las coordenadas xey intercambiadas. Útil para reparar geometrías cuyos valores de latitud y longitud se han invertido.

Sintaxis	flip_coordinates(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `geom_to_wkt(flip_coordinates(make_point(1, 2)))` → "Point (2 1)" • `geom_to_wkt(flip_coordinates(geom_from_wkt('LineString(0 2, 1 0, 1 6)')))` → "LineString (2 0, 0 1, 6 1)"

Otras lecturas: algoritmo *Intercmbiar coordenadas X e Y*

force_polygon_ccw

Obliga a una geometría a respetar la convención según la cual los anillos exteriores van en el sentido contrario a las agujas del reloj y los interiores en el sentido de las agujas del reloj.

Sintaxis	force_polygon_ccw(geometry)
Argumentos	• **geometry** - una geometría. Todas las geometrías que no sean de polígono se devuelven sin cambios.
Ejemplos	• `geom_to_wkt(force_polygon_ccw(geometry:=geom_from_wkt('Polygon ((-1 -1, 0 2, 4 2, 4 0, -1 -1))')))` → "Polygon ((-1 -1, 4 0, 4 2, 0 2, -1 -1))"

Lectura adicional: *force_polygon_cw*, *force_rhr*

force_polygon_cw

Obliga a una geometría a respetar la convención según la cual los anillos exteriores van en el sentido de las agujas del reloj y los interiores en sentido contrario.

Sintaxis	force_polygon_cw(geometry)
Argumentos	• **geometry** - una geometría. Todas las geometrías que no sean de polígono se devuelven sin cambios.
Ejemplos	• `geom_to_wkt(force_polygon_cw(geometry:=geom_from_wkt('POLYGON((-1 -1, 4 0, 4 2, 0 2, -1 -1))')))` → "Polygon ((-1 -1, 0 2, 4 2, 4 0, -1 -1))"

Lectura adicional: *force_polygon_ccw*, *force_rhr*

force_rhr

Obliga a una geometría a respetar la regla de la mano derecha, según la cual el área delimitada por un polígono se encuentra a la derecha del límite. En particular, el anillo exterior se orienta en el sentido de las agujas del reloj y los anillos interiores en sentido contrario. Debido a la inconsistencia en la definición de la Regla de la Mano Derecha en algunos contextos se recomienda utilizar la función explícita force_polygon_cw en su lugar.

Sintaxis	force_rhr(geometry)
Argumentos	• **geometry** - una geometría. Todas las geometrías que no sean de polígono se devuelven sin cambios.
Ejemplos	• `geom_to_wkt(force_rhr(geometry:=geom_from_wkt('POLYGON((-1 -1, 4 0, 4 2, 0 2, -1 -1))')))` → "Polygon ((-1 -1, 0 2, 4 2, 4 0, -1 -1))"

Lecturas adicionales: algoritmo *Fuerza la regla de la mano derecha*, *force_polygon_ccw*, *force_polygon_cw*

geom_from_gml

Devuelve una geoemtría de una representación GML de geoemtría.

Sintaxis	geom_from_gml(gml)
Argumentos	• **gml** - representación GML de una geoemtría como una cadena
Ejemplos	• `geom_from_gml('<gml:LineString srsName="EPSG:4326"><gml:coordinates>4, 4 5,5 6,6</gml:coordinates></gml:LineString>')` → un objeto de geometría lineal

geom_from_wkb

Devuelve una geoemtría creada desde un a representación Well-Known Binary (WKB).

Sintaxis	geom_from_wkb(binary)
Argumentos	• **binary** - representación Well-Known Binary (WKB) de una geometría (como un blob binario)
Ejemplos	• `geom_from_wkb( geom_to_wkb( make_point(4,5) ) )` → un objeto de geoemtría de punto

geom_from_wkt

Devuelve una geoemtría creada desde una representación Well-Known Text (WKT).

Sintaxis	geom_from_wkt(text)
Argumentos	• **text** - representación Well-Known Text (WKT) de una geoemtría
Ejemplos	• `geom_from_wkt( 'POINT(4 5)' )` → un objeto geométrico

geom_to_wkb

Devuelve la representación Well-Known Binary (WKB) de una geometría

Sintaxis	geom_to_wkb(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `geom_to_wkb( @geometry )` → blob binario que contiene un objeto geométrico.

geom_to_wkt

Devuelve la representación Well-Known Text (WKT) de una geoemtría sin metadatos SRID.

Sintaxis	geom_to_wkt(geometry, [precision=8]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **precision** - precisión numérica
Ejemplos	• `geom_to_wkt( make_point(6, 50) )` → "POINT(6 50)" • `geom_to_wkt(centroid(geom_from_wkt('Polygon((1 1, 0 0, -1 1, 1 1))')))` → "POINT(0 0.66666667)" • `geom_to_wkt(centroid(geom_from_wkt('Polygon((1 1, 0 0, -1 1, 1 1))')), 2)` → "POINT(0 0.67)"

$geometry

Devuelve la geometría del objeto espacial actual. Puede utilizarse para procesarla con otras funciones. **AVISO: Esta función está obsoleta. Se recomienda utilizar la variable @geometry en su lugar.

Sintaxis	$geometry
Ejemplos	• `geom_to_wkt( $geometry )` → "POINT(6 50)"

geometry

Devuelve la geometría de una función.

Sintaxis	geometry(feature)
Argumentos	• **feature** - un objeto característico
Ejemplos	• `geometry( @feature )` → la geometría del objeto espacial actual. Es preferible utilizar @geometry. • `geom_to_wkt( geometry( get_feature_by_id( 'streets', 1 ) ) )` → la geometría en WKT de la entidad con el id 1 en la capa «streets», e.g. "POINT(6 50)" • `intersects( @geometry, geometry( get_feature( 'streets', 'name', 'Main St.' ) ) )` → TRUE si el objeto espacial actual intersecta espacialmente el objeto espacial con nombre "Main St." en la capa «streets».

geometry_n

Devuelve una geometría específica de una colección de geometrías, o NULL si la geometría de entrada no es una colección. También devuelve una parte de una geometría multiparte.

Sintaxis	geometry_n(geometry, index)
Argumentos	• **geometry** - colección de geoemtría • **index** - índice de geometría a devolver, donde 1 es la primera geometría de la colección
Ejemplos	• `geom_to_wkt(geometry_n(geom_from_wkt('GEOMETRYCOLLECTION(POINT(0 1), POINT(0 0), POINT(1 0), POINT(1 1))'),3))` → "Point (1 0)"

geometry_type

Devuelve un valor de cadena que describe el tipo de geometría (Punto, Línea o Polígono)

Sintaxis	geometry_type(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `geometry_type( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → "Line" • `geometry_type( geom_from_wkt( 'MULTILINESTRING((2 5, 3 6, 4 8), (1 1, 0 0))') )` → "Line" • `geometry_type( geom_from_wkt( 'POINT(2 5)') )` → "Point" • `geometry_type( geom_from_wkt( 'POLYGON((-1 -1, 4 0, 4 2, 0 2, -1 -1))') )` → "Polygon"

hausdorff_distance

Devuelve la distancia de Hausdorff entre dos geometrías. Esta es básicamente una medida de cuán similares o diferentes son 2 geometrías, con una distancia menor indicando geometrías más similares.

La función se puede ejecutar con un argumento opcional de fracción densificar. Si no se especifica, se utiliza una aproximación a la distancia estándar de Hausdorff. Esta aproximación es lo suficientemente exacta o cercana para un gran subconjunto de casos útiles. Ejemplos de estos son:

- calcular la distancia entre líneas lineales que son aproximadamente paralelas entre sí y aproximadamente iguales en longitud. Esto ocurre en redes lineales coincidentes.
- Prueba de similitud de geometrías.

Si el valor aproximado predeterminado proporcionado por este método es insuficiente, especifique el argumento opcional de fracción densificar. Al especificar este argumento, se realiza una densificación de segmento antes de calcular la distancia discreta de Hausdorff. El parámetro establece la fracción por la cual densificar cada segmento. Cada segmento se dividirá en varios subsegmentos de igual longitud, cuya fracción de la longitud total es la más cercana a la fracción dada. Disminuir el parámetro de fracción de densificación hará que la distancia devuelta se acerque a la distancia real de Hausdorff para las geometrías.

Sintaxis	hausdorff_distance(geometry1, geometry2, [densify_fraction]) [] marca argumentos opcionales
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría • **densify_fraction** - densificar la cantidad de la fracción
Ejemplos	• `hausdorff_distance( geometry1:= geom_from_wkt('LINESTRING (0 0, 2 1)'),geometry2:=geom_from_wkt('LINESTRING (0 0, 2 0)'))` → 2 • `hausdorff_distance( geom_from_wkt('LINESTRING (130 0, 0 0, 0 150)'),geom_from_wkt('LINESTRING (10 10, 10 150, 130 10)'))` → 14.142135623 • `hausdorff_distance( geom_from_wkt('LINESTRING (130 0, 0 0, 0 150)'),geom_from_wkt('LINESTRING (10 10, 10 150, 130 10)'), 0.5)` → 70.0

inclination

Devuelve la inclinación medida desde el cenit (0) al nadir (180) desde el punto_a al punto_b.

Sintaxis	inclination(point_a, point_b)
Argumentos	• **point_a** - geometría de punto • **point_b** - geometría de punto
Ejemplos	• `inclination( make_point( 5, 10, 0 ), make_point( 5, 10, 5 ) )` → 0.0 • `inclination( make_point( 5, 10, 0 ), make_point( 5, 10, 0 ) )` → 90.0 • `inclination( make_point( 5, 10, 0 ), make_point( 50, 100, 0 ) )` → 90.0 • `inclination( make_point( 5, 10, 0 ), make_point( 5, 10, -5 ) )` → 180.0

interior_ring_n

Devuelve un anillo interior específico de una geometría poligonal o NULL si la geometría no es un polígono.

Sintaxis	interior_ring_n(geometry, index)
Argumentos	• **geometry** - geometría poligonal • **index** - índice de interior para volver, donde 1 es el primer anillo interior
Ejemplos	• `geom_to_wkt(interior_ring_n(geom_from_wkt('POLYGON((-1 -1, 4 0, 4 2, 0 2, -1 -1),(-0.1 -0.1, 0.4 0, 0.4 0.2, 0 0.2, -0.1 -0.1),(-1 -1, 4 0, 4 2, 0 2, -1 -1))'),1))` → "LineString (-0.1 -0.1, 0.4 0, 0.4 0.2, 0 0.2, -0.1 -0.1))"

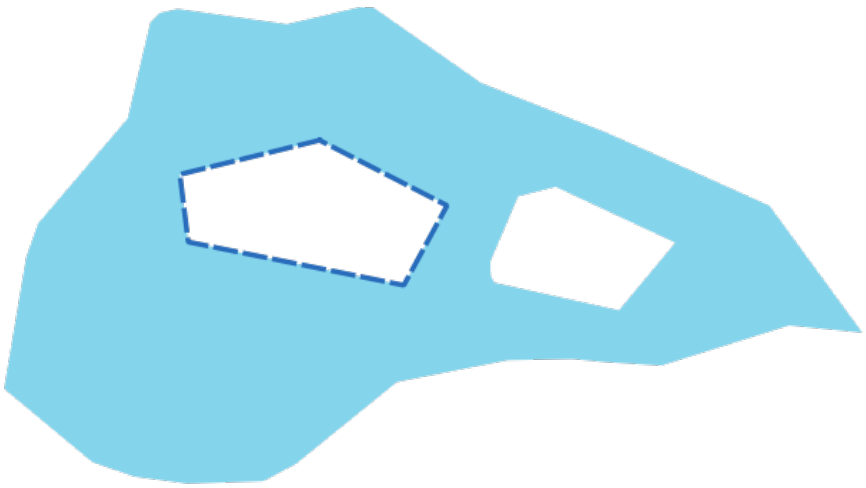


Figura 9.19: La línea discontinua representa un anillo interior específico del polígono.

Para más información: *exterior_ring*

intersection

Devuelve una geometría que representa la parte compartida de dos geometrías.

Sintaxis	intersection(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `geom_to_wkt( intersection( geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4)' ) ) )` → "LINESTRING(3 3, 4 4)" • `geom_to_wkt( intersection( geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ), geom_from_wkt( 'MULTIPOINT(3.5 3.5, 4 5)' ) ) )` → "POINT(3.5 3.5)"

Otras lecturas: algoritmo *Intersección*

intersects

Comprueba si una geometría interseca a otra. Devuelve TRUE si las geometrías se intersecan espacialmente (comparten cualquier parte del espacio) y false si no lo hacen.

Sintaxis	intersects(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `intersects( geom_from_wkt( 'POINT(4 4)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → TRUE • `intersects( geom_from_wkt( 'POINT(4 5)' ), geom_from_wkt( 'POINT(5 5)' ) )` → FALSE

Otras lecturas: *overlay_intersects*

intersects_bbox

Comprueba si la caja delimitadora de una geometría se solapa con la caja delimitadora de otra geometría. Devuelve TRUE si las geometrías intersecan espacialmente la caja delimitadora definida y false si no lo hacen.

Sintaxis	intersects_bbox(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `intersects_bbox( geom_from_wkt( 'POINT(4 5)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → TRUE • `intersects_bbox( geom_from_wkt( 'POINT(6 5)' ), geom_from_wkt( 'POLYGON((3 3, 4 4, 5 5, 3 3))' ) )` → FALSE

is_closed

Devuelve TRUE si una cadena de líneas está cerrada (coinciden os puntos de inicio y final), o false si la cadena de líneas no está cerradai. Si la geometría no es un a cadena de líneas el resultado será NULL.

Sintaxis	is_closed(geometry)
Argumentos	• **geometry** - una geometría de cadena lineal
Ejemplos	• `is_closed(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)'))` → FALSE • `is_closed(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2, 0 0)'))` → TRUE

is_empty

Devuelve TRUE si una geometría está vacía (sin coordenadas), false sí la geometría no está vacía y NULL sí no hay geometría. Vea también is_empty_or_null.

Sintaxis	is_empty(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `is_empty(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)'))` → FALSE • `is_empty(geom_from_wkt('LINESTRING EMPTY'))` → TRUE • `is_empty(geom_from_wkt('POINT(7 4)'))` → FALSE • `is_empty(geom_from_wkt('POINT EMPTY'))` → TRUE

Otras lecturas: *is_empty_or_null*

is_empty_or_null

Devuelve TRUE si una geometría es NULL o está vacía (sin coordenadas) o falso en caso contrario. Esta función es como la expresión "@geometry IS NULL o is_empty(@geometry)".

Sintaxis	is_empty_or_null(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `is_empty_or_null(NULL)` → TRUE • `is_empty_or_null(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)'))` → FALSE • `is_empty_or_null(geom_from_wkt('LINESTRING EMPTY'))` → TRUE • `is_empty_or_null(geom_from_wkt('POINT(7 4)'))` → FALSE • `is_empty_or_null(geom_from_wkt('POINT EMPTY'))` → TRUE

Otras lecturas: *is_empty*, *NULL*

is_multipart

Devuelve TRUE sí la geometría es de tipo Multi.

Sintaxis	is_multipart(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `is_multipart(geom_from_wkt('MULTIPOINT ((0 0),(1 1),(2 2))'))` → TRUE • `is_multipart(geom_from_wkt('POINT (0 0)'))` → FALSE

is_valid

Devuelve TRUE sí la geometría es válida; sí está bien formada en 2D de acuerdo a las reglas OGC.

Sintaxis	is_valid(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `is_valid(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2, 0 0)'))` → TRUE • `is_valid(geom_from_wkt('LINESTRING(0 0)'))` → FALSE

Lectura adicional: *make_valid*, algoritmo :ref:`qgischeckvalidity

$length

Devuelve la longitud de una cadena de líneas. Si necesita la longitud de un borde de un polígono, use $perimeter en su lugar. La longitud calculada por esta función respeta tanto la configuración del elipsoide del proyecto actual como la configuración de la unidad de distancia. Por ejemplo, si se ha establecido un elipsoide para el proyecto, la longitud calculada será elipsoidal, y si no se establece ningún elipsoide, la longitud calculada será planimétrica.

Sintaxis	$length
Ejemplos	• `$length` → 42.4711

length

Devuelve el número de caracteres de una cadena o la longitud de una cadena de líneas geométricas.

Variante de cadena

Devuelve el número de caracteres de una cadena.

Sintaxis	length(string)
Argumentos	• **string** - cadena de la que contar caracteres
Ejemplos	• `length('hello')` → 5

Variante Geometría

Calcula la longitud de un objeto de línea geométrica. Los cálculos son siempre planimétricos en el Sistema de referencia espacial (SRS) de esta geometría, y las unidades de la longitud devuelta coincidirán con las unidades del SRS. Esto difiere de los cálculos realizados por la función $length, que realizará cálculos elipsoidales basados en la configuración de unidad de distancia y elipsoide del proyecto.

Sintaxis	length(geometry)
Argumentos	• **geometry** - objeto geométrico lineal
Ejemplos	• `length(geom_from_wkt('LINESTRING(0 0, 4 0)'))` → 4.0

Otras lecturas: *straight_distance_2d*

length3D

Calcula la longitud 3D de un objeto de línea de geometría. Si la geometría no es un objeto de línea 3D, devuelve su longitud 2D. Los cálculos siempre son planimétricos en el Sistema de referencia espacial (SRS) de esta geometría, y las unidades de la longitud devuelta coincidirán con las unidades del SRS. Esto difiere de los cálculos realizados por la función $length, que realizará cálculos elipsoidales basados en la configuración de unidad de distancia y elipsoide del proyecto.

Sintaxis	length3D(geometry)
Argumentos	• **geometry** - objeto geométrico lineal
Ejemplos	• `length3D(geom_from_wkt('LINESTRINGZ(0 0 0, 3 0 4)'))` → 5.0

line_interpolate_angle

Devuelve el ángulo paralelo a la geometría a una distancia especificada a lo largo de una geometría de cadena lineal. Los ángulos están en grados en sentido horario desde el norte.

Sintaxis	line_interpolate_angle(geometry, distance)
Argumentos	• **geometry** - una geometría de cadena lineal • **distance** - distancia a lo largo de la línea en la que interpolar el ángulo
Ejemplos	• `line_interpolate_angle(geometry:=geom_from_wkt('LineString(0 0, 10 0)'),distance:=5)` → 90.0

line_interpolate_point

Devuelve el punto interpolado por una distancia especificada a lo largo de una geometría de cadena lineal.

Sintaxis	line_interpolate_point(geometry, distance)
Argumentos	• **geometry** - una geometría de cadena lineal • **distance** - distancia a lo largo de la línea para interpolar
Ejemplos	• `geom_to_wkt(line_interpolate_point(geometry:=geom_from_wkt('LineString(0, 8 0)'), distance:=5))` → "Point (5 0)" • `geom_to_wkt(line_interpolate_point(geometry:=geom_from_wkt('LineString(0, 1 1, 2 0)'), distance:=2.1))` → "Point (1.48492424 0.51507576)" • `geom_to_wkt(line_interpolate_point(geometry:=geom_from_wkt('LineString(0, 1 0)'), distance:=2))` → NULL

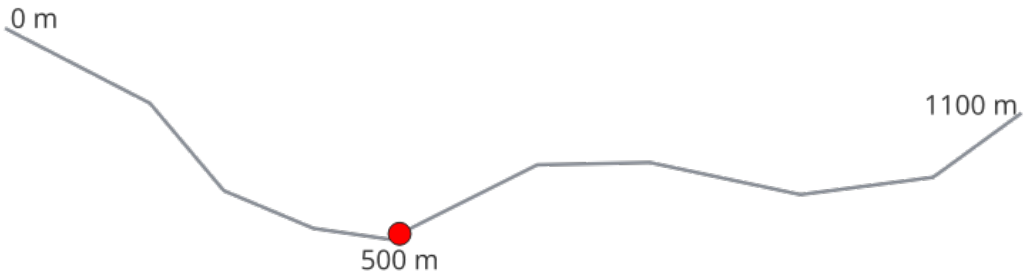


Figura 9.20: Punto interpolado a 500 m del comienzo de la línea.

Otras lecturas: algoritmo *Interpolar puntos en línea*

line_interpolate_point_by_m

Devuelve el punto interpolado por un valor M coincidente a lo largo de una geometría de cadena de líneas.

Sintaxis	line_interpolate_point_by_m(geometry, m, [use_3d_distance=false]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría de cadena lineal • **m** - un valor M • **use_3d_distance** - controla si se deben utilizar distancias 2D o 3D entre vértices durante la interpolación (esta opción solo se tiene en cuenta para líneas con valores z).
Ejemplos	• `geom_to_wkt(line_interpolate_point_by_m(geom_from_wkt('LineStringM(0 0 0, 10 10 10)'), m:=5))` → "Point (5 5)"

line_locate_m

Devuelve la distancia a lo largo de una cadena de líneas correspondiente al primer valor M interpolado coincidente.

Sintaxis	line_locate_m(geometry, m, [use_3d_distance=false]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría de cadena lineal • **m** - un valor M • **use_3d_distance** - controla si se deben utilizar distancias 2D o 3D entre vértices durante la interpolación (esta opción solo se tiene en cuenta para líneas con valores z).
Ejemplos	• `line_locate_m(geometry:=geom_from_wkt('LineStringM(0 0 0, 10 10 10)'),m:=5)` → 7.07106

line_locate_point

Devuelve la distancia a lo largo de una cadena lineal correspondiente a la posición más cercana a la que la cadena lineal llega a una geometría de punto especificada.

Sintaxis	line_locate_point(geometry, point)
Argumentos	• **geometry** - una geometría de cadena lineal • **point** - geometría de puntos para localizar la posición más cercana a la cadena de líneas
Ejemplos	• `line_locate_point(geometry:=geom_from_wkt('LineString(0 0, 10 0)'),point:=geom_from_wkt('Point(5 0)'))` → 5.0

line_merge

Devuelve una geometría LineString o MultiLineString, donde cualquier LineString conectado de la geometría de entrada se ha fusionado en una única cadena lineal. Esta función devolverá NULL si se le pasa una geometría que no sea LineString/MultiLineString.

Sintaxis	line_merge(geometry)
Argumentos	• **geometry** - a LineString/MultiLineString geometry
Ejemplos	• `geom_to_wkt(line_merge(geom_from_wkt('MULTILINESTRING((0 0, 1 1),(1 1, 2 2))')))` → “LineString(0 0,1 1,2 2)” • `geom_to_wkt(line_merge(geom_from_wkt('MULTILINESTRING((0 0, 1 1),(11 1, 21 2))')))` → “MultiLineString((0 0, 1 1),(11 1, 21 2)”

line_substring

Devuelve la parte de una geometría de línea (o curva) que se encuentra entre las distancias inicial y final especificadas (medidas desde el principio de la línea). Los valores Z y M se interpolan linealmente a partir de los valores existentes.

Sintaxis	line_substring(geometry, start_distance, end_distance)
Argumentos	• **geometry** - una geometría de línea o curva • **start_distance** - distancia al inicio de la subcadena • **end_distance** - distancia a final de subcadena
Ejemplos	• `geom_to_wkt(line_substring(geometry:=geom_from_wkt('LineString(0 0, 10 0)'),start_distance:=2,end_distance:=6))` → "LineString (2 0,6 0)"



Figura 9.21: Línea de subcadena con distancia de inicio establecida a 0 metros y distancia de finalización a 250 metros.

Otras lecturas: algoritmo *Subcadena de línea*

m

Devuelve el valor m (medida) de un punto de geometría.

Sintaxis	m(geometry)
Argumentos	• **geometry** - una geometría puntual
Ejemplos	• `m( geom_from_wkt( 'POINTM(2 5 4)' ) )` → 4

m_at

Recupera una coordenada m de la geometría, o NULL si la geometría no tiene valor m.

Sintaxis	m_at(geometry, vertex)
Argumentos	• **geometry** - objeto geometría • **vértice** - índice del vértice de la geometría (los índices comienzan en 0; los valores negativos se aplican a partir del último índice, comenzando en -1)
Ejemplos	• `m_at(geom_from_wkt('LineStringZM(0 0 0 0, 10 10 0 5, 10 10 0 0)'), 1)` → 5

m_max

Devuelve el valor m máximo (medida) de una geoemtría.

Sintaxis	m_max(geometry)
Argumentos	• **geometry** - una geometría con valor m
Ejemplos	• `m_max( make_point_m( 0,0,1 ) )` → 1 • `m_max(make_line( make_point_m( 0,0,1 ), make_point_m( -1,-1,2 ), make_point_m( -2,-2,0 ) ) )` → 2

m_min

Devuelve el valor mínimo m (medida) de una geometría.

Sintaxis	m_min(geometry)
Argumentos	• **geometry** - una geometría con valor m
Ejemplos	• `m_min( make_point_m( 0,0,1 ) )` → 1 • `m_min(make_line( make_point_m( 0,0,1 ), make_point_m( -1,-1,2 ), make_point_m( -2,-2,0 ) ) )` → 0

main_angle

Devuelve el ángulo del eje largo (en el sentido de las agujas del reloj, en grados desde el norte) del rectángulo delimitador mínimo orientado, que cubre completamente la geometría.

Sintaxis	main_angle(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `main_angle(geom_from_wkt('Polygon ((321577 129614, 321581 129618, 321585 129615, 321581 129610, 321577 129614))'))` → 38.66

make_circle

Crea un polígono circular.

Sintaxis	make_circle(center, radius, [segments=36]) [] marca argumentos opcionales
Argumentos	• **center** - punto central del círculo • **radius** - radio del círculo • **segments** - argumento opcional para la segmentación de polígonos. Por defecto, este valor es 36
Ejemplos	• `geom_to_wkt(make_circle(make_point(10,10), 5, 4))` → "Polygon ((10 15, 15 10, 10 5, 5 10, 10 15))" • `geom_to_wkt(make_circle(make_point(10,10,5), 5, 4))` → "PolygonZ ((10 15 5, 15 10 5, 10 5 5, 5 10 5, 10 15 5))" • `geom_to_wkt(make_circle(make_point(10,10,5,30), 5, 4))` → "PolygonZM ((10 15 5 30, 15 10 5 30, 10 5 5 30, 5 10 5 30, 10 15 5 30))"

make_ellipse

Crea un polígono elíptico.

Sintaxis	make_ellipse(center, semi_major_axis, semi_minor_axis, azimuth, [segments=36]) [] marca argumentos opcionales
Argumentos	• **center** - punto central de la elipse • **semi_major_axis** - semi-eje mayor de la elipse • **semi_minor_axis** - semi-eje menor de la elipse • **azimuth** - orientación de la elipse • **segments** - argumento opcional para la segmentación de polígonos. Por defecto, este valor es 36
Ejemplos	• `geom_to_wkt(make_ellipse(make_point(10,10), 5, 2, 90, 4))` → "Polygon ((15 10, 10 8, 5 10, 10 12, 15 10))" • `geom_to_wkt(make_ellipse(make_point(10,10,5), 5, 2, 90, 4))` → "PolygonZ ((15 10 5, 10 8 5, 5 10 5, 10 12 5, 15 10 5))" • `geom_to_wkt(make_ellipse(make_point(10,10,5,30), 5, 2, 90, 4))` → "PolygonZM ((15 10 5 30, 10 8 5 30, 5 10 5 30, 10 12 5 30, 15 10 5 30))"

make_line

Crea una geometría de línea a partir de una serie de geometrías de puntos.

Lista de argumentos variantes

Los vértices de línea se especifican como argumentos separados para la función.

Sintaxis	make_line(point1, point2, …)
Argumentos	• **point** -una geometría puntual (o matriz de puntos)
Ejemplos	• `geom_to_wkt(make_line(make_point(2,4),make_point(3,5)))` → "LineString (2 4, 3 5)" • `geom_to_wkt(make_line(make_point(2,4),make_point(3,5),make_point(9,7)))` → "LineString (2 4, 3 5, 9 7)"

Matriz variante

Los vértices de línea se especifican como una matriz de puntos.

Sintaxis	make_line(array)
Argumentos	• **array** - colección de puntos
Ejemplos	• `geom_to_wkt(make_line(array(make_point(2,4),make_point(3,5),make_point(9,7))))` → "LineString (2 4, 3 5, 9 7)"

make_point

Crea una geometría de punto a partir de un valor xey (y z y m opcionales).

Sintaxis	make_point(x, y, [z], [m]) [] marca argumentos opcionales
Argumentos	• **x** - coordenada x de punto • **y** - coordenada y de punto • **z** - coordenada z de punto opcional • **m** - coordenada m de punto opcional
Ejemplos	• `geom_to_wkt(make_point(2,4))` → "Point (2 4)" • `geom_to_wkt(make_point(2,4,6))` → "PointZ (2 4 6)" • `geom_to_wkt(make_point(2,4,6,8))` → "PointZM (2 4 6 8)"

make_point_m

Crea una geometría de punto a partir de una coordenada x, y y un valor m.

Sintaxis	make_point_m(x, y, m)
Argumentos	• **x** - coordenada x de punto • **y** - coordenada y de punto • **m** - valor m de punto
Ejemplos	• `geom_to_wkt(make_point_m(2,4,6))` → "PointM (2 4 6)"

make_polygon

Crea una geometría poligonal a partir de un anillo exterior y una serie opcional de geometrías de anillo interior.

Sintaxis	make_polygon(outerRing, [innerRing1], [innerRing2], …) [] marca argumentos opcionales
Argumentos	• **outerRing** - geometría de línea cerrada para el anillo exterior del polígono • **innerRing** - geometría de línea cerrada opcional para anillo interior
Ejemplos	• `geom_to_wkt(make_polygon(geom_from_wkt('LINESTRING( 0 0, 0 1, 1 1, 1 0, 0 0 )')))` → "Polygon ((0 0, 0 1, 1 1, 1 0, 0 0))" • `geom_to_wkt(make_polygon(geom_from_wkt('LINESTRING( 0 0, 0 1, 1 1, 1 0, 0 0 )'),geom_from_wkt('LINESTRING( 0.1 0.1, 0.1 0.2, 0.2 0.2, 0.2 0.1, 0.1 0.1 )'), geom_from_wkt('LINESTRING( 0.8 0.8, 0.8 0.9, 0.9 0.9, 0.9 0.8, 0.8 0.8 )')))` → "Polygon ((0 0, 0 1, 1 1, 1 0, 0 0),(0.1 0.1, 0.1 0.2, 0.2 0.2, 0.2 0.1, 0.1 0.1),(0.8 0.8, 0.8 0.9, 0.9 0.9, 0.9 0.8, 0.8 0.8))"

make_rectangle_3points

Crea un rectángulo desde 3 puntos.

Sintaxis	make_rectangle_3points(point1, point2, point3, [option=0]) [] marca argumentos opcionales
Argumentos	• **point1** - Primer punto. • **point2** - Segundo punto. • **point3** - Tercer punto. • **option** - Un argumento opcional para construir el rectángulo. Por defecto, este valor es 0. El valor puede ser 0 (distancia) o 1 (proyectado). Distancia de opción: la segunda distancia es igual a la distancia entre el segundo y el tercer punto. Opción proyectada: Segunda distancia es igual a la distancia de la proyección perpendicular del tercer punto en el segmento o su extensión.
Ejemplos	• `geom_to_wkt(make_rectangle_3points(make_point(0, 0), make_point(0,5), make_point(5, 5), 0))` → "Polygon ((0 0, 0 5, 5 5, 5 0, 0 0))" • `geom_to_wkt(make_rectangle_3points(make_point(0, 0), make_point(0,5), make_point(5, 3), 1))` → "Polygon ((0 0, 0 5, 5 5, 5 0, 0 0))"

make_regular_polygon

Crea un polígono regular.

Sintaxis	make_regular_polygon(center, radius, number_sides, [circle=0]) [] marca argumentos opcionales
Argumentos	• **center** - centro del polígono regular • **radius** - segundo punto. El primero si está inscrito el polígono regular. El punto medio del primer lado si el polígono regular está circunscrito. • **number_sides** - Número de lados/bordes del polígono regular • **circle** - Argumento opcional para construir el polígono regular. Por defecto, este valor es 0. El valor puede ser 0 (inscrito) o 1 (circunscrito)
Ejemplos	• `geom_to_wkt(make_regular_polygon(make_point(0,0), make_point(0,5), 5))` → "Polygon ((0 5, 4.76 1.55, 2.94 -4.05, -2.94 -4.05, -4.76 1.55, 0 5))" • `geom_to_wkt(make_regular_polygon(make_point(0,0), project(make_point(0,0), 4.0451, radians(36)), 5))` → "Polygon ((0 5, 4.76 1.55, 2.94 -4.05, -2.94 -4.05, -4.76 1.55, 0 5))"

make_square

Crea un cuadrado a partir de diagonal.

Sintaxis	make_square(point1, point2)
Argumentos	• **point1** - Primer punto de la diagonal • **point2** - último punto de la diagonal
Ejemplos	• `geom_to_wkt(make_square( make_point(0,0), make_point(5, 5)))` → "Polygon ((0 0, -0 5, 5 5, 5 0, 0 0))" • `geom_to_wkt(make_square( make_point(5,0), make_point(5, 5)))` → "Polygon ((5 0, 2.5 2.5, 5 5, 7.5 2.5, 5 0))"

make_triangle

Crea un polígono triangular.

Sintaxis	make_triangle(point1, point2, point3)
Argumentos	• **point1** - Primer punto del triángulo • **point2** - segund punto del triángulo • **point3** - tercer punto del triángulo
Ejemplos	• `geom_to_wkt(make_triangle(make_point(0,0), make_point(5, 5), make_point(0,10)))` → "Triangle ((0 0, 5 5, 0 10, 0 0))" • `geom_to_wkt(boundary(make_triangle(make_point(0,0), make_point(5,5), make_point(0,10))))` → "LineString (0 0, 5 5, 0 10, 0 0)"

make_valid

Devuelve una geometría válida o una geometría vacía si la geometría no se pudiera hacer válida.

Sintaxis	make_valid(geometry, [method=structure], [keep_collapsed=false]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **método** - reparar algoritmo. Puede ser "structure" o "linework". La opción "linework" combina todos los anillos en un conjunto de líneas con nodos y luego extrae todos los polígonos válidos de esa línea. El método "structure" primero hace que todos los anillos sean válidos, luego fusiona los caparazones y resta los agujeros de los caparazones para generar un resultado válido. Asume que los agujeros y los caparazones están categorizados correctamente. • **keep_collapsed** - si se establece a true, los componentes que han colapsado en una dimensionamiento más bajo se mantendrán. Por ejemplo, un anillo colapsa en una línea o una línea colapsa en un punto
Ejemplos	• `geom_to_wkt(make_valid(geom_from_wkt('POLYGON((3 2, 4 1, 5 8, 3 2, 4 2))')))` → "Polígono ((3 2, 5 8, 4 1, 3 2))" • `geom_to_wkt(make_valid(geom_from_wkt('POLYGON((3 2, 4 1, 5 8, 3 2, 4 2))'), 'linework'))` → "GeometryCollection (Polygon ((5 8, 4 1, 3 2, 5 8)),LineString (3 2, 4 2))" • `geom_to_wkt(make_valid(geom_from_wkt('POLYGON((3 2, 4 1, 5 8))'), method:='linework'))` → "Polygon ((3 2, 4 1, 5 8, 3 2))" • `make_valid(geom_from_wkt('LINESTRING(0 0)'))` → Una geometría vacía

Otras lecturas: *is_valid*, *Corregir geometrías* algorithm

minimal_circle

Devuelve el círculo envolvente mínimo de una geometría. Representa el círculo mínimo que encierra todas las geometrías dentro del conjunto.

Sintaxis	minimal_circle(geometry, [segments=36]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **segments** - argumento opcional para la segmentación de polígonos. Por defecto, este valor es 36
Ejemplos	• `geom_to_wkt( minimal_circle( geom_from_wkt( 'LINESTRING(0 5, 0 -5, 2 1)' ), 4 ) )` → "Polygon ((0 5, 5 -0, -0 -5, -5 0, 0 5))" • `geom_to_wkt( minimal_circle( geom_from_wkt( 'MULTIPOINT(1 2, 3 4, 3 2)' ), 4 ) )` → "Polygon ((3 4, 3 2, 1 2, 1 4, 3 4))"

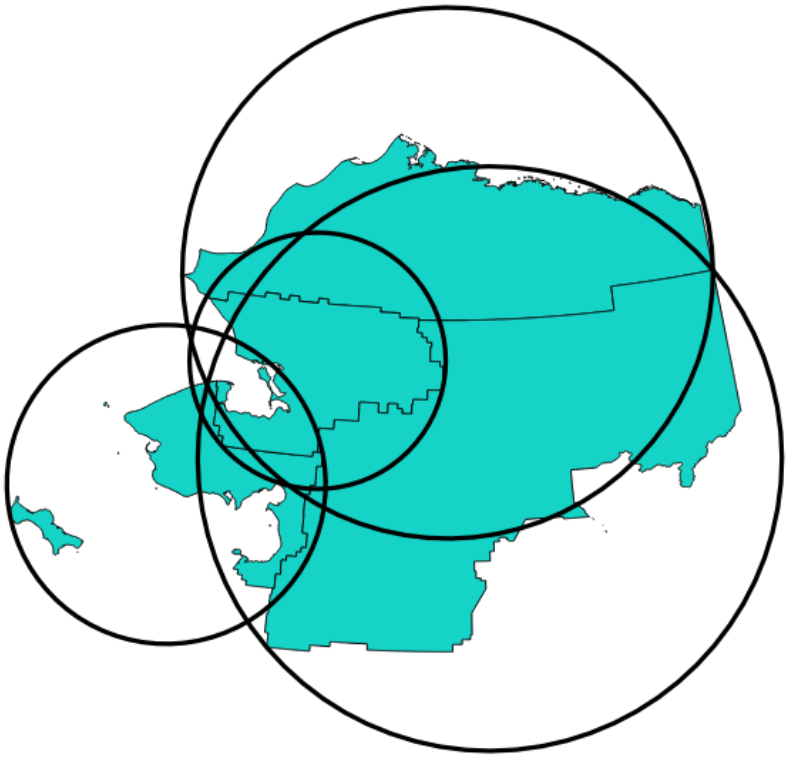


Figura 9.22: Círculo mínimo que encierra cada objeto espacial

Otras lecturas: algoritmo *Cícrculos mínimos cercantes*

nodes_to_points

Devuelve una geometría multipunto que consta de todos los nodos de la geometría de entrada.

Sintaxis	nodes_to_points(geometry, [ignore_closing_nodes=false]) [] marca argumentos opcionales
Argumentos	• **geometry** - objeto geometría • **ignore_closing_nodes** - argumento opcional que especifica si se deben incluir nodos duplicados que cierran líneas o anillos de polígonos. El valor predeterminado es falso, establecido en verdadero para evitar incluir estos nodos duplicados en la colección de salida.
Ejemplos	• `geom_to_wkt(nodes_to_points(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)')))` → "MultiPoint ((0 0),(1 1),(2 2))" • `geom_to_wkt(nodes_to_points(geom_from_wkt('POLYGON((-1 -1, 4 0, 4 2, 0 2, -1 -1))'),true))` → "MultiPoint ((-1 -1),(4 0),(4 2),(0 2))"

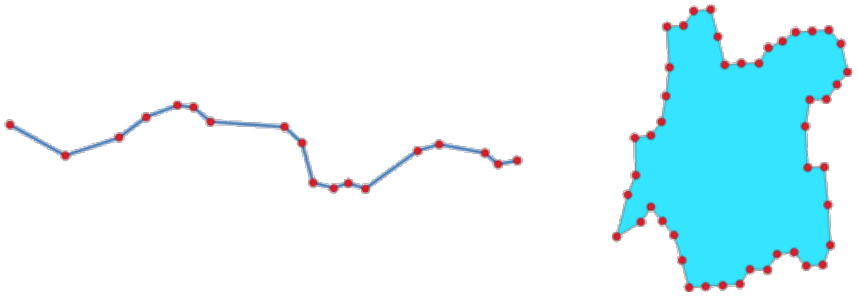


Figura 9.23: Objeto espacial multipunto extraído de los vértices

Otras lecturas: algoritmo *Extraer vértices*

num_geometries

Devuelve el número de geometrías en una colección de geometrías o el número de partes en una geometría multiparte. La función devuelve NULL si la geometría de entrada no es una colección.

Sintaxis	num_geometries(geometry)
Argumentos	• **geometry** - colección de geometría o geometría multiparte
Ejemplos	• `num_geometries(geom_from_wkt('GEOMETRYCOLLECTION(POINT(0 1), POINT(0 0), POINT(1 0), POINT(1 1))'))` → 4 • `num_geometries(geom_from_wkt('MULTIPOINT((0 1), (0 0), (1 0))'))` → 3

num_interior_rings

Devuelve el número de anillos interiores en un polígono o colección de geometría, o NULL si la geometría de entrada no es un polígono o colección.

Sintaxis	num_interior_rings(geometry)
Argumentos	• **geometry** - geoemtría entrante
Ejemplos	• `num_interior_rings(geom_from_wkt('POLYGON((-1 -1, 4 0, 4 2, 0 2, -1 -1),(-0.1 -0.1, 0.4 0, 0.4 0.2, 0 0.2, -0.1 -0.1))'))` → 1

num_points

Devuelve el número de vértices en una geoemtría.

Sintaxis	num_points(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `num_points(@geometry)` → número de vértices en la geometría del objeto espacial actual.

num_rings

Devuelve el número de anillos (incluidos los anillos exteriores) en un polígono o colección de geometría, o NULL si la geometría de entrada no es un polígono o colección.

Sintaxis	num_rings(geometry)
Argumentos	• **geometry** - geoemtría entrante
Ejemplos	• `num_rings(geom_from_wkt('POLYGON((-1 -1, 4 0, 4 2, 0 2, -1 -1),(-0.1 -0.1, 0.4 0, 0.4 0.2, 0 0.2, -0.1 -0.1))'))` → 2

offset_curve

Devuelve una geometría formada desplazando una geometría de cadena de líneas hacia un lado. Las distancias están en el sistema de referencia espacial de esta geometría.

Sintaxis	offset_curve(geometry, distance, [segments=8], [join=1], [miter_limit=2.0]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría de cadena lineal(múltiple) • **distance** - distancia de compensación. Los valores positivos se almacenarán en búfer a la izquierda de las líneas, los valores negativos a la derecha • **segments** - número de segmentos que se utilizarán para representar un cuarto de círculo cuando se utilice un estilo de unión redonda. Un número mayor da como resultado una línea más suave con más nodos. • **join** - join style for corners, where 1 = round, 2 = miter and 3 = bevel • **miter_limit** - límite en la relación de inglete que se usa para esquinas muy afiladas (cuando se usan juntas de inglete solamente)
Ejemplos	• `offset_curve( @geometry, 10.5)` → línea desplazada a la izquierda 10,5 unidades. • `offset_curve( @geometry, -10.5)` → línea desplazada a la derecha 10,5 unidades. • `offset_curve(@geometry, 10.5, segments:=16, join:=1)` → línea desplazada hacia la izquierda 10.5 unidades, utilizando más segmentos para obtener una curva más suave. • `offset_curve(@geometry, 10.5, join:=3)` → línea desplazada a la izquierda 10.5 unidades, utilizando una unión biselada.

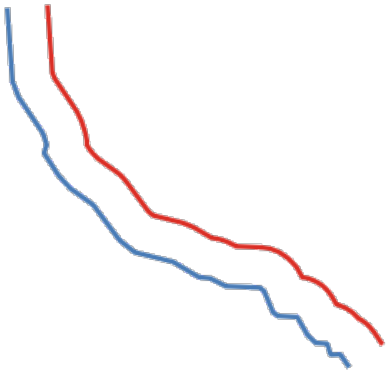


Figura 9.24: En azul la capa fuente, en rojo la compensada

Otras lecturas: algoritmo *Compensado de líneas*

order_parts

Ordena las partes de una MultiGeometría por los criterios dados

Sintaxis	order_parts(geometry, orderby, [ascending=true]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría multitipo • **orderby** - una cadena de expresión definiendo el criterio de orden • **ascending** - booleano, Verdadero para ascendente, Falso para descenso
Ejemplos	• `geom_to_wkt(order_parts(geom_from_wkt('MultiPolygon (((1 1, 5 1, 5 5, 1 5, 1 1)),((1 1, 9 1, 9 9, 1 9, 1 1)))'), 'area(@geometry)', False))` → "MultiPolygon (((1 1, 9 1, 9 9, 1 9, 1 1)),((1 1, 5 1, 5 5, 1 5, 1 1)))" • `geom_to_wkt(order_parts(geom_from_wkt('LineString(1 2, 3 2, 4 3)'), '1', True))` → "LineString(1 2, 3 2, 4 3)"

oriented_bbox

Devuelve una geometría que representa el cuadro delimitador orientado mínimo de una geometría de entrada.

Sintaxis	oriented_bbox(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `geom_to_wkt( oriented_bbox( geom_from_wkt( 'MULTIPOINT(1 2, 3 4, 3 2)' ) ) )` → "Polygon ((3 2, 3 4, 1 4, 1 2, 3 2))"

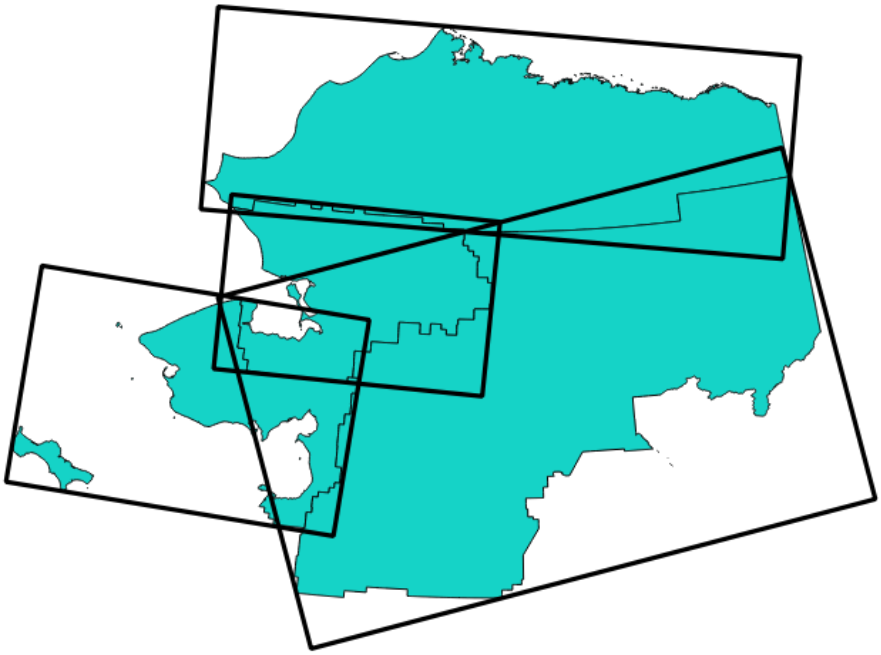


Figura 9.25: Cuador delimitador mínimo orientado

Otras lecturas: algoritmo *Cuador delimitador mínimo orientado*

overlaps

Prueba sí una geometría se superpone a otra. Devuelve TRUE si la geometrías comparten especios, son de la misma dimensión, pero no se contienen completamente una a otra.

Sintaxis	overlaps(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `overlaps( geom_from_wkt( 'LINESTRING(3 5, 4 4, 5 5, 5 3)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → TRUE • `overlaps( geom_from_wkt( 'LINESTRING(0 0, 1 1)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → FALSE

overlay_contains

Devuelve si la entidad actual contiene espacialmente al menos una entidad de una capa de destino o una matriz de resultados basados en expresiones para las entidades de la capa de destino contenidas en la entidad actual.

Más información sobre el predicado «Contiene» de GEOS subyacente, como se describe en la función PostGIS ST_Contains.

Sintaxis	overlay_contains(layer, [expression], [filter], [limit], [cache=false]) [] marca argumentos opcionales
Argumentos	• **capa** - la capa cuya superposición está marcada • **expresión** - una expresión opcional para evaluar las características de la capa de destino. Si no se establece, la función simplemente devolverá un booleano que indica si hay al menos una coincidencia. • **filtrar** - una expresión opcional para filtrar los objetos de destino a verificar. Si no se establece, se marcarán todos los objetos. • **límite** - un entero opcional para limitar el número de objetos elegidos. Si no se establece, todos los objetos elegidos serán devueltos. • **cache** - establezca esto en verdadero para construir un índice espacial local (la mayoría de las veces, esto no es deseado, a menos que esté trabajando con un proveedor de datos particularmente lento)
Ejemplos	• `overlay_contains('regions')` → TRUE si el objeto espacial actual contiene espacialmente una región. • `overlay_contains('regions', filter:= population > 10000)` → TRUE si el objeto espacial actual contiene espacialmente una región con una población superior a 10000. • `overlay_contains('regions', name)` → una matriz de nombres, para las regiones contenidas en la entidad actual • `array_to_string(overlay_contains('regions', name))` → una cadena como una lista de nombres separados por comas, para las regiones contenidas en la entidad actual • `array_sort(overlay_contains(layer:='regions', expression:="name", filter:= population > 10000))` → una matriz ordenada de nombres, para las regiones contenidas en la entidad actual y con una población superior a 10000 • `overlay_contains(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → una matriz de geometrías (en WKT), para un máximo de dos regiones contenidas en la prestación actual.

Otras lecturas: *contains*, *manipulación de array*, algoritmo *Seleccionar por ubicacion*

overlay_crosses

Devuelve si la entidad actual cruza espacialmente al menos una entidad de una capa de destino o una matriz de resultados basados en expresiones para las entidades de la capa de destino cruzadas por la entidad actual.

Más información sobre el predicado «cruces» de GEOS subyacente, como se describe en la función PostGIS`ST_Crosses <https://postgis.net/docs/ST_Crosses.html>`_.

Sintaxis	overlay_crosses(layer, [expression], [filter], [limit], [cache=false]) [] marca argumentos opcionales
Argumentos	• **capa** - la capa cuya superposición está marcada • **expresión** - una expresión opcional para evaluar las características de la capa de destino. Si no se establece, la función simplemente devolverá un booleano que indica si hay al menos una coincidencia. • **filtrar** - una expresión opcional para filtrar los objetos de destino a verificar. Si no se establece, se marcarán todos los objetos. • **límite** - un entero opcional para limitar el número de objetos elegidos. Si no se establece, todos los objetos elegidos serán devueltos. • **cache** - establezca esto en verdadero para construir un índice espacial local (la mayoría de las veces, esto no es deseado, a menos que esté trabajando con un proveedor de datos particularmente lento)
Ejemplos	• `overlay_crosses('regions')` → TRUE si el objeto espacial actual cruza espacialmente una región. • `overlay_crosses('regions', filter:= population > 10000)` → TRUE si el objeto espacial actual atraviesa espacialmente una región con una población superior a 10000. • `overlay_crosses('regions', name)` → una serie de nombres, para las regiones atravesadas por la entidad actual • `array_to_string(overlay_crosses('regions', name))` → una cadena como una lista de nombres separados por comas, para las regiones atravesadas por la entidad actual • `array_sort(overlay_crosses(layer:='regions', expression:="name", filter:= population > 10000))` → una matriz ordenada de nombres, para las regiones atravesadas por la entidad actual y con una población superior a 10000 • `overlay_crosses(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → una matriz de geometrías (en WKT), para un máximo de dos regiones atravesadas por el objeto espacial actual.

Otras lecturas: *crosses*, *manipulación de array*, algoritmo *Seleccionar por ubicacion*

overlay_disjoint

Devuelve si la entidad actual está separada espacialmente de todas las entidades de una capa de destino o una matriz de resultados basados en expresiones para las entidades de la capa de destino que están separadas de la entidad actual.

Más información sobre el predicado «Disjoint» de GEOS subyacente, como se describe en la función PostGIS ST_Disjoint.

Sintaxis	overlay_disjoint(layer, [expression], [filter], [limit], [cache=false]) [] marca argumentos opcionales
Argumentos	• **capa** - la capa cuya superposición está marcada • **expresión** - una expresión opcional para evaluar las características de la capa de destino. Si no se establece, la función simplemente devolverá un booleano que indica si hay al menos una coincidencia. • **filtrar** - una expresión opcional para filtrar los objetos de destino a verificar. Si no se establece, se marcarán todos los objetos. • **límite** - un entero opcional para limitar el número de objetos elegidos. Si no se establece, todos los objetos elegidos serán devueltos. • **cache** - establezca esto en verdadero para construir un índice espacial local (la mayoría de las veces, esto no es deseado, a menos que esté trabajando con un proveedor de datos particularmente lento)
Ejemplos	• `overlay_disjoint('regions')` → TRUE si el objeto espacial actual es espacialmente disjunta de todas las regiones. • `overlay_disjoint('regions', filter:= population > 10000)` → TRUE si el objeto espacial actual es espacialmente disjunto de todas las regiones con una población superior a 10000. • `overlay_disjoint('regions', name)` → una matriz de nombres, para las regiones espacialmente separadas de la entidad actual • `array_to_string(overlay_disjoint('regions', name))` → una cadena como una lista de nombres separados por comas, para las regiones separadas espacialmente de la entidad actual • `array_sort(overlay_disjoint(layer:='regions', expression:="name", filter:= population > 10000))` → una matriz ordenada de nombres, para las regiones espacialmente separadas de la entidad actual y con una población superior a 10000 • `overlay_disjoint(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → una matriz de geometrías (en WKT), para hasta dos regiones espacialmente disjuntas del objeto espacial actual.

Otras lecturas: *disjoint*, *manipulación array*, algoritmo *Seleccionar por ubicacion*

overlay_equals

Returns whether the current feature's geometry is exactly equal to at least one feature's geometry from a target layer, or an array of expression-based results for the features in the target layer whose geometry is exactly equal to the current feature's geometry. Note that the order of vertices matters.

Sintaxis	overlay_equals(layer, [expression], [filter], [limit], [cache=false]) [] marca argumentos opcionales
Argumentos	• **capa** - la capa cuya superposición está marcada • **expresión** - una expresión opcional para evaluar las características de la capa de destino. Si no se establece, la función simplemente devolverá un booleano que indica si hay al menos una coincidencia. • **filtrar** - una expresión opcional para filtrar los objetos de destino a verificar. Si no se establece, se marcarán todos los objetos. • **límite** - un entero opcional para limitar el número de objetos elegidos. Si no se establece, todos los objetos elegidos serán devueltos. • **cache** - establezca esto en verdadero para construir un índice espacial local (la mayoría de las veces, esto no es deseado, a menos que esté trabajando con un proveedor de datos particularmente lento)
Ejemplos	• `overlay_equals('regions')` → TRUE if the current feature's geometry is exactly the same as the geometry of a region • `overlay_equals('regions', filter:= population > 10000)` → TRUE if the current feature's geometry is exactly the same as the geometry of a region whose population is greater than 10000 • `overlay_equals('regions', name)` → an array of names, for the regions exactly equal to the current feature • `array_to_string(overlay_equals('regions', name))` → a string as a comma separated list of names, for the regions exactly equal to the current feature • `array_sort(overlay_equals(layer:='regions', expression:="name", filter:= population > 10000))` → an ordered array of names, for the regions exactly equal to the current feature and with a population greater than 10000 • `overlay_equals(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → an array of geometries (in WKT), for up to two regions exactly equal to the current feature

Nota

This function requires exact equality of geometries, meaning that the geometries "LINESTRING(0 0, 1 1)" and "LINESTRING(1 1, 0 0)" are different.

Otras lecturas: *manipulación array*, algoritmo *Seleccionar por ubicacion*

overlay_intersects

Devuelve si la entidad actual interseca espacialmente al menos una entidad de una capa de destino o una matriz de resultados basados en expresiones para las entidades de la capa de destino intersectadas por la entidad actual.

Más información sobre el predicado subyacente «Intersects» de GEOS, como se describe en la función PostGIS ST_Intersects.

Sintaxis	overlay_intersects(layer, [expression], [filter], [limit], [cache=false], [min_overlap], [min_inscribed_circle_radius], [return_details], [sort_by_intersection_size]) [] marca argumentos opcionales
Argumentos	• **capa** - la capa cuya superposición está marcada • **expresión** - una expresión opcional para evaluar las características de la capa de destino. Si no se establece, la función simplemente devolverá un booleano que indica si hay al menos una coincidencia. • **filtrar** - una expresión opcional para filtrar los objetos de destino a verificar. Si no se establece, se marcarán todos los objetos. • **límite** - un entero opcional para limitar el número de objetos elegidos. Si no se establece, todos los objetos elegidos serán devueltos. • **cache** - establezca esto en verdadero para construir un índice espacial local (la mayoría de las veces, esto no es deseado, a menos que esté trabajando con un proveedor de datos particularmente lento) • **min_overlap** - define un filtro de exclusión opcional: – para polígonos, un área mínima en unidades cuadradas de la característica actual para la intersección. Si la intersección resulta en múltiples polígonos, se devolverá la intersección si al menos uno de los polígonos tiene un área mayor o igual que el valor – para las líneas, una longitud mínima en unidades de objeto espacial actuales. Si la intersección resulta en múltiples líneas, se devolverá la intersección si al menos una de las líneas tiene una longitud mayor o igual al valor. • **radio_círculo_inscrito_mínimo** - define un filtro de exclusión opcional (sólo para polígonos): radio mínimo en unidades de objetos espaciales actuales para el círculo inscrito máximo de la intersección. Si la intersección resulta en múltiples polígonos, la intersección será devuelta si al menos uno de los polígonos tiene un radio para el círculo inscrito máximo mayor o igual al valor. Lea más sobre el predicado GEOS subyacente, tal y como se describe en la función PostGIS ST_MaximumInscribedCircle. Este argumento requiere GEOS >= 3.9. • **return_details** - Póngalo a true para devolver una lista de mapas que contengan (nombres de clave entre comillas) el "id" de la característica, la expresión "resultado" y el valor de "solapamiento" (del elemento mayor en caso de multiparte). También se devuelve el "radio" del círculo máximo inscrito cuando la capa de destino es un polígono. Sólo es válido cuando se utiliza con el parámetro de expresión • **sort_by_intersection_size** - sólo válido cuando se utiliza con una expresión, establece esto a "des" para devolver los resultados ordenados por el valor de solapamiento en orden descendente o establece esto a "asc" para orden ascendente.
Ejemplos	• `overlay_intersects('regions')` → TRUE si el objeto espacial actual intersecta espacialmente una región. • `overlay_intersects('regions', filter:= population > 10000)` → TRUE si el objeto espacial actual intersecta espacialmente una región con una población superior a 10000. • `overlay_intersects('regions', name)` → una matriz de nombres, para las regiones intersectadas por la entidad actual • `array_to_string(overlay_intersects('regions', name))` → una cadena como una lista de nombres separados por comas, para las regiones intersecadas por la entidad actual • `array_sort(overlay_intersects(layer:='regions', expression:="name", filter:= population > 10000))` → una matriz ordenada de nombres, para las regiones intersectadas por la entidad actual y con una población superior a 10000 • `overlay_intersects(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → una matriz de geometrías (en WKT), para un máximo de dos regiones intersectadas por el objeto espacial actual. • `overlay_intersects(layer:='regions', min_overlap:=0.54)` → TRUE si el objeto espacial actual intersecta espacialmente una región y el área de intersección (de al menos una de las partes en caso de multipolígonos) es mayor o igual a 0.54 • `overlay_intersects(layer:='regions', min_inscribed_circle_radius:=0.54)` → TRUE si el objeto espacial actual intersecta espacialmente una región y el radio

Otras lecturas: *intersects*, *manipulación array*, algoritmo *Seleccionar por ubicacion*

overlay_nearest

Devuelve si la entidad actual tiene entidades de una capa de destino dentro de una distancia determinada, o una matriz de resultados basados en expresiones para las entidades en la capa de destino dentro de una distancia de la entidad actual.

Nota: esta función puede ser lenta y consumir mucha memoria para capas grandes.

Sintaxis	overlay_nearest(layer, [expression], [filter], [limit=1], [max_distance], [cache=false]) [] marca argumentos opcionales
Argumentos	• **capa** - la capa destino • **expresión** - una expresión opcional para evaluar las características de la capa de destino. Si no se establece, la función simplemente devolverá un booleano que indica si hay al menos una coincidencia. • **filtro** - una expresión opcional para filtrar las entidades de destino para verificar. Si no se establece, se utilizarán todas las entidades de la capa de destino. • **límite** - un entero opcional para limitar el número de características coincidentes. Si no se configura, solo se devolverá la característica más cercana. Si se establece en -1, devuelve todas las entidades coincidentes. • **max_distance** - una distancia opcional para limitar la búsqueda de entidades coincidentes. Si no se establece, se utilizarán todas las entidades de la capa de destino. • **cache** - establezca esto en verdadero para construir un índice espacial local (la mayoría de las veces, esto no es deseado, a menos que esté trabajando con un proveedor de datos particularmente lento)
Ejemplos	• `overlay_nearest('airports')` → TRUE si la capa «airports» tiene al menos un objeto espacial. • `overlay_nearest('airports', max_distance:= 5000)` → TRUE si hay un aeropuerto a una distancia de 5000 unidades de mapa del objeto espacial actual. • `overlay_nearest('airports', name)` → el nombre del aeropuerto más cercano a la entidad actual, como una matriz • `array_to_string(overlay_nearest('airports', name))` → el nombre del aeropuerto más cercano a la entidad actual, como una cadena • `overlay_nearest(layer:='airports', expression:= name, max_distance:= 5000)` → el nombre del aeropuerto más cercano dentro de una distancia de 5000 unidades de mapa de la entidad actual, como una matriz • `overlay_nearest(layer:='airports', expression:="name", filter:= "Use"='Civilian', limit:=3)` → una variedad de nombres, hasta para los tres aeropuertos civiles más cercanos ordenados por distancia • `overlay_nearest(layer:='airports', expression:="name", limit:= -1, max_distance:= 5000)` → una serie de nombres, para todos los aeropuertos dentro de una distancia de 5000 unidades de mapa de la entidad actual, ordenados por distancia

Otras lecturas: *manipulación array*, algoritmo *Unir atributos por proximidad*

overlay_touches

Devuelve si la entidad actual toca espacialmente al menos una entidad de una capa de destino o una matriz de resultados basados en expresiones para las entidades de la capa de destino tocadas por la entidad actual.

Obtenga más información sobre el predicado subyacente «Toques» de GEOS, como se describe en la función PostGIS. ST_Touches.

Sintaxis	overlay_touches(layer, [expression], [filter], [limit], [cache=false]) [] marca argumentos opcionales
Argumentos	• **capa** - la capa cuya superposición está marcada • **expresión** - una expresión opcional para evaluar las características de la capa de destino. Si no se establece, la función simplemente devolverá un booleano que indica si hay al menos una coincidencia. • **filtrar** - una expresión opcional para filtrar los objetos de destino a verificar. Si no se establece, se marcarán todos los objetos. • **límite** - un entero opcional para limitar el número de objetos elegidos. Si no se establece, todos los objetos elegidos serán devueltos. • **cache** - establezca esto en verdadero para construir un índice espacial local (la mayoría de las veces, esto no es deseado, a menos que esté trabajando con un proveedor de datos particularmente lento)
Ejemplos	• `overlay_touches('regions')` → TRUE si el objeto espacial actual toca espacialmente una región. • `overlay_touches('regions', filter:= population > 10000)` → TRUE si el objeto espacial actual toca espacialmente una región con una población superior a 10000. • `overlay_touches('regions', name)` → una serie de nombres, para las regiones tocadas por la entidad actual • `array_to_string(overlay_touches('regions', name))` → a string as a comma separated list of names, for the regions touched by the current feature • `array_sort(overlay_touches(layer:='regions', expression:="name", filter:= population > 10000))` → una matriz ordenada de nombres, para las regiones afectadas por la entidad actual y con una población superior a 10000 • `overlay_touches(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → una matriz de geometrías (en WKT), para un máximo de dos regiones tocadas por el objeto espacial actual.

Otras lecturas: *touches*, *manipulación array*, algoritmo *Seleccionar por ubicacion*

overlay_within

Devuelve si la entidad actual está espacialmente dentro de al menos una entidad de una capa de destino o una matriz de resultados basados en expresiones para las entidades de la capa de destino que contienen la entidad actual.

Obtenga más información sobre el predicado subyacente «Dentro» de GEOS, como se describe en la función PostGIS ST_Within.

Sintaxis	overlay_within(layer, [expression], [filter], [limit], [cache=false]) [] marca argumentos opcionales
Argumentos	• **capa** - la capa cuya superposición está marcada • **expresión** - una expresión opcional para evaluar las características de la capa de destino. Si no se establece, la función simplemente devolverá un booleano que indica si hay al menos una coincidencia. • **filtrar** - una expresión opcional para filtrar los objetos de destino a verificar. Si no se establece, se marcarán todos los objetos. • **límite** - un entero opcional para limitar el número de objetos elegidos. Si no se establece, todos los objetos elegidos serán devueltos. • **cache** - establezca esto en verdadero para construir un índice espacial local (la mayoría de las veces, esto no es deseado, a menos que esté trabajando con un proveedor de datos particularmente lento)
Ejemplos	• `overlay_within('regions')` → TRUE si el objeto espacial actual se encuentra espacialmente dentro de una región. • `overlay_within('regions', filter:= population > 10000)` → TRUE si el objeto espacial actual se encuentra espacialmente dentro de una región con una población superior a 10000. • `overlay_within('regions', name)` → una matriz de nombres, para las regiones que contienen la entidad actual • `array_to_string(overlay_within('regions', name))` → una cadena como una lista de nombres separados por comas, para las regiones que contienen la entidad actual • `array_sort(overlay_within(layer:='regions', expression:="name", filter:= population > 10000))` → una matriz ordenada de nombres, para las regiones que contienen la entidad actual y con una población superior a 10000 • `overlay_within(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → una matriz de geometrías (en WKT), para un máximo de dos regiones que contengan el objeto espacial actual.

Otras lecturas: *within*, *manipulación array*, algoritmo *Seleccionar por ubicacion*

$perimeter

Devuelve la longitud del perímetro de la entidad actual. El perímetro calculado por esta función respeta tanto la configuración del elipsoide del proyecto actual como la configuración de la unidad de distancia. Por ejemplo, si se ha establecido un elipsoide para el proyecto, el perímetro calculado será elipsoidal, y si no se establece ningún elipsoide, el perímetro calculado será planimétrico.

Sintaxis	$perimeter
Ejemplos	• `$perimeter` → 42

perimeter

Devuelve el perímetro de un objeto poligonal de geometría. Los cálculos son siempre planimétricos en el Sistema de referencia espacial (SRS) de esta geometría, y las unidades del perímetro devuelto coincidirán con las unidades del SRS. Esto difiere de los cálculos realizados por la función $perimeter, que realizará cálculos elipsoidales basados en la configuración de unidad de distancia y elipsoide del proyecto.

Sintaxis	perimeter(geometry)
Argumentos	• **geometry** - objeto poligonal de geometría
Ejemplos	• `perimeter(geom_from_wkt('POLYGON((0 0, 4 0, 4 2, 0 2, 0 0))'))` → 12.0

point_n

Devuelve un nodo especiífico de una geometría

Sintaxis	point_n(geometry, index)
Argumentos	• **geometry** - objeto geometría • **index** - índice del nodo a devolver, donde 1 es el primer nodo; si el valor es negativo, el índice de vértice seleccionado será su recuento total menos el valor absoluto
Ejemplos	• `geom_to_wkt(point_n(geom_from_wkt('POLYGON((0 0, 4 0, 4 2, 0 2, 0 0))'),2))` → "Point (4 0)"

Otras lecturas: algoritmo *Extráe vértices específicos*

point_on_surface

Devuelve un punto que se garantiza que se encuentra en la superficie de una geometría.

Sintaxis	point_on_surface(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `point_on_surface(@geometry)` → una geometría de puntos.

Otras lecturas: algoritmo *Punto en la superficie*

pole_of_inaccessibility

Calcula el polo de inaccesibilidad aproximado de una superficie, que es el punto interno más distante del límite de la superficie. Esta función utiliza el algoritmo "polylabel" (Vladimir Agafonkin, 2016), que es un enfoque iterativo garantizado para encontrar el verdadero polo de inaccesibilidad dentro de una tolerancia especificada. Las tolerancias más precisas requieren más iteraciones y tomará más tiempo calcularlas.

Sintaxis	pole_of_inaccessibility(geometry, tolerance)
Argumentos	• **geometry** - una geometría • **tolerance** - distancia máxima entre el punto devuelto y la verdadera localización del polo
Ejemplos	• `geom_to_wkt(pole_of_inaccessibility( geom_from_wkt('POLYGON((0 1, 0 9, 3 10, 3 3, 10 3, 10 1, 0 1))'), 0.1))` → "Point(1.546875 2.546875)"



Figura 9.26: Polo de inaccesibilidad

Otras lecturas: algoritmo *Polo de inaccesibilidad*

project

Devuelve un punto proyextado desde un punto inicial usando una distancia, una dirección (azimut) y una elevación en radianes.

Sintaxis	project(point, distance, azimuth, [elevation]) [] marca argumentos opcionales
Argumentos	• **point** - punto inicial • **distance** - distancia a proyecto • **azimuth** - azimut en radianes horarios, donde 0 corresponde al norte • **elevation** - ángulo de inclinación en radianes
Ejemplos	• `geom_to_wkt(project(make_point(1, 2), 3, radians(270)))` → "Point(-2, 2)"

Otras lecturas: algoritmo *Proyección de puntos (Cartesiano)*

relate

Prueba la representación del modelo dimensional extendido de 9 intersecciones (DE-9IM) de la relación entre dos geometrías.

Variante de relación

Devuelve la representación del modelo dimensional extendido de 9 intersecciones (DE-9IM) de la relación entre dos geometrías.

Sintaxis	relate(geometry, geometry)
Argumentos	• **geometry** - una geometría • **geometry** - una geometría
Ejemplos	• `relate( geom_from_wkt( 'LINESTRING(40 40,120 120)' ), geom_from_wkt( 'LINESTRING(40 40,60 120)' ) )` → "FF1F00102"

Variante de patrón coincidente

Comprueba si la relación DE-9IM entre dos geometrías coincide con un patrón especificado.

Sintaxis	relate(geometry, geometry, pattern)
Argumentos	• **geometry** - una geometría • **geometry** - una geometría • **pattern** - DE-9IM patrón a coincidir
Ejemplos	• `relate( geom_from_wkt( 'LINESTRING(40 40,120 120)' ), geom_from_wkt( 'LINESTRING(40 40,60 120)' ), '**1F001**' )` → TRUE

reverse

Reverses the direction of a line string or reverses a string of text.

Variante de cadena

Reverses the order of characters in a string.

Sintaxis	reverse(string)
Argumentos	• **string** - string to reverse
Ejemplos	• `reverse('hello')` → "olleh"

Variante Geometría

Invierte la dirección de una cadena lineal invirtiendo el orden de sus vértices.

Sintaxis	reverse(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `geom_to_wkt(reverse(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)')))` → "LINESTRING(2 2, 1 1, 0 0)"

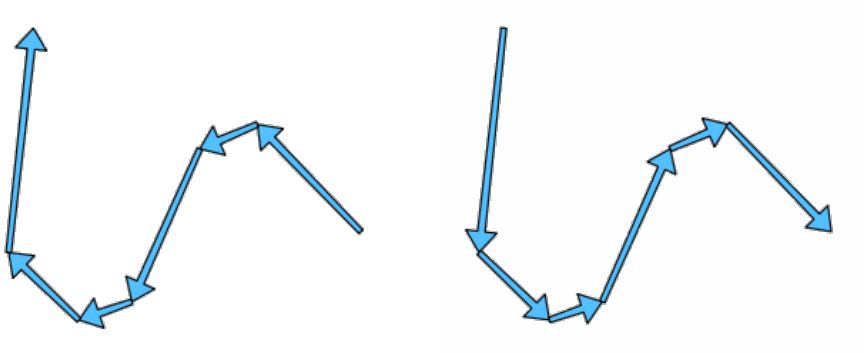


Figura 9.27: Inversión del sentido de la línea

Otras lecturas: algoritmo *Línea de dirección reversible*

rotate

Devuelve una versión rotada de una geometría. Los cálculos son en el SRS de esta geometría.

Sintaxis	rotate(geometry, rotation, [center=NULL], [per_part=false]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **rotation** - rotación en sentido horario en grados • **center** - punto de centro de rotación. Si no se especifica, se usará el centro del recuadro delimitador de la geometría. • **per_part** - aplica la rotación por pieza. Si es verdadero, la rotación se aplicará alrededor del centro del cuadro delimitador de cada pieza cuando la geometría de entrada sea multiparte y no se especifique un punto central de rotación explícito.
Ejemplos	• `rotate(@geometry, 45, make_point(4, 5))` → geometría girada 45 grados en el sentido de las agujas del reloj alrededor del punto (4, 5). • `rotate(@geometry, 45)` → geometría girada 45 grados en el sentido de las agujas del reloj alrededor del centro de su caja delimitadora.

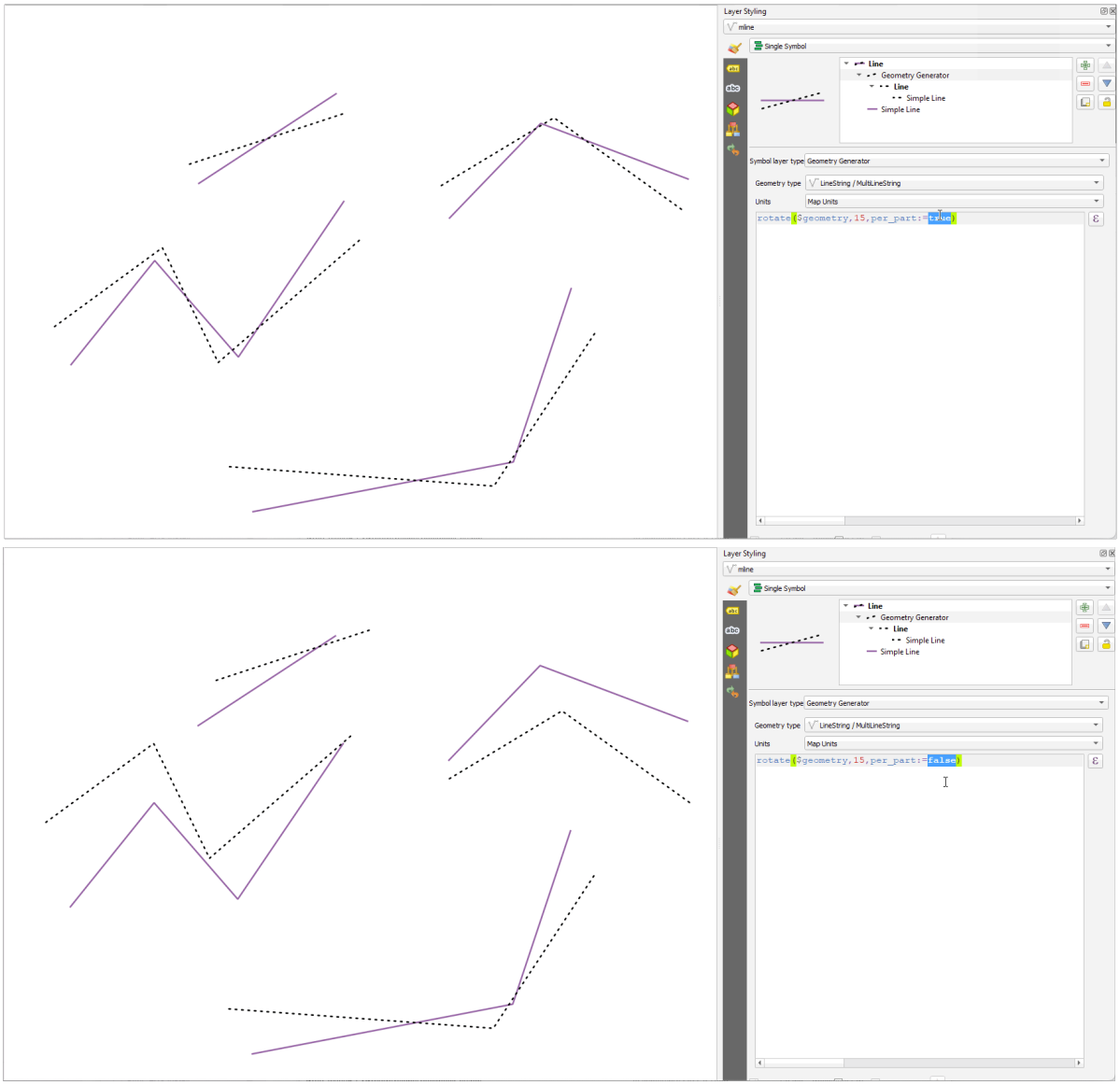


Figura 9.28: Características giratorias

roundness

Calcula el grado de similitud entre una forma poligonal y un círculo. La función devuelve 1 si la forma poligonal es un círculo perfecto y 0 si es completamente plana.

Sintaxis	roundness(geometry)
Argumentos	• **geometría** - un polígono
Ejemplos	• `round(roundness(geom_from_wkt('POLYGON(( 0 0, 0 1, 1 1, 1 0, 0 0))')), 3)` → 0.785 • `round(roundness(geom_from_wkt('POLYGON(( 0 0, 0 0.1, 1 0.1, 1 0, 0 0))')), 3)` → 0.260

Más información: *Redondez* algoritmo

scale

Devuelve una versión a escala de una geometría. Los cálculos se realizan en el Sistema de Referencia Espacial de esta geometría.

Sintaxis	scale(geometry, x_scale, y_scale, [center]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **escala_x** - factor de escala del eje x • **escala_y** - factor de escala del eje y • **centro** - punto central de la escala. Si no se especifica, se utiliza el centro de la caja delimitadora de la geometría.
Ejemplos	• `scale(@geometry, 2, 0.5, make_point(4, 5))` → geometría escalada dos veces horizontalmente y reducida a la mitad verticalmente, alrededor del punto (4, 5). • `scale(@geometry, 2, 0.5)` → geometría dos veces horizontal y dos veces vertical, alrededor del centro de su cuadro delimitador.

segments_to_lines

Devuelve una geometría multilínea que consta de una línea para todos los segmentos en la geoemtría entrante.

Sintaxis	segments_to_lines(geometry)
Argumentos	• **geometry** - objeto geometría
Ejemplos	• `geom_to_wkt(segments_to_lines(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)')))` → "MultiLineString ((0 0, 1 1),(1 1, 2 2))"

Otras lecturas: algoritmo *Explotar lineas*

shared_paths

Devuelve una colección que contiene las trayectorias compartidas por las dos geometrías de entrada. Los que van en la misma dirección están en el primer elemento de la colección, los que van en la dirección opuesta están en el segundo elemento. Las trayectorias se indican en la dirección de la primera geometría.

Sintaxis	shared_paths(geometry1, geometry2)
Argumentos	• **geometría1** - una geometría LineString/MultiLineString • **geometría2** - una geometría LineString/MultiLineString
Ejemplos	• `geom_to_wkt(shared_paths(geom_from_wkt('MULTILINESTRING((26 125,26 200,126 200,126 125,26 125),(51 150,101 150,76 175,51 150))'),geom_from_wkt('LINESTRING(151 100,126 156.25,126 125,90 161, 76 175)')))` → "GeometryCollection (MultiLineString ((126 156.25, 126 125),(101 150, 90 161),(90 161, 76 175)),MultiLineString EMPTY)" • `geom_to_wkt(shared_paths(geom_from_wkt('LINESTRING(76 175,90 161,126 125,126 156.25,151 100)'), geom_from_wkt('MULTILINESTRING((26 125,26 200,126 200, 126 125,26 125),(51 150,101 150,76 175,51 150))')))` → "GeometryCollection (MultiLineString EMPTY,MultiLineString ((76 175, 90 161),(90 161, 101 150),(126 125, 126 156.25)))"

shortest_line

Devuelve la línea mas corta uniendo geometria1 a geometría2. La línea resultante empezará en geometría1 y finalizará en geometría2.

Sintaxis	shortest_line(geometry1, geometry2)
Argumentos	• **geometry1** - geometría origen para encontrar la línea mas corta • **geometry2** - geometría destino para encontrar la línea más corta
Ejemplos	• `geom_to_wkt(shortest_line(geom_from_wkt('LINESTRING (20 80, 98 190, 110 180, 50 75 )'),geom_from_wkt('POINT(100 100)')))` → "LineString(73.0769 115.384, 100 100)"

simplify

Simplifica una geometría eliminando nodos mediante un umbral basado en la distancia (es decir, el algoritmo de Douglas Peucker). El algoritmo conserva grandes desviaciones en las geometrías y reduce el número de vértices en segmentos casi rectos.

Sintaxis	simplify(geometry, tolerance)
Argumentos	• **geometry** - una geometría • **tolerance** - desviación máxima de los segmentos rectos para los puntos a eliminar
Ejemplos	• `geom_to_wkt(simplify(geometry:=geom_from_wkt('LineString(0 0, 5 0.1, 10 0)'),tolerance:=5))` → "LineString(0 0, 10 0)"



Figura 9.29: De izquierda a derecha, capa de origen y tolerancias de simplificación crecientes

Otras lecturas: algoritmo *Simplificar*

simplify_vw

Simplifica una geometría eliminando nodos utilizando un umbral basado en áreas (es decir, el algoritmo Visvalingam-Whyatt). El algoritmo elimina los vértices que crean áreas pequeñas en geometrías, por ejemplo, picos estrechos o segmentos casi rectos.

Sintaxis	simplify_vw(geometry, tolerance)
Argumentos	• **geometry** - una geometría • **tolerance** - una medida del área máxima creada por un nodo para que el nodo sea eliminado
Ejemplos	• `geom_to_wkt(simplify_vw(geometry:=geom_from_wkt('LineString(0 0, 5 0, 5.01 10, 5.02 0, 10 0)'),tolerance:=5))` → "LineString(0 0, 10 0)"

Otras lecturas: algoritmo *Simplificar*

single_sided_buffer

Devuelve una geometría formada al almacenar en búfer solo un lado de una geometría de cadena lineal. Las distancias están en el sistema de referencia espacial de esta geometría.

Sintaxis	single_sided_buffer(geometry, distance, [segments=8], [join=1], [miter_limit=2.0]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría de cadena lineal(múltiple) • **distance** - distancia de amortiguación. Los valores positivos se almacenarán en búfer a la izquierda de las líneas, los valores negativos a la derecha • **segments** - número de segmentos que se utilizarán para representar un cuarto de círculo cuando se utilice un estilo de unión redonda. Un número mayor da como resultado un búfer más suave con más nodos. • **join** - join style for corners, where 1 = round, 2 = miter and 3 = bevel • **miter_limit** - límite en la relación de inglete que se usa para esquinas muy afiladas (cuando se usan juntas de inglete solamente)
Ejemplos	• `single_sided_buffer(@geometry, 10.5)` → línea con zona de influencia a la izquierda en 10,5 unidades. • `single_sided_buffer(@geometry, -10.5)` → línea con zona de influencia a la derecha en 10,5 unidades. • `single_sided_buffer(@geometry, 10.5, segments:=16, join:=1)` → línea con zona de influencia a la izquierda en 10.5 unidades, usando más segmentos para resultar en una zona de influencia más suave. • `single_sided_buffer(@geometry, 10.5, join:=3)` → línea con zona de influencia a la izquierda en 10.5 unidades, usando una unión biselada.



Figura 9.30: Izquierda versus lado derecho en la misma capa vectorial lineal

Otras lecturas: algoritmo *Buffer a un solo lado*

sinuosity

Devuelve la sinuosidad de una curva, que es la relación entre la longitud de la curva y la distancia recta (2D) entre sus extremos.

Sintaxis	sinuosity(geometry)
Argumentos	• **geometría** - Curva de entrada (cadena circular, cadena lineal)
Ejemplos	• `round(sinuosity(geom_from_wkt('LINESTRING(2 0, 2 2, 3 2, 3 3')), 3)` → 1.265 • `sinuosity(geom_from_wkt('LINESTRING( 3 1, 5 1)'))` → 1.0

smooth

Suaviza una geometría agregando nodos adicionales que redondean las esquinas de la geometría. Si las geometrías de entrada contienen valores Z o M, estos también se suavizarán y la geometría de salida conservará la misma dimensionalidad que la geometría de entrada.

Sintaxis	smooth(geometry, [iterations=1], [offset=0.25], [min_length=-1], [max_angle=180]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **iterations** - número de iteraciones de suavizado para aplicar. Los números más grandes dan como resultado geometrías más suaves pero más complejas. • **offset** - valor entre 0 y 0,5 que controla la precisión con la que la geometría suavizada sigue la geometría original. Los valores más pequeños dan como resultado un suavizado más estricto, los valores más grandes dan como resultado un suavizado más suelto. • **min_length** - longitud mínima de los segmentos a los que aplicar suavizado. Este parámetro se puede utilizar para evitar colocar excesivos nodos adicionales en segmentos más cortos de la geometría. • **max_angle** - ángulo máximo en el nodo para aplicar el suavizado (0-180). Al reducir el ángulo máximo de forma intencionada, se pueden conservar las esquinas afiladas de la geometría. Por ejemplo, un valor de 80 grados conservará ángulos rectos en la geometría.
Ejemplos	• `geom_to_wkt(smooth(geometry:=geom_from_wkt('LineString(0 0, 5 0, 5 5)'),iterations:=1,offset:=0.2,min_length:=-1, max_angle:=180))` → "LineString (0 0, 4 0, 5 1, 5 5)"

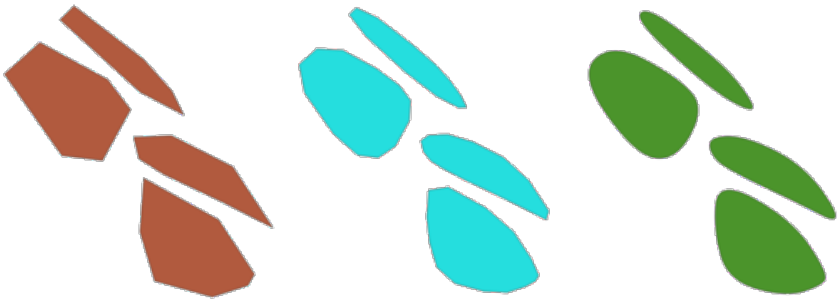


Figura 9.31: Aumentando el número de iteraciones provoca geometrías mas suaves

Otras lecturas: algoritmo *Suavizar*

square_wave

Construye ondas cuadradas/rectangulares a lo largo de la frontera de una geometría.

Sintaxis	square_wave(geometry, wavelength, amplitude, [strict=False]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **wavelength** - longitud de onda de la forma de onda cuadrada • **amplitude** - amplitud de la forma de onda cuadrada • **strict** - De forma predeterminada, el argumento longitud de onda se trata como una «longitud de onda máxima», donde la longitud de onda real se ajustará dinámicamente para que se cree un número exacto de ondas cuadradas a lo largo de los límites de la geometría. Si el argumento estricto se establece como verdadero, entonces la longitud de onda se utilizará exactamente y se podrá utilizar un patrón incompleto para la forma de onda final.
Ejemplos	• `square_wave(geom_from_wkt('LineString(0 0, 10 0)'), 3, 1)` → Ondas cuadradas con longitud de onda 3 y amplitud 1 a lo largo de la linetring.

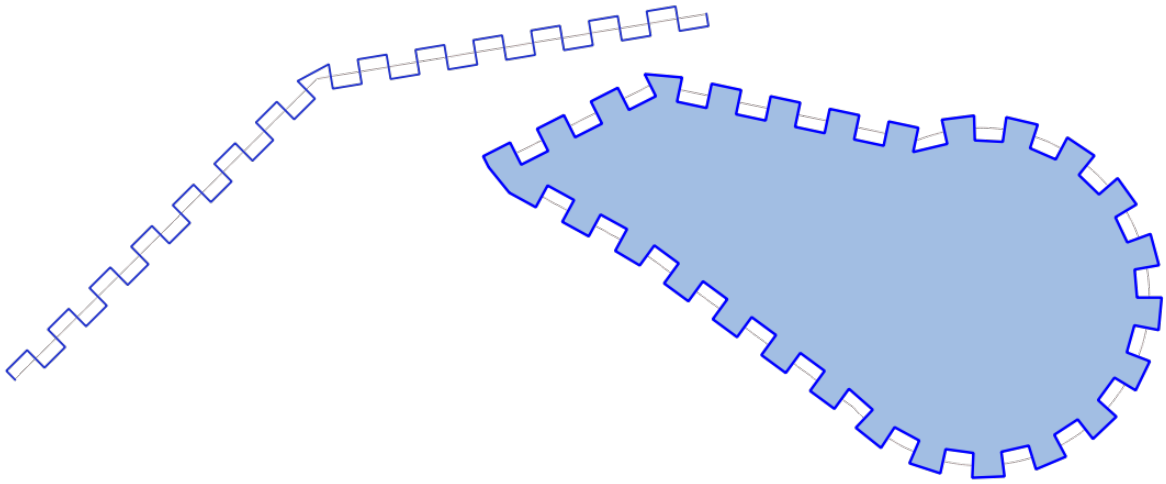


Figura 9.32: Simbolización de objetos espaciales con ondas cuadradas

square_wave_randomized

Construye ondas cuadradas/rectangulares aleatorias a lo largo de la frontera de una geometría.

Sintaxis	square_wave_randomized(geometry, min_wavelength, max_wavelength, min_amplitude, max_amplitude, [seed=0]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **min_wavelength** - longitud de onda mínima de las ondas • **max_wavelength** - longitud de onda máxima de las ondas • **min_amplitude** - amplitud mínima de las ondas • **max_amplitude** - amplitud máxima de las ondas • **seed** - especifica una semilla aleatoria para generar ondas. Si la semilla es 0, se generará un conjunto de ondas completamente aleatorio.
Ejemplos	• `square_wave_randomized(geom_from_wkt('LineString(0 0, 10 0)'), 2, 3, 0.1, 0.2)` → Ondas cuadradas de tamaño aleatorio con longitudes de onda entre 2 y 3 y amplitudes entre 0.1 y 0.2 a lo largo de la linetring.

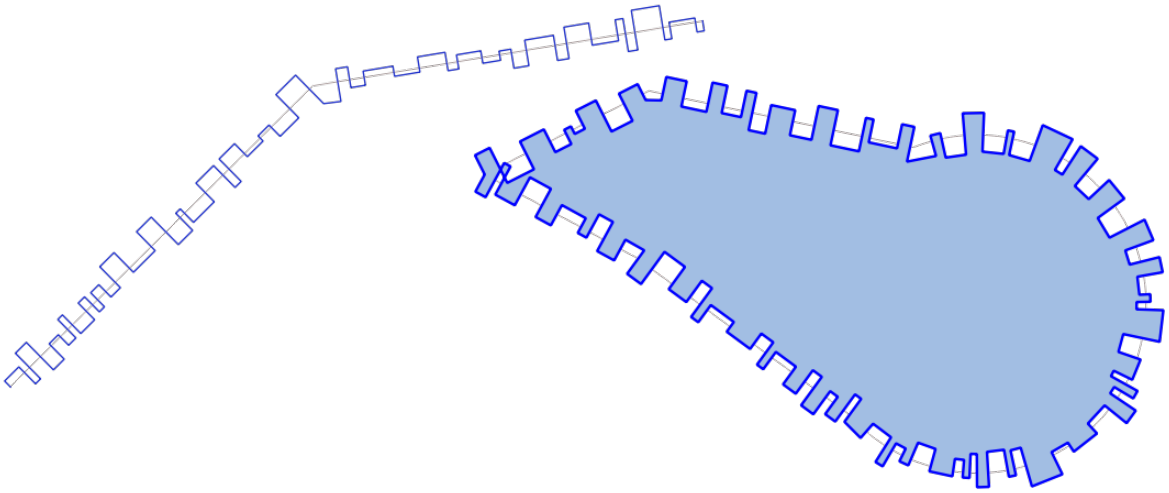


Figura 9.33: Simbolización de objeto espaciales con ondas cuadradas aleatorias

start_point

Devuelve el primer nodo de una geoemtría.

Sintaxis	start_point(geometry)
Argumentos	• **geometry** - objeto geometría
Ejemplos	• `geom_to_wkt(start_point(geom_from_wkt('LINESTRING(4 0, 4 2, 0 2)')))` → “Point (4 0)”

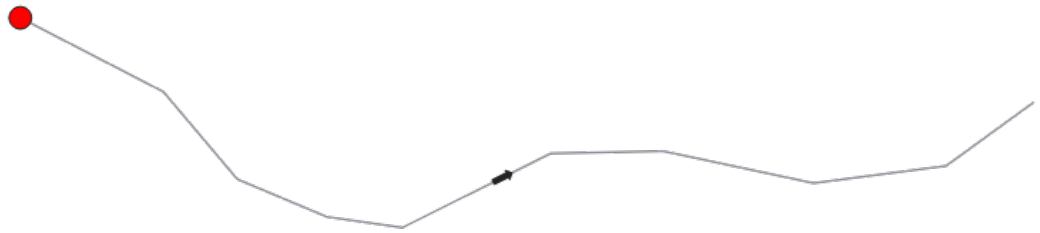


Figura 9.34: Punto de partida de un objeto espacial línea

Lectura adicional: *end_point*, algoritmo :ref:`qgisextractspecificvertices

straight_distance_2d

Devuelve la distancia directa/euclidiana entre el primer y el último vértice de una geometría. La geometría debe ser una curva (cadena circular, cadena lineal).

Sintaxis	straight_distance_2d(geometry)
Argumentos	• **geometría** - La geometría.
Ejemplos	• `straight_distance_2d(geom_from_wkt('LINESTRING(1 0, 1 1)'))` → 1 • `round(straight_distance_2d(geom_from_wkt('LINESTRING(1 4, 3 5, 5 0)')), 3)` → 5.657

Otras lecturas: *length*

sym_difference

Devuelve una geometría que representa las partes de dos geometrías que no se intersecan.

Sintaxis	sym_difference(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `geom_to_wkt( sym_difference( geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ), geom_from_wkt( 'LINESTRING(3 3, 8 8)' ) ) )` → “LINESTRING(5 5, 8 8)”

Otras lecturas: algoritmo *Diferencia simétrica*

tapered_buffer

Crea una zona de influencia a lo largo de una geometría de línea donde el diámetro de la zona de influencia varía uniformemente a lo largo de la línea.

Sintaxis	tapered_buffer(geometry, start_width, end_width, [segments=8]) [] marca argumentos opcionales
Argumentos	• **geometry** - geometría entrante. Debe ser una geometría (multi)lineal. • **start_width** - anchura del bufer al inicio de línea. • **end_width** - anchura del búfer al final de su línea. • **segments** - número de segmentos para aproximar las curvas de un cuarto de círculo en el búfer.
Ejemplos	• `tapered_buffer(geometry:=geom_from_wkt('LINESTRING(1 2, 4 2)'),start_width:=1,end_width:=2,segments:=8)` → Un búfer afilado que comienza con un diámetro de 1 y termina con un diámetro de 2 a lo largo de la geometría de la cadena lineal.

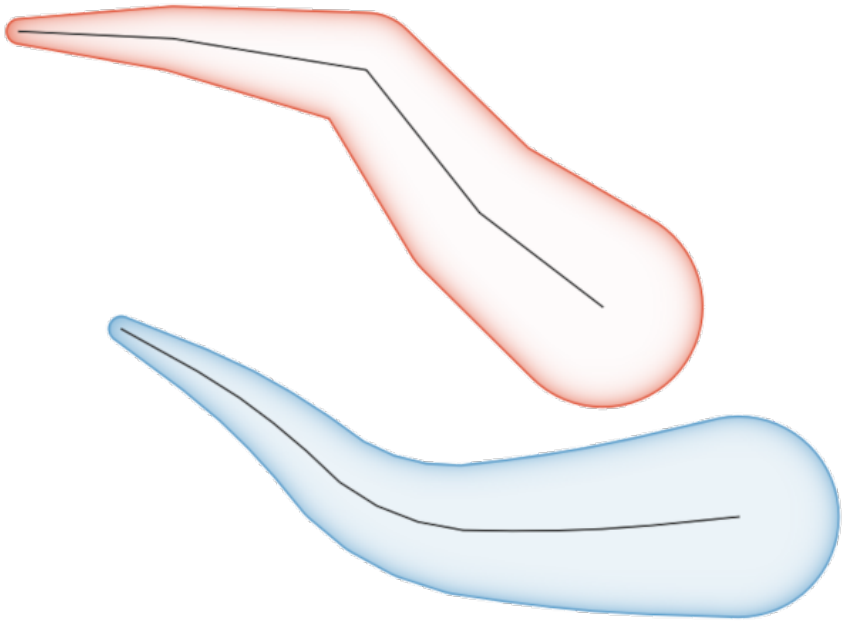


Figura 9.35: Buffer cónico en características de línea

Otras lecturas: algoritmo *Zonas de influencia estrechas*

touches

Comprueba si una geometría toca a otra. Devuelve TRUE si las geometrías tienen al menos un punto en común, pero sus interiores no se cruzan.

Sintaxis	touches(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `touches( geom_from_wkt( 'LINESTRING(5 3, 4 4)' ), geom_from_wkt( 'LINESTRING(3 3, 4 4, 5 5)' ) )` → TRUE • `touches( geom_from_wkt( 'POINT(4 4)' ), geom_from_wkt( 'POINT(5 5)' ) )` → FALSE

Otras lecturas: *overlay_touches*

transform

Devuelve la geometría transformada de un CRS de origen a un CRS de destino.

Sintaxis	transform(geometry, source_auth_id, dest_auth_id)
Argumentos	• **geometry** - una geometría • **source_auth_id** - the source CRS definition or CRS object • **dest_auth_id** - the destination CRS definition or CRS object
Ejemplos	• `geom_to_wkt( transform( make_point(488995.53240249, 7104473.38600835), 'EPSG:2154', 'EPSG:4326' ) )` → "POINT(0 51)"

Otras lecturas: algoritmo *Capa reproyectada*

translate

Devuelve una versión traducida de una geometría. Los cálculos están en el Sistema de referencia espacial de esta geometría.

Sintaxis	translate(geometry, dx, dy)
Argumentos	• **geometry** - una geometría • **dx** - delta x • **dy** - delta y
Ejemplos	• `translate(@geometry, 5, 10)` → una geometría del mismo tipo que la original.

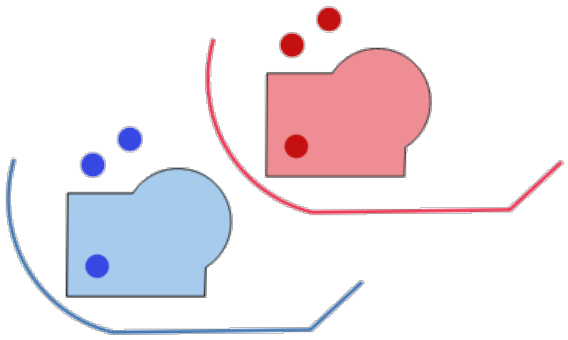


Figura 9.36: Características de traducción

Otras lecturas: algoritmo *Traslado*

triangular_wave

Construye ondas triangulares a lo largo de la frontera de una geometría.

Sintaxis	triangular_wave(geometry, wavelength, amplitude, [strict=False]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **wavelength** - longitud de onda de la forma de onda triangular • **amplitude** - amplitud de la forma de onda triangular • **strict** - De forma predeterminada, el argumento longitud de onda se trata como una «longitud de onda máxima», donde la longitud de onda real se ajustará dinámicamente para que se cree un número exacto de ondas triangulares a lo largo de los límites de la geometría. Si el argumento estricto se establece como verdadero, entonces la longitud de onda se utilizará exactamente y se podrá utilizar un patrón incompleto para la forma de onda final.
Ejemplos	• `onda_triangular(geom_from_wkt('Lineacadena(0 0, 10 0)'), 3, 1)` → Ondas triangulares con longitud de onda 3 y amplitud 1 a lo largo de la linestring.

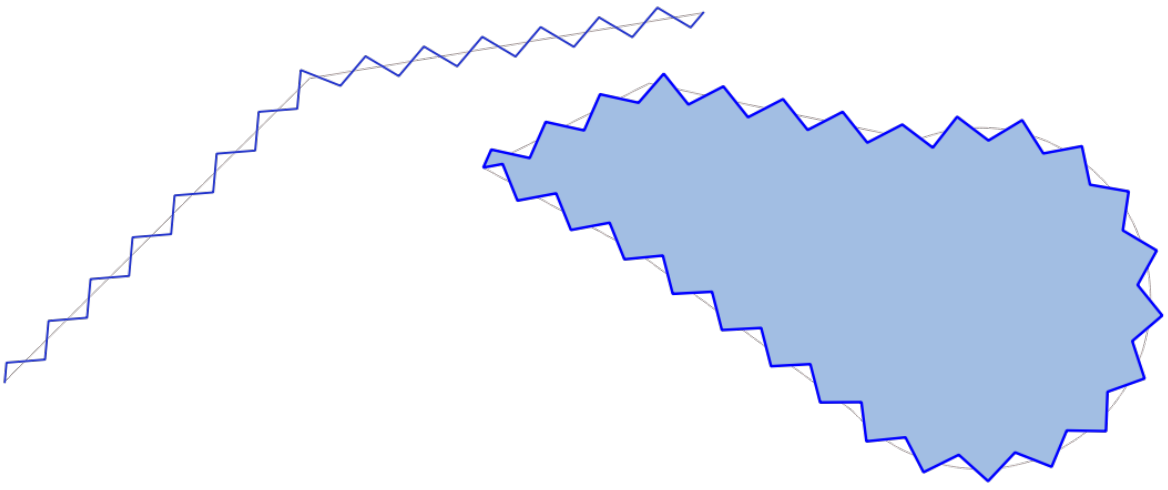


Figura 9.37: Simbolización de objetos espaciales con ondas triangulares

triangular_wave_randomized

Construye ondas triangulares aleatorias a lo largo de la frontera de una geometría.

Sintaxis	triangular_wave_randomized(geometry, min_wavelength, max_wavelength, min_amplitude, max_amplitude, [seed=0]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **min_wavelength** - longitud de onda mínima de las ondas • **max_wavelength** - longitud de onda máxima de las ondas • **min_amplitude** - amplitud mínima de las ondas • **max_amplitude** - amplitud máxima de las ondas • **seed** - especifica una semilla aleatoria para generar ondas. Si la semilla es 0, se generará un conjunto de ondas completamente aleatorio.
Ejemplos	• `onda_triangular_randomized(geom_from_wkt('LineString(0 0, 10 0)'), 2, 3, 0.1, 0.2)` → Ondas triangulares de tamaño aleatorio con longitudes de onda entre 2 y 3 y amplitudes entre 0.1 y 0.2 a lo largo de la linetring.

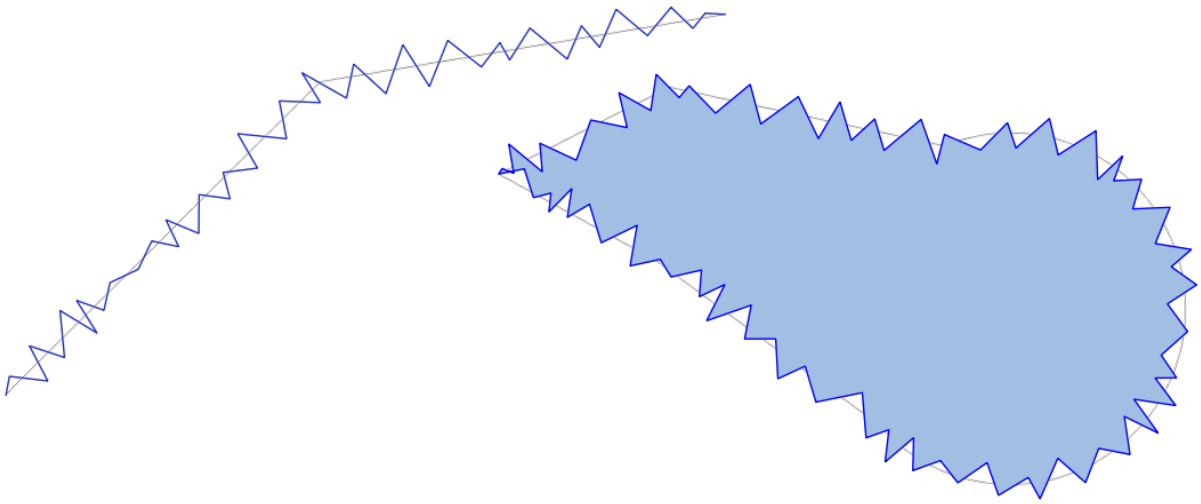


Figura 9.38: Simbolización de objetos espaciales con ondas triangulares aleatorias

union

Devuelve una geometría que representa la unión de conjuntos de puntos de las geometrías.

Sintaxis	union(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `geom_to_wkt( union( make_point(4, 4), make_point(5, 5) ) )` → “MULTIPOINT(4 4, 5 5)”

wave

Construye ondas redondeadas (sinusoidales) a lo largo del límite de una geometría.

Sintaxis	wave(geometry, wavelength, amplitude, [strict=False]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **wavelength** - longitud de onda de la onda sinusoidal • **amplitude** - amplitud de la onda sinusoidales • **strict** - De forma predeterminada, el argumento longitud de onda se trata como una «longitud de onda máxima», donde la longitud de onda real se ajustará dinámicamente para que se cree un número exacto de ondas a lo largo de los límites de la geometría. Si el argumento estricto se establece como verdadero, entonces la longitud de onda se utilizará exactamente y se podrá utilizar un patrón incompleto para la forma de onda final.
Ejemplos	• `wave(geom_from_wkt('LineString(0 0, 10 0)'), 3, 1)` → Ondas sinusoidales con longitud de onda 3 y amplitud 1 a lo largo de la linestring.

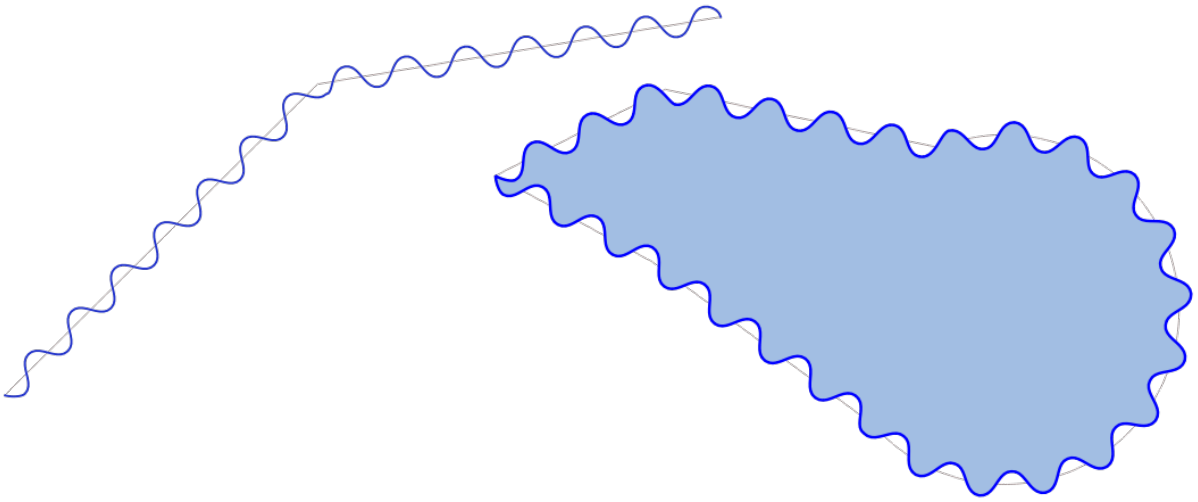


Figura 9.39: Simbolizar objetos espaciales con ondas

wave_randomized

Construye ondas curvas aleatorias (sinusoidales) a lo largo de la frontera de una geometría.

Sintaxis	wave_randomized(geometry, min_wavelength, max_wavelength, min_amplitude, max_amplitude, [seed=0]) [] marca argumentos opcionales
Argumentos	• **geometry** - una geometría • **min_wavelength** - longitud de onda mínima de las ondas • **max_wavelength** - longitud de onda máxima de las ondas • **min_amplitude** - amplitud mínima de las ondas • **max_amplitude** - amplitud máxima de las ondas • **seed** - especifica una semilla aleatoria para generar ondas. Si la semilla es 0, se generará un conjunto de ondas completamente aleatorio.
Ejemplos	• `wave_randomized(geom_from_wkt('LineString(0 0, 10 0)'), 2, 3, 0.1, 0.2)` → Ondas curvas de tamaño aleatorio con longitudes de onda entre 2 y 3 y amplitudes entre 0.1 y 0.2 a lo largo de la linestring.

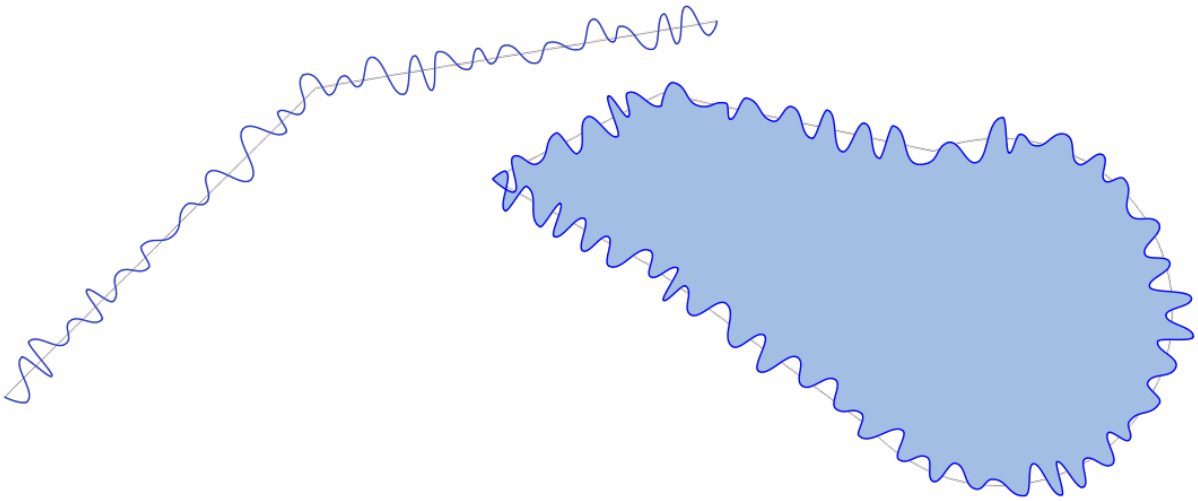


Figura 9.40: Simbolización de objetos espaciales con ondas aleatorias

wedge_buffer

Devuelve una zona de influencia en forma de cuña que se origina en una geometría de puntos.

Sintaxis	wedge_buffer(center, azimuth, width, outer_radius, [inner_radius=0.0]) [] marca argumentos opcionales
Argumentos	• **center** - punto central (origen) del búfer. Debe ser una geometría de puntos. • **azimuth** - ángulo (en grados) a punto para la mitad de la cuña. • **width** - ancho del búfer (en grados). Tenga en cuenta que la cuña se extenderá a la mitad del ancho angular a ambos lados de la dirección del azimut. • **outer_radius** - radio exterior para bufers • **inner_radius** - radio interior opcional para bufers
Ejemplos	• `wedge_buffer(center:=geom_from_wkt('POINT(1 2)'), azimuth:=90,width:=180,outer_radius:=1)` → A wedge shaped buffer centered on the point (1,2), facing to the East, with a width of 180 degrees and outer radius of 1.

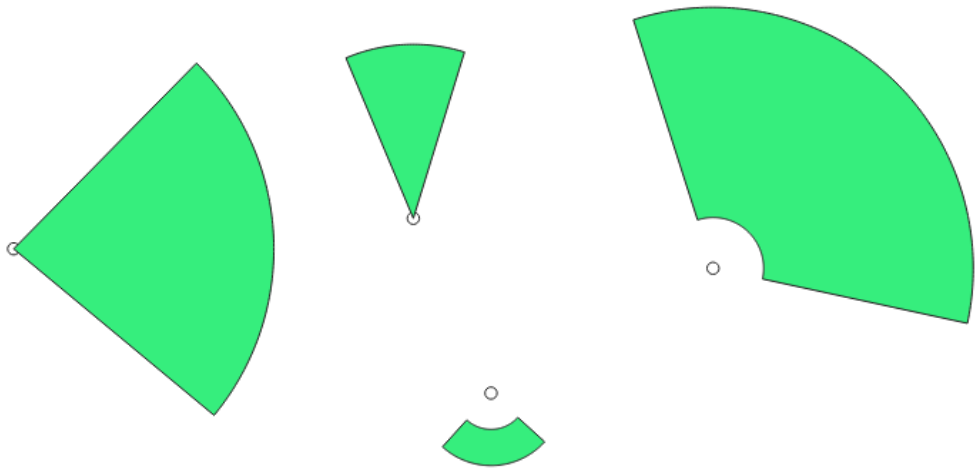


Figura 9.41: Objetos espaciales de amortiguación en cuña

Otras lecturas: algoritmo *Crear buffer de cuñas*

within

Comprueba si una geometría está dentro de otra. Devuelve TRUE si la geometry1 está completamente dentro de la geometry2.

Sintaxis	within(geometry1, geometry2)
Argumentos	• **geometry1** - una geometría • **geometry2** - una geoemtría
Ejemplos	• `within( geom_from_wkt( 'POINT( 0.5 0.5)' ), geom_from_wkt( 'POLYGON((0 0, 0 1, 1 1, 1 0, 0 0))' ) )` → TRUE • `within( geom_from_wkt( 'POINT( 5 5 )' ), geom_from_wkt( 'POLYGON((0 0, 0 1, 1 1, 1 0, 0 0 ))' ) )` → FALSE

Otras lecturas: *overlay_within*

$x

Devuelve la coordenada x del objeto espacial punto actual. Si el objeto espacial es multipunto, se devolverá la coordenada x del primer punto. **AVISO: Esta función está obsoleta. Se recomienda utilizar la función de sustitución x() con la variable @geometry.

Sintaxis	$x
Ejemplos	• `$x` → 42

Lectura adicional: *x*

x

Devuelve la coordenada x de una geometría puntual, o la coordenada x del centroide para una geometría no puntual.

Sintaxis	x(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `x( geom_from_wkt( 'POINT(2 5)' ) )` → 2 • `x( @geometría )` → coordenada x del centroide del objeto espacial actual.

$x_at

Obtiene una coordenada x de la geometría del objeto espacial actual. **AVISO: Esta función está obsoleta. Se recomienda utilizar la función de reemplazo x_at con la variable @geometry en su lugar.

Sintaxis	$x_at(vertex)
Argumentos	• **vértice** - índice del vértice de la geometría actual (los índices comienzan en 0; los valores negativos se aplican a partir del último índice, comenzando en -1)
Ejemplos	• `$x_at(1)` → 5

Lectura adicional: *x_at*

x_at

Recupera una coordenada x de la geometría.

Sintaxis	x_at(geometry, vertex)
Argumentos	• **geometry** - objeto geometría • **vértice** - índice del vértice de la geometría (los índices comienzan en 0; los valores negativos se aplican a partir del último índice, comenzando en -1)
Ejemplos	• `x_at( geom_from_wkt( 'POINT(4 5)' ), 0 )` → 4

x_max

Devuelve la coordenada x máxima de una geometría. Los cálculos están en el sistema de referencia espacial de esta geometría.

Sintaxis	x_max(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `x_max( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 4

x_min

Devuelve la coordenada x mínima de una geometría. Los cálculos están en el sistema de referencia espacial de esta geometría.

Sintaxis	x_min(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `x_min( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 2

$y

Devuelve la coordenada y del objeto espacial punto actual. Si el objeto espacial es multipunto, se devolverá la coordenada y del primer punto. **AVISO: Esta función está obsoleta. Se recomienda utilizar en su lugar la función de sustitución y() con la variable @geometry.**

Sintaxis	$y
Ejemplos	• `$y` → 42

Lectura adicional: *y*

y

Devuelve la coordenada y de una geometría puntual, o la coordenada y del centroide para una geometría no puntual.

Sintaxis	y(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `y( geom_from_wkt( 'POINT(2 5)' ) )` → 5 • `y( @geometry )` → coordenada y del centroide del objeto espacial actual.

$y_at

Obtiene una coordenada y de la geometría del objeto espacial actual. **AVISO: Esta función está obsoleta. Se recomienda utilizar en su lugar la función de sustitución y_at con la variable @geometry.**

Sintaxis	$y_at(vertex)
Argumentos	• **vértice** - índice del vértice de la geometría actual (los índices comienzan en 0; los valores negativos se aplican a partir del último índice, comenzando en -1)
Ejemplos	• `$y_at(1)` → 2

Lectura adicional: *y_at*

y_at

Recupera una coordenada y de la geometría.

Sintaxis	y_at(geometry, vertex)
Argumentos	• **geometry** - objeto geometría • **vértice** - índice del vértice de la geometría (los índices comienzan en 0; los valores negativos se aplican a partir del último índice, comenzando en -1)
Ejemplos	• `y_at( geom_from_wkt( 'POINT(4 5)' ), 0 )` → 5

y_max

Devuelve la coordenada y máxima de una geometría. Los cálculos están en el sistema de referencia espacial de esta geometría.

Sintaxis	y_max(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `y_max( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 8

y_min

Devuelve la coordenada y mínima de una geometría. Los cálculos están en el sistema de referencia espacial de esta geometría.

Sintaxis	y_min(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `y_min( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 5

$z

Devuelve el valor z del objeto espacial punto actual si es 3D. Si el objeto espacial es multipunto, devolverá el valor z del primer punto. **AVISO: Esta función está obsoleta. Se recomienda utilizar la función de sustitución z() con la variable @geometry.

Sintaxis	$z
Ejemplos	• `$z` → 123

z

Devuelve la coordenada z de una geometría de punto, o NULL si la geometría no tiene valor z.

Sintaxis	z(geometry)
Argumentos	• **geometry** - una geometría puntual
Ejemplos	• `z( geom_from_wkt( 'POINTZ(2 5 7)' ) )` → 7

z_at

Recupera una coordenada z de la geometría, o NULL si la geometría no tiene valor z.

Sintaxis	z_at(geometry, vertex)
Argumentos	• **geometry** - objeto geometría • **vértice** - índice del vértice de la geometría (los índices comienzan en 0; los valores negativos se aplican a partir del último índice, comenzando en -1)
Ejemplos	• `z_at(geom_from_wkt('LineStringZ(0 0 0, 10 10 5, 10 10 0)'), 1)` → 5

z_max

Devuelve la coordenada z máxima de una geometría, o NULL si la geometría no tiene valor z.

Sintaxis	z_max(geometry)
Argumentos	• **geometry** - una geometría con coordenada z
Ejemplos	• `z_max( geom_from_wkt( 'POINT ( 0 0 1 )' ) )` → 1 • `z_max( geom_from_wkt( 'MULTIPOINT ( 0 0 1 , 1 1 3 )' ) )` → 3 • `z_max( make_line( make_point( 0,0,0 ), make_point( -1,-1,-2 ) ) )` → 0 • `z_max( geom_from_wkt( 'LINESTRING( 0 0 0, 1 0 2, 1 1 -1 )' ) )` → 2 • `z_max( geom_from_wkt( 'POINT ( 0 0 )' ) )` → NULL

z_min

Devuelve la coordenada z mínima de una geometría, o NULL si la geometría no tiene valor z.

Sintaxis	z_min(geometry)
Argumentos	• **geometry** - una geometría con coordenada z
Ejemplos	• `z_min( geom_from_wkt( 'POINT ( 0 0 1 )' ) )` → 1 • `z_min( geom_from_wkt( 'MULTIPOINT ( 0 0 1 , 1 1 3 )' ) )` → 1 • `z_min( make_line( make_point( 0,0,0 ), make_point( -1,-1,-2 ) ) )` → -2 • `z_min( geom_from_wkt( 'LINESTRING( 0 0 0, 1 0 2, 1 1 -1 )' ) )` → -1 • `z_min( geom_from_wkt( 'POINT ( 0 0 )' ) )` → NULL

9.2.15 Funciones de diseño

Este grupo contiene funciones para manipular las propiedades de elementos del diseño de impresión.

item_variables

Devuelve un mapa de variables de un elemento de diseño dentro de este diseño de impresión.

Sintaxis	item_variables(id)
Argumentos	• **id** - ID del elemento de diseño
Ejemplos	• `map_get( item_variables('Map 0'), 'map_scale')` → escala del elemento “Map 0” en el actual diseñador de impresión

Otras lecturas: Lista de *variables* predeterminadas

map_credits

Devuelve una lista de cadenas de crédito (derechos de uso) para las capas mostradas en un diseño, o un elemento de mapa de diseño específico.

Sintaxis	map_credits([id], [include_layer_names=false], [layer_name_separator=": "]) [] marca argumentos opcionales
Argumentos	• **id** - ID del elemento de mapa. Si no se especifica, se utilizarán las capas de todos los mapas de la composición. • **include_layer_names** - Establézcalo en verdadero para incluir los nombres de las capas antes de sus cadenas de crédito • **layer_name_separator** - Cadena para insertar entre los nombres de las capas y sus cadenas de crédito, si include_layer_names es verdadero
Ejemplos	• `array_to_string( map_credits() )` → lista separada por comas de créditos de capa para todas las capas mostradas en todos los elementos de mapa de la presentación, por ejemplo "CC-BY-NC, CC-BY-SA". • `array_to_string( map_credits( 'Main Map' ) )` → lista separada por comas de créditos de capa para las capas que se muestran en la "Main Map" layout item, e.g "CC-BY-NC, CC-BY-SA" • `array_to_string( map_credits( 'Main Map', include_layer_names := true, layer_name_separator := ': ' ) )` → lista separada por comas de los nombres de las capas y sus créditos para las capas que se muestran en la "Main Map" layout item, e.g. "Railway lines: CC-BY-NC, Basemap: CC-BY-SA"

Esta función requiere que se hayan rellenado las *Access metadata properties* de las capas.

9.2.16 Capas de mapa

Este grupo contiene una lista de las capas disponibles en el proyecto actual y, para cada capa, sus campos (almacenados en el conjunto de datos, los virtuales o auxiliares así como las uniones). Con los campos se puede interactuar de la misma forma que se ha mencionado en *Campos y Valores*, excepto que la doble pulsación añadirá el nombre como una cadena (cita simple) a la expresión en lugar de com o una referencia de campo dado que no pertenece a la capa activa. Esto ofrece una manera conveniente de escribir expresiones que se refieran a diferentes capas, como cuando se llevan a cabo consultas *aggregates*, *attribute* o *spatial*.

También proporciona algunas funciones convenientes para manipular capas.

decode_uri

Toma una capa y decodifica el uri del proveedor de datos subyacente. Depende del proveedor de datos, qué datos están disponibles.

Sintaxis	decode_uri(layer, [part]) [] marca argumentos opcionales
Argumentos	• **layer** - La capa para la que se debe decodificar el uri. • **part** - La parte del uri que regresa. Si no se especifica, se devolverá un mapa con todas las partes de uri.
Ejemplos	• `decode_uri(@layer)` → {“layerId”: “0”, “layerName”: “”, “path”: “/home/qgis/shapefile.shp”} • `decode_uri(@layer)` → {“layerId”: NULL, “layerName”: “layer”, “path”: “/home/qgis/geopackage.gpkg”} • `decode_uri(@layer, 'path')` → “C:\my_data\qgis\shape.shp”

layer_property

Devuelve una propiedad de capa coincidente o un valor de metadatos.

Sintaxis	layer_property(layer, property)
Argumentos	• **layer** - una cadena, que representa un nombre capa o ID de capa • **property** - una cadena correspondiente a la propiedad a devolver. Las opciones válidas son: – name: nombre de capa – id: ID de la capa – title: cadena de título de metadatos – abstract: cadena abstracta de metadatos – keywords: palabras clave de los metadatos – data_url: URL de metadatos – attribution: cadena de atribución de metadatos – attribution_url: URL de atribución de metadatos – source: capa fuente – min_scale: escala de visualización mínima para la capa – max_scale: escala de visualización máxima para la capa – is_editable: si la capa está en modo edición – crs: SRC de la capa – crs_definition: definición completa SRC de la capa – crs_description: descripción SRC de capa – crs_ellipsoid: acrónimo del elipsoide SRC de la capa – extent: extensión de la capa (como un objeto de geometría) – distance_units: unidades de distancia de la capa – type: tipo de capa, p.ej., Vectorial o Ráster – storage_type: formato de almacenamiento (solo capas vectoriales) – geometry_type: tipo de geometría, por ejemplo, Punto (solo capas vectoriales) – feature_count: recuento aproximado de objetos por capa (solo capas vectoriales) – path: Ruta del archivo a la fuente de datos de la capa. Solo disponible para capas basadas en archivos.
Ejemplos	• `layer_property('streets','title')` → “Basemap Streets” • `layer_property('airports','feature_count')` → 120 • `layer_property('landsat','crs')` → “EPSG:4326”

Otras lecturas: *vector*, *raster* and *mesh* propiedades de la capa

load_layer

Carga una capa por URI de origen y nombre de proveedor.

Sintaxis	load_layer(uri, provider)
Argumentos	• **uri** - cadena URI de origen de la capa • **proveedor** - nombre del proveedor de datos de la capa
Ejemplos	• `layer_property(load_layer('c:/data/roads.shp', 'ogr'), 'feature_count')` → recuento de objetos espaciales de la capa vectorial c:/data/roads.shp.

9.2.17 Funciones de Mapas

Este grupo contiene funciones para crear o manipular claves y valores de estructuras de datos de mapas (también conocidos como objetos de diccionario, pares clave-valor o matrices asociativas). A diferencia del *list data structure* donde importa el orden de valores, el orden de los pares clave-valor en el objeto de mapa no es relevante y los valores se identifican por sus claves.

from_json

Carga una cadena con formato JSON.

Sintaxis	from_json(string)
Argumentos	• **string** - cadena JSON
Ejemplos	• `from_json('{"1":"one","2":"two"}')` → { “1”: “one”, “2”: “two” } • `from_json('[1,2,3]')` → [1,2,3]

hstore_to_map

Crea un mapa de una cadena con formato hstore.

Sintaxis	hstore_to_map(string)
Argumentos	• **string** - la cadena entrante
Ejemplos	• `hstore_to_map('qgis=>rocks')` → { “qgis”: “rocks” }

map

Devuelve un mapa conteniendo todas las claves y valores pasados como para de parámetros.

Sintaxis	map(key1, value1, key2, value2, …)
Argumentos	• **key** - una clave (cadena) • **value** - un valor
Ejemplos	• `map('1','one','2', 'two')` → { “1”: “one”, “2”: “two” } • `map('1','one','2', 'two')['1']` → “one”

map_akeys

Devuelve todas las claves de un mapa como una matriz.

Sintaxis	map_akeys(map)
Argumentos	• **map** - un mapa
Ejemplos	• `map_akeys(map('1','one','2','two'))` → [“1”, “2”]

map_avals

Devuelve todos los valore de un mapa como una matriz.

Sintaxis	map_avals(map)
Argumentos	• **map** - un mapa
Ejemplos	• `map_avals(map('1','one','2','two'))` → [“one”, “two”]

map_concat

Devuelve un mapa que contiene todas las entradas de los mapas dados. Si dos mapas contienen la misma clave, se toma el valor del segundo mapa.

Sintaxis	map_concat(map1, map2, …)
Argumentos	• **map** - un mapa
Ejemplos	• `map_concat(map('1','one', '2','overridden'),map('2','two', '3','three'))` → { “1”: “one”, “2”: “two”, “3”: “three” }

map_delete

Devuelve un mapa con la clave dada y su valor borrado correspondiente.

Sintaxis	map_delete(map, key)
Argumentos	• **map** - un mapa • **key** - la clave a borrar
Ejemplos	• `map_delete(map('1','one','2','two'),'2')` → { “1”: “one” }

map_exist

Devuelve TRUE si la clave dada existe en el mapa.

Sintaxis	map_exist(map, key)
Argumentos	• **map** - un mapa • **key** - la clave para buscar
Ejemplos	• `map_exist(map('1','one','2','two'),'3')` → FALSE

map_get

Devuelve el valor de un mapa, dada su clave. Devuelve NULL si la clave no existe.

Sintaxis	map_get(map, key)
Argumentos	• **map** - un mapa • **key** - la clave para buscar
Ejemplos	• `map_get(map('1','one','2','two'),'2')` → "two" • `map_get( item_variables('Map 0'), 'map_scale')` → escala del elemento "Map 0" (si existe) en el diseño de impresión actual

Consejo

También puede utilizar el *index operator ([])* para obtener un valor de un mapa.

map_insert

Devuelve un mapa con una clave/valor agregado. Si la clave ya existe, su valor se anula.

Sintaxis	map_insert(map, key, value)
Argumentos	• **map** - un mapa • **key** - la clave a añadir • **value** - el valor a agregar
Ejemplos	• `map_insert(map('1','one'),'3','three')` → { "1": "one", "3": "three" } • `map_insert(map('1','one','2','overridden'),'2','two')` → { "1": "one", "2": "two" }

map_prefix_keys

Devuelve un mapa con todas las claves prefijadas por una cadena dada.

Sintaxis	map_prefix_keys(map, prefix)
Argumentos	• **map** - un mapa • **prefix** - una cadena
Ejemplos	• `map_prefix_keys(map('1','one','2','two'), 'prefix-')` → { “prefix-1”: “one”, “prefix-2”: “two” }

map_to_hstore

Mezclar elementos de mapa en una cadena de formato hstore.

Sintaxis	map_to_hstore(map)
Argumentos	• **map** - el mapa de entrada
Ejemplos	• `map_to_hstore(map('1','one','2','two'))` → “«1»=>»one»”,“«2»=>»two»”

map_to_html_dl

Combinar elementos de mapa en una cadena de lista de definición HTML.

Sintaxis	map_to_html_dl(map)
Argumentos	• **map** - el mapa de entrada
Ejemplos	• `map_to_html_dl(map('1','one','2','two'))` → <dl><dt>1</dt><dd>one</dd><dt>2</dt><dd>two</dd></dl>

map_to_html_table

Combinar elementos de mapa en una cadena de tabla HTML.

Sintaxis	map_to_html_table(map)
Argumentos	• **map** - el mapa de entrada
Ejemplos	• `map_to_html_table(map('1','one','2','two'))` → <table><thead><tr><th>1</th><th>2</th></tr></thead><tbody><tr><td>one</td><td>two</td></tr></tb

to_json

Crea una cedena con formato JSON a partir de un mapa, matriz u otro valor.

Sintaxis	to_json(value)
Argumentos	• **value** - El valor de entrada
Ejemplos	• `to_json(map('1','one','2','two'))` → {«1»:»one»,»2»:»two»} • `to_json(array(1,2,3))` → [1,2,3]

url_encode

Devuelve una cadena URL codificada a partir de un mapa. Transforma todos los caracteres en su forma correctamente codificada produciendo una cadena de consulta totalmente compatible.

Tenga en cuenta que el signo «+» no se convierte.

Sintaxis	url_encode(map)
Argumentos	• **map** - un mapa.
Ejemplos	• `url_encode(map('a&+b', 'a and plus b', 'a=b', 'a equals b'))` → “a%26+b=a%20and%20plus%20b&a%3Db=a%20equals%20b”

9.2.18 Funciones Matemáticas

Este grupo contiene funciones matemáticas (por ejemplo raíz cuadrada, seno y coseno).

abs

Devuelve el valor absoluto de un número.

Sintaxis	abs(value)
Argumentos	• **value** - un número
Ejemplos	• `abs(-2)` → 2

acos

Devuelve el arcocoseno de un valor en radianes.

Sintaxis	acos(value)
Argumentos	• **value** - coseno de un ángulo en radianes
Ejemplos	• `acos(0.5)` → 1.0471975511966

asin

Devuelve el arcoseno de un valor en radianes.

Sintaxis	asin(value)
Argumentos	• **value** - seno de un ángulo en radianes
Ejemplos	• `asin(1.0)` → 1.5707963267949

atan

Devuelve el arcotangente de un valor en radianes.

Sintaxis	atan(value)
Argumentos	• **value** - tangente de un ángulo en radianes
Ejemplos	• `atan(0.5)` → 0.463647609000806

atan2

Devuelve el arcotangente de dy/dx usando los signos de dos argumentos para determinar el cuadrante del resultado.

Sintaxis	atan2(dy, dx)
Argumentos	• **dy** - diferencia de coordenadas y • **dx** - diferencia de coordenadas x
Ejemplos	• `atan2(1.0, 1.732)` → 0.523611477769969

ceil

Redondea un número hacia arriba.

Sintaxis	ceil(value)
Argumentos	• **value** - un número
Ejemplos	• `ceil(4.9)` → 5 • `ceil(-4.9)` → -4

clamp

Restringe un valor de entrada a un rango especificado.

Sintaxis	clamp(minimum, input, maximum)
Argumentos	• **minimum** - se permite tomar el valor más pequeño *entrada*. • **input** - un valor que estará restringido al rango especificado por *mínimo* y *máximo* • **maximum** - el mayor valor *entrada* se permite tomar
Ejemplos	• `clamp(1,5,10)` → 5 *entrada* está entre 1 y 10, por lo que se devuelve sin cambios • `clamp(1,0,10)` → 1 *entrada* es menor que el valor mínimo de 1, por lo que la función devuelve 1 • `clamp(1,11,10)` → 10 *entrada* es mayor que el valor máximo de 10, por lo que la función devuelve 10

cos

Devuelve el coseno de un ángulo.

Sintaxis	cos(angle)
Argumentos	• **angle** - ángulo en radianes
Ejemplos	• `cos(1.571)` → 0.000796326710733263

degrees

Convierte de radianes a grados.

Sintaxis	degrees(radians)
Argumentos	• **radians** - valor numérico
Ejemplos	• `degrees(3.14159)` → 180 • `degrees(1)` → 57.2958

exp

Devuelve exponencial de un valor.

Sintaxis	exp(value)
Argumentos	• **value** - número del cuál devolver el exponente
Ejemplos	• `exp(1.0)` → 2.71828182845905

floor

Redondea un número hacia abajo.

Sintaxis	floor(value)
Argumentos	• **value** - un número
Ejemplos	• `floor(4.9)` → 4 • `floor(-4.9)` → -5

ln

Devuelve el logaritmo natural de un valor.

Sintaxis	ln(value)
Argumentos	• **value** - valor numérico
Ejemplos	• `ln(1)` → 0 • `ln(2.7182818284590452354)` → 1

log

Devuelve el valor del logaritmo del valor pasado y la base.

Sintaxis	log(base, value)
Argumentos	• **base** - cualquier número positivo • **value** - cualquier número positivo
Ejemplos	• `log(2, 32)` → 5 • `log(0.5, 32)` → -5

log10

Devuelve el valor del logaritmo de base 10 de la expresión pasada.

Sintaxis	log10(value)
Argumentos	• **value** - cualquier número positivo
Ejemplos	• `log10(1)` → 0 • `log10(100)` → 2

max

Devuelve el valor más grande de un conjunto de valores.

Sintaxis	max(value1, value2, …)
Argumentos	• **value** - un número
Ejemplos	• `max(2,10.2,5.5)` → 10.2 • `max(20.5,NULL,6.2)` → 20.5

min

Devuelve el valor más pequeño de un conjunto de valores.

Sintaxis	min(value1, value2, …)
Argumentos	• **value** - un número
Ejemplos	• `min(20.5,10,6.2)` → 6.2 • `min(2,-10.3,NULL)` → -10.3

pi

Devuelve el valor de pi para los cálculos.

Sintaxis	pi()
Ejemplos	• `pi()` → 3.14159265358979

radians

Convierte de grados a radianes.

Sintaxis	radians(degrees)
Argumentos	• **degrees** - valor numérico
Ejemplos	• `radians(180)` → 3.14159 • `radians(57.2958)` → 1

rand

Devuelve un número entero aleatorio dentro del rango especificado por el argumento mínimo y máximo (inclusive). Si se proporciona una semilla, la devolución siempre será la misma, dependiendo de la semilla.

Sintaxis	rand(min, max, [seed=NULL]) [] marca argumentos opcionales
Argumentos	• **min** - un entero que representa el número aleatorio más pequeño posible deseado • **max** - un entero que representa el mayor número aleatorio posible deseado • **seed** - cualquier valor para usar como semilla
Ejemplos	• `rand(1, 10)` → 8

randf

Devuelve un flotante aleatorio dentro del rango especificado por el argumento mínimo y máximo (inclusive). Si se proporciona una semilla, la devolución siempre será la misma, dependiendo de la semilla.

Sintaxis	randf([min=0.0], [max=1.0], [seed=NULL]) [] marca argumentos opcionales
Argumentos	• **min** - un flotador que representa el número aleatorio más pequeño posible deseado • **max** - un flotador que representa el mayor número aleatorio posible deseado • **seed** - cualquier valor para usar como semilla
Ejemplos	• `randf(1, 10)` → 4.59258286403147

round

Redondea un número al número de posiciones decimales.

Sintaxis	round(value, [places=0]) [] marca argumentos opcionales
Argumentos	• **value** - número decimal a redondear • **places** - Entero opcional que representa el número de lugares para redondear decimales. Puede ser negativo.
Ejemplos	• `round(1234.567, 2)` → 1234.57 • `round(1234.567)` → 1235 • `round(1234.567, -1)` → 1230

scale_exponential

Transforma un valor dado de un dominio de entrada a un rango de salida usando una curva exponencial. Esta función se puede utilizar para facilitar valores dentro o fuera del rango de salida especificado.

Sintaxis	scale_exponential(value, domain_min, domain_max, range_min, range_max, exponent)
Argumentos	• **value** - Un valor en el dominio de entrada. La función devolverá un valor escalado correspondiente en el rango de salida. • **domain_min** - Especifica el valor mínimo en el dominio de entrada, el valor más pequeño que debe tomar el valor de entrada. • **domain_max** - Especifica el valor máximo en el dominio de entrada, el valor más grande que debe tomar el valor de entrada. • **range_min** - Especifica el valor mínimo en el rango de salida, el valor más pequeño que debe generar la función. • **range_max** - Especifica el valor máximo en el rango de salida, el valor más grande que debe generar la función. • **exponent** - Un valor positivo (mayor que 0), que dicta la forma en que los valores de entrada se asignan al rango de salida. Los exponentes grandes harán que los valores de salida "ease in",, comenzando lentamente antes de acelerar a medida que los valores de entrada se acercan al máximo del dominio. Los exponentes más pequeños (menos de 1) harán que los valores de salida se "ease out,, donde el mapeo comienza rápidamente pero se ralentiza a medida que se acerca al máximo del dominio.
Ejemplos	• `scale_exponential(5,0,10,0,100,2)` → 3.030 entrando, usando un exponente de 2 • `scale_exponential(3,0,10,0,100,0.5)` → 87.585 saliendo, usando un exponente de 0.5

scale_linear

Transforma un valor dado de un dominio de entrada a un rango de salida mediante interpolación lineal.

Sintaxis	scale_linear(value, domain_min, domain_max, range_min, range_max)
Argumentos	• **value** - Un valor en el dominio de entrada. La función devolverá un valor escalado correspondiente en el rango de salida. • **domain_min** - Especifica el valor mínimo en el dominio de entrada, el valor más pequeño que debe tomar el valor de entrada. • **domain_max** - Especifica el valor máximo en el dominio de entrada, el valor más grande que debe tomar el valor de entrada. • **range_min** - Especifica el valor mínimo en el rango de salida, el valor más pequeño que debe generar la función. • **range_max** - Especifica el valor máximo en el rango de salida, el valor más grande que debe generar la función.
Ejemplos	• `scale_linear(5,0,10,0,100)` → 50 • `scale_linear(0.2,0,1,0,360)` → 72 escalar un valor entre 0 y 1 a un ángulo entre 0 y 360 • `scale_linear(1500,1000,10000,9,20)` → 9.6111111 escalar una población que varía entre 1000 y 10000 a un tamaño de fuente entre 9 y 20

scale_polynomial

Transforma un valor dado de un dominio de entrada a un rango de salida utilizando una curva polinómica. Esta función puede utilizarse para facilitar valores dentro o fuera del rango de salida especificado.

Sintaxis	scale_polynomial(value, domain_min, domain_max, range_min, range_max, exponent)
Argumentos	• **value** - Un valor en el dominio de entrada. La función devolverá un valor escalado correspondiente en el rango de salida. • **domain_min** - Especifica el valor mínimo en el dominio de entrada, el valor más pequeño que debe tomar el valor de entrada. • **domain_max** - Especifica el valor máximo en el dominio de entrada, el valor más grande que debe tomar el valor de entrada. • **range_min** - Especifica el valor mínimo en el rango de salida, el valor más pequeño que debe generar la función. • **range_max** - Especifica el valor máximo en el rango de salida, el valor más grande que debe generar la función. • **exponent** - Un valor positivo (mayor que 0), que dicta la forma en que los valores de entrada se asignan al rango de salida. Los exponentes grandes harán que los valores de salida "ease in",, comenzando lentamente antes de acelerar a medida que los valores de entrada se acercan al máximo del dominio. Los exponentes más pequeños (menos de 1) harán que los valores de salida se "ease out,, donde el mapeo comienza rápidamente pero se ralentiza a medida que se acerca al máximo del dominio.
Ejemplos	• `scale_polynomial(5,0,10,0,100,2)` → 25 entrando, usando un exponente de 2 • `scale_polynomial(3,0,10,0,100,0.5)` → 54.772 saliendo, usando un exponente de 0.5

sen

Devuelve el seno de un ángulo.

Sintaxis	sin(angle)
Argumentos	• **angle** - ángulo en radianes
Ejemplos	• `sin(1.571)` → 0.999999682931835

sqrt

Devuelve la raiz cuadrada de un valor.

Sintaxis	sqrt(value)
Argumentos	• **value** - un número
Ejemplos	• `sqrt(9)` → 3

tan

Devuelve la tangente de un ángulo.

Sintaxis	tan(angle)
Argumentos	• **angle** - ángulo en radianes
Ejemplos	• `tan(1.0)` → 1.5574077246549

9.2.19 Funciones de mallas

Este grupo contiene funciones que calculan o devuelven valores relacionados con la malla.

$face_area

Devuelve el área de la cara de malla actual. El área calculada por esta función respeta tanto la configuración del elipsoide del proyecto actual como la configuración de la unidad de área. Por ejemplo, si se ha establecido un elipsoide para el proyecto, el área calculada será elipsoidal, y si no se establece ningún elipsoide, el área calculada será planimétrica.

Sintaxis	$face_area
Ejemplos	• `$face_area` → 42

$face_index

Devuelve el índice de la cara de malla actual.

Sintaxis	$face_index
Ejemplos	• `$face_index` → 4581

$vertex_as_point

Devuelve el vértice actual como una geometría de punto.

Sintaxis	$vertex_as_point
Ejemplos	• `geom_to_wkt( $vertex_as_point )` → “POINT(800 1500 41)”

$vertex_index

Devuelve el índice del vértice de la malla actual.

Sintaxis	$vertex_index
Ejemplos	• `$vertex_index` → 9874

$vertex_x

Devuelve la coordenada X del vértice de la malla actual.

Sintaxis	$vertex_x
Ejemplos	• `$vertex_x` → 42.12

$vertex_y

Devuelve la coordenada Y del vértice de la malla actual.

Sintaxis	$vertex_y
Ejemplos	• `$vertex_y` → 12.24

$vertex_z

Devuelve el valor Z del vértice de malla actual.

Sintaxis	$vertex_z
Ejemplos	• `$vertex_z` → 42

9.2.20 Operadores

Este grupo contiene operadores (por ejemplo, +, -, *). Tenga en cuenta que para la mayoría de las funciones matemáticas a continuación, si una de las entradas es NULL, el resultado es NULL.

%

Resto de la división. Toma el signo del dividendo.

Sintaxis	a % b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `9 % 2` → 1 • `9 % -2` → 1 • `-9 % 2` → -1 • `5 % NULL` → NULL

*

Multiplicación de dos valores

Sintaxis	a * b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 * 4` → 20 • `5 * NULL` → NULL

+

Suma de dos valores. Si uno de los valores es NULL, el resultado será NULL.

Sintaxis	a + b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 + 4` → 9 • `5 + NULL` → NULL • `'QGIS ' + 'ROCKS'` → "QGIS ROCKS" • `to_datetime('2020-08-01 12:00:00') + '1 day 2 hours'` → 2020-08-02T14:00:00

Otras lecturas: *concat*, ||

-

Resta de dos valores. Si uno de los valores es NULL, el resultado será NULL.

Sintaxis	a - b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 - 4` → 1 • `5 - NULL` → NULL • `to_datetime('2012-05-05 12:00:00') - to_interval('1 day 2 hours')` → 2012-05-04T10:00:00

/

División de dos valores

Sintaxis	a / b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 / 4` → 1.25 • `5 / NULL` → NULL

//

División entera de dos valores

Sintaxis	a // b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `9 // 2` → 4

<

Compara dos valores y evalúa a 1 si el valor de la izquierda es menor que el valor de la derecha.

Sintaxis	a < b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 < 4` → FALSE • `5 < 5` → FALSE • `4 < 5` → TRUE

<=

Compara dos valores y evalúa a 1 si el valor de la izquierda es menor o igual que el valor de la derecha.

Sintaxis	a <= b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 <= 4` → FALSE • `5 <= 5` → TRUE • `4 <= 5` → TRUE

<>

Compara dos valores y evalúa a 1 si no son iguales.

Sintaxis	a <> b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 <> 4` → TRUE • `4 <> 4` → FALSE • `5 <> NULL` → NULL • `NULL <> NULL` → NULL

=

Compara dos valores y evalúa a 1 si son iguales.

Sintaxis	a = b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 = 4` → FALSE • `4 = 4` → TRUE • `5 = NULL` → NULL • `NULL = NULL` → NULL

>

Compara dos valores y evalúa a 1 si el valor de la izquierda es mayor que el valor de la derecha.

Sintaxis	a > b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 > 4` → TRUE • `5 > 5` → FALSE • `4 > 5` → FALSE

>=

Compara dos valores y evalúa a 1 si el valor de la izquierda es mayor o igual que el valor de la derecha.

Sintaxis	a >= b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 >= 4` → TRUE • `5 >= 5` → TRUE • `4 >= 5` → FALSE

AND

Devuelve TRUE cuando las condiciones a y b son verdaderas.

Sintaxis	a AND b
Argumentos	• **a** - condición • **b** - condición
Ejemplos	• `TRUE AND TRUE` → TRUE • `TRUE AND FALSE` → FALSE • `4 = 2+2 AND 1 = 1` → TRUE • `4 = 2+2 AND 1 = 2` → FALSE

BETWEEN

Devuelve TRUE si el valor está dentro del rango especificado. El rango se considera inclusivo de los límites. Para comprobar la exclusión se puede utilizar NOT BETWEEN.

Sintaxis	valor lower_bound AND higher_bound
Argumentos	• **value** - el valor a comparar con un rango. Puede ser una cadena, un número o una fecha. • **lower_bound AND higher_bound** - límites del intervalo
Ejemplos	• `'B' BETWEEN 'A' AND 'C'` → TRUE • `2 BETWEEN 1 AND 3` → TRUE • `2 BETWEEN 2 AND 3` → TRUE • `'B' BETWEEN 'a' AND 'c'` → FALSE • `lower('B') BETWEEN 'a' AND 'b'` → TRUE

Nota

value BETWEEN lower_bound AND higher_bound es lo mismo que «*value* >= *lower_bound AND value* <= *higher_bound*».

Lectura adicional: *NOT BETWEEN*

ILIKE

Devuelve TRUE si el primer parámetro coincide con el patrón suministrado sin distinguir mayúsculas de minúsculas. Se puede utilizar LIKE en lugar de ILIKE para que la coincidencia distinga entre mayúsculas y minúsculas. También funciona con números.

Sintaxis	cadena de texto/número ILIKE patrón
Argumentos	• **string/number** - cadena para buscar • **pattern** - patrón para encontrar, puede usar "%" como comodín, "_" como un solo carácter y "\\" para escapar de estos caracteres especiales.
Ejemplos	• `'A' ILIKE 'A'` → TRUE • `'A' ILIKE 'a'` → TRUE • `'A' ILIKE 'B'` → FALSE • `'ABC' ILIKE 'b'` → FALSE • `'ABC' ILIKE 'B'` → FALSE • `'ABC' ILIKE '_b_'` → TRUE • `'ABC' ILIKE '_B_'` → TRUE • `'ABCD' ILIKE '_b_'` → FALSE • `'ABCD' ILIKE '_B_'` → FALSE • `'ABCD' ILIKE '_b%'` → TRUE • `'ABCD' ILIKE '_B%'` → TRUE • `'ABCD' ILIKE '%b%'` → TRUE • `'ABCD' ILIKE '%B%'` → TRUE • `'ABCD%' ILIKE 'abcd\\%'` → TRUE • `'ABCD' ILIKE '%B\\%'` → FALSE

IN

Devuelve TRUE si el valor se encuentra dentro de una lista de valores.

Sintaxis	a IN b
Argumentos	• **a** - valor • **b** - lista de valores
Ejemplos	• `'A' IN ('A','B')` → TRUE • `'A' IN ('C','B')` → FALSE

IS

Devuelve TRUE si a es igual a b.

Sintaxis	a IS b
Argumentos	• **a** - algún valor • **b** - algún valor
Ejemplos	• `'A' IS 'A'` → TRUE • `'A' IS 'a'` → FALSE • `4 IS 4` → TRUE • `4 IS 2+2` → TRUE • `4 IS 2` → FALSE • `@geometry IS NULL` → 0, si tu geometría no es NULL.

IS NOT

Devuelve TRUE si a no es igual a b.

Sintaxis	a IS NOT b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `'a' IS NOT 'b'` → TRUE • `'a' IS NOT 'a'` → FALSE • `4 IS NOT 2+2` → FALSE

LIKE

Devuelve TRUE si el primer parámetro coincide con el patrón suministrado. También funciona con números.

Sintaxis	cadena de texto/número ILIKE patrón
Argumentos	• **cadena de texto/número** - valor • **pattern** - patrón con el que comparar el valor, puede usar "%" como comodín, "_" como un solo carácter y "\\" para escapar de estos caracteres especiales.
Ejemplos	• `'A' LIKE 'A'` → TRUE • `'A' LIKE 'a'` → FALSE • `'A' LIKE 'B'` → FALSE • `'ABC' LIKE 'B'` → FALSE • `'ABC' LIKE '_B_'` → TRUE • `'ABCD' LIKE '_B_'` → FALSE • `'ABCD' LIKE '_B%'` → TRUE • `'ABCD' LIKE '%B%'` → TRUE • `'1%' LIKE '1\\%'` → TRUE • `'1_' LIKE '1\\%'` → FALSE

NOT

Niega una condición.

Sintaxis	NOT a
Argumentos	• **a** - condición
Ejemplos	• `NOT 1` → FALSE • `NOT 0` → TRUE

NOT BETWEEN

Devuelve TRUE si el valor no está dentro del rango especificado. El rango se considera inclusivo de los límites.

Sintaxis	value NOT BETWEEN lower_bound AND higher_bound
Argumentos	• **value** - el valor a comparar con un rango. Puede ser una cadena, un número o una fecha. • **lower_bound AND higher_bound** - límites del intervalo
Ejemplos	• `'B' NOT BETWEEN 'A' AND 'C'` → FALSE • `1.0 NOT BETWEEN 1.1 AND 1.2` → TRUE • `2 NOT BETWEEN 2 AND 3` → FALSE • `'B' NOT BETWEEN 'a' AND 'c'` → TRUE • `lower('B') NOT BETWEEN 'a' AND 'b'` → FALSE

Nota

value NOT BETWEEN lower_bound AND higher_bound es lo mismo que «*value* < *lower_bound OR value* > *higher_bound*».

Lectura adicional: *BETWEEN*

OR

Devuelve TRUE cuando la condición a o b es verdadera.

Sintaxis	a OR b
Argumentos	• **a** - condición • **b** - condición
Ejemplos	• `4 = 2+2 OR 1 = 1` → TRUE • `4 = 2+2 OR 1 = 2` → TRUE • `4 = 2 OR 1 = 2` → FALSE

[]

Index operator. Devuelve un elemento de una matriz o un valor de mapa.

Sintaxis	[índice]
Argumentos	• **index** - índice de matriz o valor de clave de mapa
Ejemplos	• `array(1,2,3)[0]` → 1 • `array(1,2,3)[2]` → 3 • `array(1,2,3)[-1]` → 3 • `map('a',1,'b',2)['a']` → 1 • `map('a',1,'b',2)['b']` → 2

Otras lecturas: *array_get*, *map_get*

^

Potencia de dos valores.

Sintaxis	a ^ b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `5 ^ 4` → 625 • `5 ^ NULL` → NULL

||

Une dos valores en una cadena.

Si uno de los valores es NULL, el resultado será NULL. Ver la función CONCAT para un comportamiento diferente.

Sintaxis	a || b
Argumentos	• **a** - valor • **b** - valor
Ejemplos	• `'Here' || ' and ' || 'there'` → “Here and there” • `'Nothing' || NULL` → NULL • `'Dia: ' || "Diameter"` → “Dia: 25” • `1 || 2` → “12”

Otras lecturas: *concat*, *+*

~

Realiza una coincidencia de expresión regular en un valor de cadena. Los caracteres de barra invertida deben tener doble escape (por ejemplo, «\\s» para que coincida con un carácter de espacio en blanco).

Sintaxis	string ~ regex
Argumentos	• **string** - un valor cadena • **regex** - Una expresión regular. Las barras deben tener caracteres de escape, por ejemplo, \\d.
Ejemplos	• `'hello' ~ 'll'` → TRUE • `'hello' ~ '^ll'` → FALSE • `'hello' ~ 'llo$'` → TRUE • `'abc123' ~ '\\d+'` → TRUE

Otras lecturas: *regexp_match*

9.2.21 Funciones de Procesamiento

Este grupo contiene funciones que operan en algoritmos de procesamiento.

parameter

Devuelve el valor de un parámetro de entrada del algoritmo de procesamiento.

Sintaxis	parameter(name)
Argumentos	• **name** - nombre del correspondiente parámetro de entrada
Ejemplos	• `parameter('BUFFER_SIZE')` → 5.6

9.2.22 Funciones Ráster

Este grupo contiene funciones a operar en una capa ráster.

raster_attributes

Devuelve un mapa con los nombres de los campos como claves y los valores de la tabla de atributos raster como valores de la entrada de la tabla de atributos que coincide con el valor raster dado.

Sintaxis	raster_attributes(layer, band, value)
Argumentos	• **layer** - el nombre o id de una capa ráster • **banda** - el número de banda para la búsqueda en la tabla de atributos asociada. • **valor** - valor ráster
Ejemplos	• `raster_attributes('vegetation', 1, raster_value('vegetation', 1, make_point(1,1)))` → {"class": "Vegetated", "subclass": "Trees"} • `raster_attributes('vegetation', 1, raster_value('vegetation', 1, @layer_cursor_point))` → {"class": "Vegetated", "subclass": "Trees"}

raster_statistic

Devuelve estadísticas de una capa ráster.

Sintaxis	raster_statistic(layer, band, property)
Argumentos	• **layer** - una cadena, que representa un nombre de capa ráster o un ID de capa • **band** - entero que representa el número de banda de la capa ráster, comenzando en 1 • **property** - una cadena correspondiente a la propiedad a devolver. Las opciones válidas son: – min: valor mínimo – max: valor máximo – avg: valor promedio (media) – stdev: desviación estándar de valores – range: rango de valores (máx - mín) – sum: suma de todos los valores de un ráster
Ejemplos	• `raster_statistic('lc',1,'avg')` → Valor promedio de la banda 1 de la capa ráster "lc" • `raster_statistic('ac2010',3,'min')` → Valor mínimo de la banda 3 de la capa ráster "ac2010"

raster_value

Devuelve el valor de ráster encontrado en el punto proporcionado.

Sintaxis	raster_value(layer, band, point)
Argumentos	• **layer** - el nombre o id de una capa ráster • **band** - el número de banda de la que muestrear el valor. • **point** - geometría de puntos (para geometrías multiparte que tienen más de una parte, se devolverá un valor NULL)
Ejemplos	• `raster_value('dem', 1, make_point(1,1))` → 25 • `raster_value('ndvi', 2, @layer_cursor_point)` → 25

9.2.23 Funciones de Registro y Atributos

Este grupo contiene funciones que operan sobre identificadores de registros.

attribute

Devuelve un atributo de un objeto.

Variante 1

Devuelve el valor de un atributo de la entidad actual.

Sintaxis	attribute(attribute_name)
Argumentos	• **attribute_name** - nombre del atributo a devolver
Ejemplos	• `attribute( 'name' )` → valor almacenado en el atributo "name" para la entidad actual

Variante 2

Permite ser especificados a la entidad objjetivo y al nombre de atributo.

Sintaxis	attribute(feature, attribute_name)
Argumentos	• **feature** - un objeto espacial • **attribute_name** - nombre del atributo a devolver
Ejemplos	• `attribute( @atlas_feature, 'name' )` → valor almacenado en el atributo "name" para la actual entidad del atlas

attributes

Devuelve un mapa conteniendo todos los atributos para una entidad, con nombres de campo como claves de mapa.

Variante 1

Devuelve un mapa de todos los atributos de la entidad actual.

Sintaxis	attributes()
Ejemplos	• `attributes()['name']` → valor almacenado en el atributo "name" para la entida actual

Variante 2

Permite ser especificada a la entidad objetivo.

Sintaxis	attributes(feature)
Argumentos	• **feature** - un objeto espacial
Ejemplos	• `attributes( @atlas_feature )['name']` → valor almacenado en el atributo "name" para la entidad actual de atlas

Otras lecturas: *Funciones de Mapas*

$currentfeature

Devuelve la característica actual que se está evaluando. Puede utilizarse con la función «attribute» para evaluar los valores de los atributos de la característica actual. **AVISO: Esta función está obsoleta. Se recomienda utilizar en su lugar la variable de sustitución @característica.**

Sintaxis	$currentfeature
Ejemplos	• `attribute( $currentfeature, 'name' )` → valor almacenado en el atributo "name" para la entidad actual

display_expression

Devuelve la expresión de visualización de una entidad determinada en una capa. La expresión se evalúa de forma predeterminada. Se puede utilizar con cero, uno o más argumentos; consulte los detalles a continuación.

Sin parámetros

Si se llama sin parámetros, la función evaluará la expresión de visualización de la entidad actual en la capa actual.

Sintaxis	display_expression()
Ejemplos	• `display_expression()` → La expresión de visualización de la entidad actual en la capa actual.

** Un parámetro de "entidad" **

Si se llama solo con un parámetro de "entidad", la función evaluará la característica especificada de la capa actual.

Sintaxis	display_expression(feature)
Argumentos	• **feature** - La entidad a ser evaluada.
Ejemplos	• `display_expression(@atlas_feature)` → La expresión de visualización de la entidad actual de atlas.

Parámetros de capa y entidad

Si se llama a la función con una capa y una entidad, evaluará la entidad especificada de la capa especificada.

Sintaxis	display_expression(layer, feature, [evaluate=true]) [] marca argumentos opcionales
Argumentos	• **layer** - La capa (o su ID o nombre) • **feature** - La entidad a ser evaluada. • **evaluate** - Si la expresión debe evaluarse. Si es falso, la expresión se devolverá solo como una cadena literal (que potencialmente podría evaluarse más adelante mediante la función "eval").
Ejemplos	• `display_expression( 'streets', get_feature_by_id('streets', 1))` → La expresión de visualización de la entidad con el ID 1 en la capa "calles". • `display_expression('a_layer_id', @característica, 'False')` → No se ha evaluado la expresión de visualización del objeto espacial dado.

feature_id

Devuelve el ID único de un objeto espacial, o NULL si el objeto espacial no es válido.

Sintaxis	feature_id(feature)
Argumentos	• **feature** - un objeto característico
Ejemplos	• `feature_id( @feature )` → el ID del objeto espacial actual.

Lectura adicional: *get_feature_by_id*

get_feature

Devuelve la primera entidad de una capa coincidente con un valor de atributo dado.

Variante de valor único

Junto con el ID de la capa, se especifican una única columna y un valor.

Sintaxis	get_feature(layer, attribute, value)
Argumentos	• **layer** - nombre de capa o ID • **attribute** - nombre del atributo que se utilizará para la coincidencia • **value** - valor de atributo a coincidir
Ejemplos	• `get_feature('streets','name','main st')` → primera entidad encontrada en la capa «calles» con el valor «main st» en el campo «nombre»

Map variant

Junto con el ID de la capa, un mapa que contiene las columnas (clave) y su respectivo valor a utilizar.

Sintaxis	get_feature(layer, attribute)
Argumentos	• **layer** - nombre de capa o ID • **attribute** - Mapa que contiene los pares de columnas y valores a utilizar
Ejemplos	• `get_feature('streets',map('name','main st','lane_num','4'))` → primer elemento encontrado en la capa «streets» con el valor «main st» en el campo «name» y el valor «4» en el campo «lane_num».

get_feature_by_id

vuelve la entidad con una id en una capa.

Sintaxis	get_feature_by_id(layer, feature_id)
Argumentos	• **layer** - capa, nombre de capa o id de capa • **feature_id** - la id de la entidad que debe ser devuelta
Ejemplos	• `get_feature_by_id('streets', 1)` → la entidad con el id 1 en la capa «streets»

Lectura adicional: *feature_id*

$id

Devuelve el id del objeto espacial de la fila actual. **AVISO: Esta función está obsoleta. Se recomienda utilizar la variable @id en su lugar.

Sintaxis	$id
Ejemplos	• `$id` → 42

Lectura adicional: *feature_id*, *get_feature_by_id*

is_attribute_valid

Devuelve TRUE si un atributo de objeto espacial específico cumple todas las restricciones.

Sintaxis	is_attribute_valid(attribute, [feature], [layer], [strength]) [] marca argumentos opcionales
Argumentos	• **atributo** - un nombre de atributo • **objeto espacial** - Un objeto espacial. Si no se establece, se utilizará el objeto espacial actual. • **Capa** - Una capa vectorial. Si no se establece, se utilizará la capa actual. • **fuerza** - Establecer como «dura» o «blanda» para restringirse a un tipo de restricción específico. Si no se establece, la función devolverá FALSE si falla una restricción dura o blanda.
Ejemplos	• `is_attribute_valid('HECTARES')` → TRUE si el valor del objeto espacial actual en el campo «HECTARES» cumple todas las restricciones. • `is_attribute_valid('HOUSES',get_feature('my_layer', 'FID', 10), 'my_layer')` → FALSE si el valor del campo «HOUSES» del objeto espacial con «FID»=10 en “my_layer” no cumple todas las restricciones.

Lectura adicional: *Restricciones*

is_feature_valid

Devuelve TRUE si un objeto espacial cumple todas las restricciones de campo.

Sintaxis	is_feature_valid([feature], [layer], [strength]) [] marca argumentos opcionales
Argumentos	• **objeto espacial** - Un objeto espacial. Si no se establece, se utilizará el objeto espacial actual. • **Capa** - Una capa vectorial. Si no se establece, se utilizará la capa actual. • **fuerza** - Establecer como «dura» o «blanda» para restringirse a un tipo de restricción específico. Si no se establece, la función devolverá FALSE si falla una restricción dura o blanda.
Ejemplos	• `is_feature_valid(strength:='hard')` → TRUE si todos los campos del objeto espacial actual cumplen sus restricciones duras. • `is_feature_valid(get_feature('mi_capa', 'FID', 10), 'mi_capa')` → FALSE si todos los campos del objeto espacial con «FID»=10 en “mi_capa” no cumplen todas las restricciones.

Lectura adicional: *Restricciones*

is_selected

Devuelve TRUE si se ha seleccionado un objeto espacial. Se puede utilizar con cero, uno o dos argumentos, véase más abajo para más detalles.

Sin parámetros

Si se llama sin parámetros, la función devolverá TRUE si la característica actual en la capa actual está seleccionada.

Sintaxis	is_selected()
Ejemplos	• `is_selected()` → TRUE si el objeto espacial actual de la capa actual está seleccionada.

** Un parámetro de “entidad” **

Si se invoca sólo con el parámetro “feature”, la función devuelve TRUE si se ha seleccionado la característica especificada de la capa actual.

Sintaxis	is_selected(feature)
Argumentos	• **feature** - La entidad que se debe verificar para su selección.
Ejemplos	• `is_selected(@atlas_feature)` → TRUE si el objeto espacial actual del atlas está seleccionado. • `is_selected(get_feature('streets', 'name', 'Main St.'))` → TRUE if the unique named «Main St.» feature on the active «streets» layer is selected. • `is_selected(get_feature_by_id('streets', 1))` → TRUE si el objeto espacial con el id 1 en la capa «streets» activa está seleccionado.

Dos parámetros

Si se llama a la función con una capa y un objeto espacial, devolverá TRUE si se selecciona la característica especificada de la capa especificada.

Sintaxis	is_selected(layer, feature)
Argumentos	• **capa** - La capa (su ID o nombre) en la que se comprobará la selección. • **feature** - La entidad que se debe verificar para su selección.
Ejemplos	• `is_selected( 'streets', get_feature('streets', 'name', "street_name"))` → TRUE si la calle del edificio actual está seleccionada (suponiendo que la capa del edificio tiene un campo llamado “street_name” y la capa “streets” tiene un campo llamado “name” con valores únicos). • `is_selected('streets', get_feature_by_id('streets', 1))` → TRUE si el objeto espacial con el id 1 en la capa «streets» está seleccionado.

maptip

Devuelve el maptip para una entidad determinada en una capa. La expresión se evalúa de forma predeterminada. Se puede utilizar con cero, uno o más argumentos; consulte los detalles a continuación.

Sin parámetros

Si se llama sin parámetros, la función evaluará el mapa de la entidad actual en la capa actual.

Sintaxis	maptip()
Ejemplos	• `maptip()` → El mapa de la entidad actual en la capa actual.

** Un parámetro de “entidad” **

Si se llama solo con un parámetro de "entidad", la función evaluará la característica especificada de la capa actual.

Sintaxis	maptip(feature)
Argumentos	• **feature** - La entidad a ser evaluada.
Ejemplos	• `maptip(@atlas_feature)` → El mapa de la entidad de atlas actual.

Parámetros de capa y entidad

Si se llama a la función con una capa y una entidad, evaluará la entidad especificada de la capa especificada.

Sintaxis	maptip(layer, feature, [evaluate=true]) [] marca argumentos opcionales
Argumentos	• **layer** - La capa (o su ID o nombre) • **feature** - La entidad a ser evaluada. • **evaluate** - Si la expresión debe evaluarse. Si es falso, la expresión se devolverá solo como una cadena literal (que potencialmente podría evaluarse más adelante usando la función "eval_template").
Ejemplos	• `maptip('streets', get_feature_by_id('streets', 1))` → El maptip de la entidad con el ID 1 en la capa "calles". • `maptip('a_layer_id', @feature, 'Falso')` → No se ha evaluado la sugerencia de mapa del objeto espacial dado.

num_selected

Devuelve el número de entidades seleccionadas en una capa determinada. Por defecto funciona en la capa en la que se evalúa la expresión.

Sintaxis	num_selected([layer=current layer]) [] marca argumentos opcionales
Argumentos	• **layer** - La capa (o su identificación o nombre) en la que se comprobará la selección.
Ejemplos	• `num_selected()` → El número de entidades seleccionadas en la capa actual. • `num_selected('streets')` → El número de entidades seleccionadas en las calles de la capa.

represent_attributes

Devuelve un mapa con los nombres de los atributos como claves y los valores de representación configurados como valores. El valor de representación de los atributos depende del tipo de widget configurado para cada atributo. Se puede utilizar con cero, uno o más argumentos, véase más abajo para más detalles.

Sin parámetros

Si se llama sin parámetros, la función devolverá la representación de los atributos de la característica actual en la capa actual.

Sintaxis	represent_attributes()
Ejemplos	• `represent_attributes()` → La representación de los atributos del objeto espacial actual.

** Un parámetro de “entidad” **

Si se llama sólo con el parámetro “feature”, la función devolverá la representación de los atributos del objeto espacial especificado de la capa actual.

Sintaxis	represent_attributes(feature)
Argumentos	• **feature** - La entidad a ser evaluada.
Ejemplos	• `represent_attributes(@atlas_feature)` → La representación de los atributos del objeto espacial especificado de la capa actual.

Parámetros de capa y entidad

Si se invoca con un parámetro «layer» y «feature», la función devolverá la representación de los atributos del objeto espacial especificado a partir de la capa especificada.

Sintaxis	represent_attributes(layer, feature)
Argumentos	• **layer** - La capa (o su ID o nombre). • **feature** - La entidad a ser evaluada.
Ejemplos	• `represent_attributes('atlas_layer', @atlas_feature)` → La representación de los atributos del objeto espacial especificado de la capa especificada.

Lectura adicional: *represent_value*

represent_value

Devuelve el valor de representación configurado para un valor de campo. Depende del tipo de widget configurado. A menudo, esto es útil para los widgets “Value Map”.

Sintaxis	represent_value(value, [fieldName]) [] marca argumentos opcionales
Argumentos	• **value** - El valor que debe resolverse. Probablemente un campo. • **fieldName** - El nombre del campo para el que debe cargarse la configuración del widget.
Ejemplos	• `represent_value("field_with_value_map")` → Descripción para valor • `represent_value('static value', 'field_name')` → Descripción para valor estático

Otras lecturas: *widget types*, *represent_attributes*

sqlite_fetch_and_increment

Administrar valores de autoincremento en bases de datos sqlite.

Los valores predeterminados de SQlite solo se pueden aplicar en la inserción y no se pueden obtener previamente.

Esto hace que sea imposible adquirir una clave primaria incrementada a través de AUTO_INCREMENT antes de crear la fila en la base de datos. Nota al margen: con postgres, esto funciona a través de la opción *evaluar valores predeterminados*.

Al agregar nuevas entidades con relaciones, es realmente bueno poder agregar hijos para un padre, mientras que el formulario de los padres todavía está abierto y, por lo tanto, la función de los padres no está comprometida.

Para evitar esta limitación, esta función se puede utilizar para administrar valores de secuencia en una tabla separada en formatos basados en sqlite como gpkg.

La tabla de secuencia se filtrará por un ID de secuencia (filter_attribute y filter_value) y el valor actual del id_field se incrementará en 1 y el valor incrementado se devolverá.

Si las columnas adicionales requieren que se especifiquen valores, el mapa default_values se puede utilizar para este propósito.

Nota

Esta función modifica la tabla sqlite de destino. Está diseñado para su uso con configuraciones de valores predeterminados para atributos.

Cuando el parámetro de la base de datos es una capa y la capa está en modo de transacción, el valor solo se recuperará una vez durante la vida útil de una transacción y se almacenará en caché y se incrementará. Esto hace que no sea seguro trabajar en la misma base de datos desde varios procesos en paralelo.

Sintaxis	sqlite_fetch_and_increment (base de datos, tabla, id_field, filter_attribute, filter_value, [default_values]) [] marca argumentos opcionales
Argumentos	• **database** - Ruta al archivo sqlite o capa de geopaquete • **table** - Nombre de la tabla que gestiona las secuencias • **id_field** - Nombre del campo que contiene el valor actual • **filter_attribute** - Nombra el campo que contiene un identificador único para esta secuencia. Debe tener un índice ÚNICO. • **filter_value** - Nombre de la secuencia a utilizar. • **default_values** - Mapa con valores predeterminados para columnas adicionales en la tabla. Los valores deben cotizarse en su totalidad. Se permiten funciones.
Ejemplos	• `sqlite_fetch_and_increment(@layer, 'sequence_table', 'last_unique_id', 'sequence_id', 'global', map('last_change', 'date(''now'')', 'user', '''' || @user_account_name || '''')) → 0` • `sqlite_fetch_and_increment(layer_property(@layer, 'path'), 'sequence_table', 'last_unique_id', 'sequence_id', 'global', map('last_change', 'date(''now'')', 'user', '''' || @user_account_name || '''')) → 0`

Otras lecturas: *Propiedades de Fuentes de Datos*, *Establecer relaciones entre varias capas*

uuid

Genera un identificador único universal (UUID) para cada fila usando el Qt QUuid::createUuid method.

Sintaxis	uuid([format="WithBraces"]) [] marca argumentos opcionales
Argumentos	• **format** - El formato, como se formateará el UUID. "WithBraces", "WithoutBraces" o "Id128".
Ejemplos	• `uuid()` → "{0bd2f60f-f157-4a6d-96af-d4ba4cb366a1}" • `uuid('WithoutBraces')` → "0bd2f60f-f157-4a6d-96af-d4ba4cb366a1" • `uuid('Id128')` → "0bd2f60ff1574a6d96afd4ba4cb366a1"

9.2.24 Relaciones

Este grupo contiene la lista de *relaciones* disponibles en el proyecto actual, con su descripción. Proporciona un acceso rápido al ID de relación para escribir una expresión (con p.ej. la función *relation_aggregate*) o personalizando un formulario.

9.2.25 Funciones de sensores

Este grupo contiene funciones para interactuar con los sensores.

sensor_data

Devuelve el último valor capturado (o valores como un mapa para sensores que informan de múltiples valores) de un sensor registrado.

Sintaxis	sensor_data(name, [expiration]) [] marca argumentos opcionales
Argumentos	• **nombre** - el nombre del sensor • **expiración** - milisegundo máximo desde el último valor capturado permitido
Ejemplos	• `sensor_data('geiger_1')` → "2000"

9.2.26 Funciones de cadena

Este grupo contiene funciones que operan sobre cadenas (por ejemplo that replace, convert to upper case).

ascii

Devuelve el código Unicode asociado con el primer carácter de una cadena.

Sintaxis	ascii(string)
Argumentos	• **string** - la cadena para convertir a código Unicode
Ejemplos	• `ascii('Q')` → 81

char

Devuelve el carácter asociado con un código Unicode.

Sintaxis	char(code)
Argumentos	• **code** - un número código unicode
Ejemplos	• `char(81)` → "Q"

concat

Concatena varias cadenas en una. Los valores NULL se convierten en cadenas vacías. Otros valores (como números) se convierten en cadenas.

Sintaxis	concat(string1, string2, …)
Argumentos	• **string** - un valor cadena
Ejemplos	• `concat('sun', 'set')` → "sunset" • `concat('a','b','c','d','e')` → "abcde" • `concat('Anno ', 1984)` → "Anno 1984" • `concat('The Wall', NULL)` → "The Wall"

Sobre concatenación de campos

También puede concatenar cadenas o valores de campo usando los operadores `||` o `+`, con algunas características especiales:

- El operador + también significa expresión resumida, por lo que si tiene un operando entero (campo o valor numérico), puede ser propenso a errores y es mejor usar los otros:

```
'My feature id is: ' + "gid" => triggers an error as gid returns an integer
```

- Cuando cualquiera de los argumentos es un valor NULL, `||` o + devolverá un valor NULL. Para devolver los otros argumentos sin tener en cuenta el valor NULL, es posible que desee utilizar la función `concat` o `array_to_string`:

```
'My feature id is: ' + NULL ==> NULL
'My feature id is: ' || NULL => NULL
concat('My feature id is: ', NULL) => 'My feature id is: '
array_to_string( array('My feature id is: ', NULL) ) => 'My feature id is: '
```

lecturas adicionales: *array_to_string*, ||, +

format

Formatee una cadena utilizando los argumentos proporcionados.

Sintaxis	format(string, arg1, arg2, …)
Argumentos	• **string** - Una cadena con marcadores de posición %1, %2, etc., para los argumentos. Los marcadores de posición se pueden repetir. El marcador de posición con el número más bajo se reemplaza por arg1, el siguiente por arg2, etc. • **arg** - cualquier tipo. Cualquier cantidad de argumentos.
Ejemplos	• `format('This %1 a %2','is', 'test')` → "This is a test" • `format('This is %2','a bit unexpected but 2 is lowest number in string','normal')` → "This is a bit unexpected but 2 is lowest number in string"

format_date

Formatea un tipo de fecha o una cadena en un formato de cadena personalizado. Utiliza cadenas de formato de fecha/hora Qt. Ver QDateTime::toString.

Sintaxis	format_date(datetime, format, [language]) [] marca argumentos opcionales
Argumentos	• **datetime** - valor de fecha, hora o fecha y hora • **format** - Plantilla de cadena utilizada para formatear la cadena.

Expresión	Salida
d	el día como número sin un cero inicial (1 a 31)
dd	el día como número con un cero inicial (01 a 31)
ddd	el nombre abreviado del día localizado (p. ej., de “Mon” a “Sun”)
dddd	el nombre largo del día localizado (por ejemplo, “Moday” a “Sunday”)
M	El mes como número sin cero inicial (1-12)
MM	el mes como número con un cero por delante (01-12)
MMM	el nombre de mes abreviado localizado (e.g. “Jan” a “Dec”)
MMMM	el nombre de mes largo localizado (e.g. “January” a “December”)
yy	el año como número de dos dígitos (00-99)
yyyy	el año como número de cuatro dígitos

Estas expresiones pueden ser usadas para la parte temporal del formato cadena:

Expresión	Salida
h	la hora sin un cero a la izquierda (0 a 23 o 1 a 12 si se muestra AM / PM)
hh	la hora con un cero a la izquierda (00 a 23 o 01 a 12 si se muestra AM / PM)
H	la hora sin un cero a la izquierda (0 a 23, incluso con pantalla AM / PM)
HH	la hora con un cero a la izquierda (00 a 23, incluso con pantalla AM / PM)
m	el minuto sin un cero a la izquierda (0 a 59)
mm	el minuto con un cero a la izquierda (00 a 59)
s	el segundo sin un cero a la izquierda (0 a 59)
ss	el segundo con un cero a la izquierda (00 a 59)
z	los milisegundos sin ceros finales (0 a 999)
zzz	los milisegundos con ceros finales (000 a 999)
AP o A	interpreta como una hora AM/PM. *AP* deber ser “AM” o “PM”.
ap o a	Interprete como hora AM / PM. *ap* debe ser “am” o “pm”.

- **language** - language (lowercase, two- or three-letter, ISO 639 language code) se utiliza para dar formato a la fecha en una cadena personalizada. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS.

Ejemplos

- `format_date('2012-05-15','dd.MM.yyyy')` → “15.05.2012”
- `format_date('2012-05-15','d MMMM yyyy','fr')` → “15 mai 2012”
- `format_date('2012-05-15','dddd')` → “Tuesday”, si la configuración regional actual es una variante en inglés
- `format_date('2012-05-15 13:54:20','dd.MM.yy')` → “15.05.12”
- `format_date('13:54:20','hh:mm AP')` → “01:54 PM”

format_number

Devuelve un número formateado con el separador de configuración regional de miles. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS. También trunca los lugares decimales al número de lugares proporcionados.

Sintaxis	format_number(number, [places=0], [language], [omit_group_separators=false], [trim_trailing_zeroes=false]) [] marca argumentos opcionales
Argumentos	• **number** - número a formatear • **places** - entero que representa el número de posiciones decimales para truncar la cadena. • **language** - language (lowercase, two- or three-letter, ISO 639 language code) se utiliza para formatear el número en una cadena. De forma predeterminada, se utiliza la configuración regional actual del usuario de QGIS. • **omit_group_separators** - si se establece en true, los separadores de grupo no se incluirán en la cadena • **trim_trailing_zeroes** - si se establece en true, los ceros que siguen al punto decimal se eliminarán de la cadena.
Ejemplos	• `format_number(10000000.332,2)` → "10,000,000.33" si por ej. la configuración regional actual es una variante en inglés • `format_number(10000000.332,2,'fr')` → "10 000 000,33"

left

Devuelve una subcadena que contiene los *n* caracteres más a la izquierda de la cadena.

Sintaxis	left(string, length)
Argumentos	• **string** - una cadena • **length** - entero. El número de caracteres de la izquierda de la cadena para devolver.
Ejemplos	• `left('Hello World',5)` → "Hello"

Lectura adicional: expression_function_String_right`lectura adicional: :ref:`expression_function_Arrays_array_to_string, ||, +

length

Devuelve el número de caracteres de una cadena o la longitud de una cadena de líneas geométricas.

Variante de cadena

Devuelve el número de caracteres de una cadena.

Sintaxis	length(string)
Argumentos	• **string** - cadena de la que contar caracteres
Ejemplos	• `length('hello')` → 5

Variante Geometría

Calcula la longitud de un objeto de línea geométrica. Los cálculos son siempre planimétricos en el Sistema de referencia espacial (SRS) de esta geometría, y las unidades de la longitud devuelta coincidirán con las unidades del SRS. Esto difiere de los cálculos realizados por la función $length, que realizará cálculos elipsoidales basados en la configuración de unidad de distancia y elipsoide del proyecto.

Sintaxis	length(geometry)
Argumentos	• **geometry** - objeto geométrico lineal
Ejemplos	• `length(geom_from_wkt('LINESTRING(0 0, 4 0)'))` → 4.0

lower

Convierte una cadena en minúsculas.

Sintaxis	lower(string)
Argumentos	• **string** - la cadena para convertir a minúsculas
Ejemplos	• `lower('HELLO World')` → “hello world”

Lectura adicional: *upper*, *title*

lpad

Devuelve una cadena rellenada a la izquierda hasta el ancho especificado, utilizando un carácter de relleno. Si el ancho objetivo es menor que la longitud de la cadena, la cadena se trunca.

Sintaxis	lpad(string, width, fill)
Argumentos	• **string** - cadena a rellenar • **width** - longitud de nueva cadena • **fill** - carácter con el que rellenar el espacio restante
Ejemplos	• `lpad('Hello', 10, 'x')` → “xxxxxHello” • `lpad('Hello', 3, 'x')` → “Hel”

Lectura adicional: *rpad*

ltrim

Elimina la cadena más larga que contenga sólo los caracteres especificados (un espacio por defecto) del inicio de la cadena.

Sintaxis	ltrim(string, [characters=" "]) [] marca argumentos opcionales
Argumentos	• **string** - cadena a recortar • **caracteres** - caracteres a recortar
Ejemplos	• `ltrim(' hola mundo ')` → "hola mundo " • `ltrim('zzzytest', 'xyz')` → "test"

Lectura adicional: *rtrim*, *trim*

regexp_match

Devuelve la primera posición coincidente que coincide con una expresión regular dentro de una cadena Unicode, o 0 si no se encuentra la subcadena.

Sintaxis	regexp_match(input_string, regex)
Argumentos	• **input_string** - la cadena para probar con la expresión regular • **regex** - La expresión regular con la que realizar la prueba. Los caracteres de barra invertida deben tener doble escape (e.g., «\\s» para que coincida con un carácter de espacio en blanco o «\\b» para que coincida con el límite de una palabra).
Ejemplos	• `regexp_match('QGIS ROCKS','\\sROCKS')` → 5 • `regexp_match('Budač','udač\\b')` → 2

Lectura adicional: *strpos*

regexp_replace

Devuelve una cadena con la expresión regular proporcionada reemplazada.

Sintaxis	regexp_replace(input_string, regex, replacement)
Argumentos	• **input_string** - la cadena en la que reemplazar coincidencias • **regex** - La expresión regular que se reemplazará. Los caracteres de barra invertida deben tener doble escape (por ejemplo, «\\s» para que coincida con un carácter de espacio en blanco). • **replacement** - La cadena que reemplazará las apariciones coincidentes de la expresión regular proporcionada. Los grupos capturados se pueden insertar en la cadena de reemplazo usando \\1, \\2, etc.
Ejemplos	• `regexp_replace('QGIS SHOULD ROCK','\\sSHOULD\\s',' DOES ')` → "QGIS DOES ROCK" • `regexp_replace('ABC123','\\d+','')` → "ABC" • `regexp_replace('my name is John','(.*) is (.*)','\\2 is \\1')` → "John is my name"

Lectura adicional: *replace*, *array_replace*

regexp_substr

Devuelve la parte de una cadena que coincide con una expresión regular proporcionada.

Sintaxis	regexp_substr(input_string, regex)
Argumentos	• **input_string** - La cadena en la cuál encontrar coincidencias • **regex** - La expresión regular con la que comparar. Los caracteres de barra invertida deben tener doble escape (por ejemplo, «\\s» para que coincida con un carácter de espacio en blanco).
Ejemplos	• `regexp_substr('abc123','(\\d+)')` → "123"

Lectura adicional: *substr*, *regexp_matches*

repeat

Repeats a string a specified number of times.

Sintaxis	repeat(string, number)
Argumentos	• **string** - string to repeat • **number** - number of times to repeat string
Ejemplos	• `repeat('Hello', 3)` → "HelloHelloHello"

replace

Devuelve una cadena con la cadena, matriz o mapa de cadenas reemplazados.

String & array variant

Devuelve una cadena con la cadena proporcionada o la matriz de cadenas reemplazada por una cadena o una matriz de cadenas.

Sintaxis	replace(string, before, after)
Argumentos	• **string** - la cadena entrante • **before** - la cadena o matriz de cadenas a reemplazar • **after** - la cadena o matriz de cadenas para usar como reemplazo
Ejemplos	• `replace('QGIS SHOULD ROCK','SHOULD','DOES')` → "QGIS DOES ROCK" • `replace('QGIS ABC',array('A','B','C'),array('X','Y','Z'))` → "QGIS XYZ" • `replace('QGIS',array('Q','S'),'')` → "GI"

Map variant

Devuelve una cadena con las claves de mapa proporcionadas reemplazadas por valores emparejados. Las claves de mapa más largas se evalúan primero.

Sintaxis	replace(string, map)
Argumentos	• **string** - la cadena entrante • **map** - el mapa que contiene claves y valores
Ejemplos	• `replace('APP SHOULD ROCK',map('APP','QGIS','SHOULD','DOES'))` → "QGIS DOES ROCK" • `replace('forty two',map('for','4','two','2','forty two','42'))` → "42"

Más información: *regexp_replace*, *array_replace*

reverse

Reverses the direction of a line string or reverses a string of text.

Variante de cadena

Reverses the order of characters in a string.

Sintaxis	reverse(string)
Argumentos	• **string** - string to reverse
Ejemplos	• `reverse('hello')` → "olleh"

Variante Geometría

Invierte la dirección de una cadena lineal invirtiendo el orden de sus vértices.

Sintaxis	reverse(geometry)
Argumentos	• **geometry** - una geometría
Ejemplos	• `geom_to_wkt(reverse(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)')))` → "LINESTRING(2 2, 1 1, 0 0)"

right

Devuelve una subcadena que contiene los *n* caracteres del extremo derecho de la cadena.

Sintaxis	right(string, length)
Argumentos	• **string** - una cadena • **length** - entero. El número de caracteres desde la derecha de la cadena a devolver.
Ejemplos	• `right('Hello World',5)` → "World"

Más información: *left*

rpad

Devuelve una cadena rellenada a la derecha hasta el ancho especificado, utilizando un carácter de relleno. Si el ancho objetivo es menor que la longitud de la cadena, la cadena se trunca.

Sintaxis	rpad(string, width, fill)
Argumentos	• **string** - cadena a rellenar • **width** - longitud de nueva cadena • **fill** - carácter con el que rellenar el espacio restante
Ejemplos	• `rpad('Hello', 10, 'x')` → "Helloxxxxx" • `rpad('Hello', 3, 'x')` → "Hel"

Más información: *lpad*

rtrim

Elimina la cadena más larga que contenga sólo los caracteres especificados (un espacio por predeterminado) del final de la cadena.

Sintaxis	rtrim(string, [characters=" "]) [] marca argumentos opcionales
Argumentos	• **string** - cadena a recortar • **caracteres** - caracteres a recortar
Ejemplos	• `rtrim(' hola mundo ')` → " hola mundo" • `rtrim('testxxzx', 'xyz')` → "test"

Lectura adicional: *ltrim*, *trim*

strpos

Devuelve la primera posición coincidente de una subcadena dentro de otra cadena, o 0 si no se encuentra la subcadena.

Sintaxis	strpos(haystack, needle)
Argumentos	• **haystack** - cadena que se va a buscar • **needle** - cadena a buscar
Ejemplos	• `strpos('HELLO WORLD','WORLD')` → 7 • `strpos('HELLO WORLD','GOODBYE')` → 0

Otras lecturas: *regexp_match*

substr

Devuelve una parte de una cadena.

Sintaxis	substr(string, start, [length]) [] marca argumentos opcionales
Argumentos	• **string** - la cadena entrante entrante • **start** - número entero que representa la posición de inicio para extraer comenzando con 1; si el inicio es negativo, la cadena de retorno comenzará al final de la cadena menos el valor de inicio • **length** - entero que representa la longitud de la cadena a extraer; si la longitud es negativa, la cadena de retorno omitirá la longitud dada de caracteres desde el final de la cadena
Ejemplos	• `substr('HELLO WORLD',3,5)` → "LLO W" • `substr('HELLO WORLD',6)` → " WORLD" • `substr('HELLO WORLD',-5)` → "WORLD" • `substr('HELLO',3,-1)` → "LL" • `substr('HELLO WORLD',-5,2)` → "WO" • `substr('HELLO WORLD',-5,-1)` → "WORL"

Más información: *regexp_substr*, *regexp_matches*

title

Convierte todas las palabras de una cadena a tipo de título (todas las palabras en minúsculas con mayúscula inicial).

Sintaxis	title(string)
Argumentos	• **string** - la cadena para convertir a tipo título
Ejemplos	• `title('hello WOrld')` → "Hello World"

Más información: *lower*, *upper*

to_string

Converts a number to string. The conversion is not locale-aware, see "format_number" for a locale-aware alternative.

Sintaxis	to_string(number)
Argumentos	• **number** - Entero o valor real. El número a convertir a cadena.
Ejemplos	• `to_string(1.23)` → "1.23"

Más información: *format_number*

trim

Elimina todos los espacios en blanco iniciales y finales (espacios, tabulaciones, etc.) de una cadena.

Sintaxis	trim(string)
Argumentos	• **string** - cadena a recortar
Ejemplos	• `trim('  hello world  ')` → "hello world"

Lectura adicional: *ltrim*, *rtrim*

upper

Convierte una cadena a letras mayúsculas.

Sintaxis	upper(string)
Argumentos	• **string** - la cadena para convertir a mayúsculas
Ejemplos	• `upper('hello WOrld')` → "HELLO WORLD"

Más información: *lower*, *title*

wordwrap

Devuelve una cadena ajustada a un número máximo/mínimo de carácteres.

Sintaxis	wordwrap(string, wrap_length, [delimiter_string]) [] marca argumentos opcionales
Argumentos	• **string** - la cadena a ajustar • **wrap_length** - un entero. Si wrap_length es positivo, el número representa el número máximo ideal de caracteres a ajustar; si es negativo, el número representa el número mínimo de caracteres a ajustar. • **delimiter_string** - Cadena de delimitador opcional para ajustar a una nueva línea.
Ejemplos	• `wordwrap('UNIVERSITY OF QGIS',13)` → "UNIVERSITY OF QGIS" • `wordwrap('UNIVERSITY OF QGIS',-3)` → "UNIVERSITY OF QGIS"

9.2.27 Expresiones de Usuario

Este grupo contiene las expresiones guardadas como *user expressions*.

9.2.28 Variables

Este grupo contiene variables dinámicas relacionadas con la aplicación, el archivo del proyecto y otras configuraciones. La disponibilidad de variables depende del contexto:

- desde el diálogo  Select by expression
- desde el diálogo  Field calculator
- desde el diálogo de propiedades de la capa
- desde el diseñador de impresión

Para usar estas variables en una expresión, deben estar precedidas por el carácter `@`` (p.ej, ``@row_number`).

Variable	Descripción
algorithm_id	El ID único de un algoritmo
animation_end_time	Fin del rango de tiempo sobre todo el tiempo de la animación (como valor de fecha y hora)
animation_interval	Duración del rango de tiempo sobre todo el tiempo de animación (como un valor de intervalo)
animation_start_time	Inicio del rango de tiempo sobre todo el tiempo de la animación (como valor de fecha y hora)
atlas_feature	La entidad actual del atlas (como objeto de entidad)
atlas_featureid	La ID de la entidad actual del atlas
atlas_featurenumber	El número de entidades actuales del atlas en el diseño
atlas_filename	El nombre de archivo del atlas actual
atlas_geometry	La geometría de la entidad actual del atlas
atlas_layerid	La ID de la capa de cobertura del atlas actual
atlas_layername	El nombre de la capa de cobertura del atlas actual
atlas_pagename	El nombre de la página del atlas actual
atlas_totalfeatures	El número total de entidades en el atlas
band	El número de la banda en la capa raster
band_description	La descripción de la banda en la capa raster

continúe en la próxima página

Tabla 9.1 – proviene de la página anterior

Variable	Descripción
band_name	El nombre de la banda en la capa raster
canvas_cursor_point	La última posición del cursos en el lienzo en coordenadas geográficas del proyecto
cluster_color	El color de los símbolos dentro de un grupo, o NULL si los símbolos tienen colores mezclados
cluster_size	La cantidad de símbolos contenidos en un grupo
current_feature	La entidad que se está editando actualmente en el formulario de atributo o la fila de la tabla
current_geometry	La geometría de la entidad que se está editando actualmente en el formulario o la fila de la tabla
current_parent_feature	representa la función que se está editando actualmente en el formulario principal. Solo se puede utilizar en un contexto de formulario incrustado.
current_parent_geometry	representa la geometría de la función que se está editando actualmente en el formulario principal. Solo se puede utilizar en un contexto de formulario incrustado.
form_mode	Para qué se usa el formulario, como AddFeatureMode, SingleEditMode, MultiEditMode, SearchMode, AggregateSearchMode o IdentifyMode como cadena.
feature	La característica actual que se está evaluando. Puede utilizarse con la función «attribute» para evaluar los valores de los atributos de la característica actual.
frame_duration	Duración temporal de cada cuadro de animación (como valor de intervalo)
frame_number	Número de fotograma actual durante la reproducción de la animación
frame_rate	Número de fotogramas por segundo durante la reproducción de la animación
fullextent_maxx	Valor x máximo de la extensión del lienzo completo (incluidas todas las capas)
fullextent_maxy	Valor y máximo de la extensión del lienzo completo (incluidas todas las capas)
fullextent_minx	Valor x mínimo de la extensión del lienzo completo (incluidas todas las capas)
fullextent_miny	Valor y mínimo de la extensión del lienzo completo (incluidas todas las capas)
geometry	La geometría del objeto espacial actual que se está evaluando
geometry_part_count	El número de piezas en la geometría de la entidad renderizada.
geometry_part_num	El número de partes de la geometría actual para la entidad que se está renderizando
geometry_point_count	El número de puntos en la parte de la geometría renderizada.
geometry_point_num	El número de puntos actual en la parte de la geometría renderizada
geometry_ring_num	Número de anillo de geometría actual para la entidad que se está representando (solo para entidades poligonales). El anillo exterior tiene un valor de 0.
grid_axis	El eje de anotación de la cuadrícula actual (p. Ej., “X” para la longitud, “y” para la latitud)
grid_number	El valor de anotación de cuadrícula actual
id	ID de la función que se está evaluando
item_id	El ID de usuario del elemento del diseño (no necesariamente único)
item_uuid	El ID único del elemento de diseño
layer	La capa actual
layer_crs	El ID de autoridad del sistema de referencia de coordenadas de la capa actual
layer_crs_ellipsoid	ID de la autoridad elipsoidal de el SRC de la capa actual
layer_cursor_point	Geometría del punto bajo la posición del ratón en el lienzo del mapa (o la posición GetFeatureInfo en el contexto de QGIS Server), en el SRC de la capa activa.
layer_id	La ID de la capa actual
layer_ids	Las ID de todas las capas del mapa en el proyecto actual como una lista
layer_name	El nombre de la capa actual
layer_vertical_crs	El identificador del sistema de referencia de coordenadas verticales de la capa (por ejemplo, “EPSG:5703”).
layer_vertical_crs_definition	La definición completa del sistema de referencia de coordenadas verticales de la capa.
layer_vertical_crs_description	El nombre del sistema de referencia de coordenadas verticales de la capa.

continúe en la próxima página

Tabla 9.1 – proviene de la página anterior

Variable	Descripción
layer_vertical_crs_wkt	La definición WKT del sistema de referencia de coordenadas verticales de la capa actual.
layers	Todas las capas del mapa en el proyecto actual como una lista
layout_dpi	La resolución de la composición (DPI)
layout_name	El nombre del diseño
layout_numpages	El número de páginas en el diseño
layout_page	El número de página del elemento actual en el diseño
layout_pageheight	La altura de la página activa en el diseño (en mm para tamaños de papel estándar o cualquier unidad que se haya utilizado para el tamaño de papel personalizado)
layout_pageoffsets	Matriz de coordenadas Y de la parte superior de cada página. Permite posicionar dinámicamente elementos en páginas en un contexto donde los tamaños de página pueden cambiar
layout_pagewidth	El ancho de página activo en el diseño (en mm para tamaños de papel estándar, o cualquier unidad que se haya utilizado para el tamaño de papel personalizado)
legend_column_count	El número de columnas en la leyenda
legend_filter_by_map	Indica si el contenido de la leyenda es filtrada por el mapa
legend_filter_out_atlas	indica si el atlas es filtrado fuera de la leyenda
legend_split_layers	Indica si las capas pueden ser cortadas en la leyenda
legend_title	El título de la leyenda
legend_wrap_string	El carácter(es) usado(s) para el texto de la leyenda
map_crs	El sistema de coordenadas de Referencia del mapa actual
map_crs_acronym	El acrónimo del sistema de referencia de coordenadas del mapa actual
map_crs_definition	La definición completa del sistema de referencia de coordenadas del mapa actual
map_crs_description	El nombre del sistema de referencia de coordenadas del mapa actual
map_crs_ellipsoid	El acrónimo del elipsoide del sistema de referencia de coordenadas del mapa actual
map_crs_proj4	La definición de Proj4 del sistema de referencia de coordenadas del mapa actual
map_crs_projection	El nombre descriptivo del método de proyección utilizado por el sistema de referencia de coordenadas del mapa (por ejemplo, “Albers Equal Area”)
map_crs_wkt	La definición WKT del sistema de referencia de coordenadas del mapa actual
map_end_time	El final del rango de tiempo del mapa temporal (como valor de fecha y hora)
map_extent	La geometría que representa la extensión actual del mapa.
map_extent_center	La entidad de puntos en el centro del mapa
map_extent_height	La altura actual del mapa
map_extent_width	El ancho actual del mapa
map_id	El ID del destino actual del mapa. Este será “lienzo” para las representaciones de lienzo y el ID del elemento para las representaciones del mapa de diseño
map_interval	La duración del rango de tiempo del mapa temporal (como un valor de intervalo)
map_layer_ids	La lista de IDs de capas de mapa visibles en el mapa
map_layers	La lista de capas de mapa visibles en el mapa.
map_rotation	La rotación actual del mapa
map_scale	La escala actual del mapa
map_start_time	El inicio del rango de tiempo del mapa temporal (como un valor de fecha y hora)
map_units	Las unidades de medida del mapa
map_z_range_lower	Elevación más baja del rango de elevación del mapa
map_z_range_upper	Elevación superior del rango de elevación del mapa
model_path	Ruta completa (incluido el nombre del archivo) del modelo actual (o ruta del proyecto si el modelo está incrustado en un proyecto).
model_folder	Carpeta que contiene el modelo actual (o carpeta del proyecto si el modelo está incrustado en un proyecto).
model_name	Nombre del modelo actual
model_group	Grupo del modelo actual

continúe en la próxima página

Tabla 9.1 – proviene de la página anterior

Variable	Descripción
notification_message	Contenido del mensaje de notificación enviado por el proveedor (disponible solo para acciones activadas por notificaciones del proveedor).
parent	Se refiere a la entidad actual en la capa principal, que proporciona acceso a sus atributos y geometría al filtrar una función *aggregate*
plot_axis	El eje de coordenadas asociado, por ejemplo, “x” o “y”.
plot_axis_value	El valor actual del eje del gráfico.
project_abstract	El resumen del proyecto, tomado de los metadatos del proyecto.
project_area_units	La unidad de área del proyecto actual, que se utiliza al calcular áreas de geometrías.
project_author	El autor del proyecto, tomado de los metadatos del proyecto.
project_basename	El nombre base del nombre de archivo del proyecto actual (sin ruta ni extensión)
project_creation_date	La fecha de creación del proyecto, tomada de los metadatos del proyecto.
project_crs	Identificador del sistema de referencia de coordenadas del proyecto (por ejemplo, “EPSG:4326”).
project_crs_arconym	El acrónimo del sistema de referencia de coordenadas del proyecto.
project_crs_definition	La definición completa del sistema de referencia de coordenadas del proyecto.
project_crs_description	La descripción del sistema de referencia de coordenadas del proyecto.
project_crs_ellipsoid	El elipsoide del sistema de referencia de coordenadas del proyecto
project_crs_proj4	La representación Proj4 del sistema de referencia de coordenadas del proyecto
project_crs_wkt	La representación WKT (texto bien conocido) del sistema de referencia de coordenadas del proyecto
project_distance_units	La unidad de distancia para el proyecto actual, utilizada al calcular longitudes de geometrías y distancias.
project_ellipsoid	El nombre del elipsoide del proyecto actual, utilizado al calcular áreas geodésicas o longitudes de geometrías.
project_filename	El nombre de archivo del proyecto actual
project_folder	La carpeta del proyecto actual
project_home	La ruta a casa del proyecto actual
project_identifier	El identificador del proyecto, tomado de los metadatos del proyecto.
project_keywords	Las palabras clave del proyecto, tomadas de los metadatos del proyecto.
project_last_saved	Fecha / hora de la última vez que se guardó el proyecto.
project_path	La ruta completa (incluido el nombre del archivo) del proyecto actual
project_title	El título del proyecto actual
project_units	Las unidades del CRS del proyecto
project_vertical_crs	Identificador del sistema de referencia de coordenadas verticales del proyecto (por ejemplo, “EPSG:5703”).
project_vertical_crs_definition	La definición completa del sistema de referencia de coordenadas verticales del proyecto.
project_vertical_crs_description	La descripción del sistema de referencia de coordenadas verticales del proyecto
project_vertical_crs_wkt	La representación WKT (texto bien conocido) del sistema de referencia de coordenadas verticales del proyecto.
qgis_locale	El idioma actual de QGIS
qgis_os_name	El nombre del sistema operativo actual, por ejemplo, “windows”, “linux” u “osx”
qgis_platform	La plataforma QGIS, por ejemplo, “escritorio” o “servidor”
qgis_release_name	El nombre de la versión actual de QGIS
qgis_short_version	La cadena corta de la versión actual de QGIS
qgis_version	La cadena de versión actual de QGIS
qgis_version_no	El número de versión actual de QGIS
row_number	Guarda el número de la fila actual.
snapping_results	Da acceso a los resultados de ajuste mientras se digitaliza una función (solo disponible en agregar función)
scale_value	El valor de distancia de la barra de escala actual
selected_file_path	Ruta de archivo seleccionada desde el selector de widget de archivo al cargar un archivo con un sistema de almacenamiento externo

continúe en la próxima página

Tabla 9.1 – proviene de la página anterior

Variable	Descripción
symbol_angle	El ángulo del símbolo utilizado para representar la entidad (válido solo para símbolos de marcador)
symbol_color	El color del símbolo utilizado para representar la entidad.
symbol_count	El número de entidades representadas por el símbolo (en la leyenda del diseño)
symbol_frame	El número de marco (sólo para símbolos animados)
symbol_id	El ID interno del símbolo (en la leyenda del diseño)
symbol_label	La etiqueta del símbolo (ya sea una etiqueta definida por el usuario o la etiqueta predeterminada generada automáticamente, en la leyenda del diseño)
symbol_layer_count	Número total de capas de símbolo en el símbolo
symbol_layer_index	Índice de capa de símbolo actual
symbol_marker_column	Número de columna para el marcador (válido solo para rellenos de patrón de puntos).
symbol_marker_row	Número de fila para el marcador (válido solo para rellenos de patrón de puntos).
user_account_name	El nombre de la cuenta del sistema operativo del usuario actual
user_full_name	El nombre de usuario del sistema operativo del usuario actual
value	El valor actual
vector_tile_zoom	Nivel de zoom de mosaico vectorial exacto del mapa que se está representando (derivado de la escala del mapa actual). Normalmente en el intervalo [0, 20]. A diferencia de @zoom_level, esta variable es un valor de punto flotante que se puede usar para interpolar valores entre dos niveles de zoom enteros.
with_variable	Permite configurar una variable para usar dentro de una expresión y evitar recalcular el mismo valor repetidamente
zoom_level	Nivel de zoom de mosaico vectorial del mapa que se está representando (derivado de la escala del mapa actual). Normalmente en el intervalo [0, 20].

Algún ejemplo:

- Devuelve la coordenada X del centro de un elemento de mapa en el diseño:

```
x( map_get( item_variables( 'map1'), 'map_extent_center' ) )
```

- Devuelve, para cada entidad de la capa actual, el número de entidades aeroportuarias superpuestas:

```
aggregate( layer:='airport', aggregate:='count', expression:="code",
               filter:=intersects( $geometry, geometry( @parent ) ) )
```

- Obtener la object_id del primer punto ajustado de una línea:

```
with_variable(
  'first_snapped_point',
  array_first( @snapping_results ),
  attribute(
    get_feature_by_id(
      map_get( @first_snapped_point, 'layer' ),
      map_get( @first_snapped_point, 'feature_id' )
    ),
    'object_id'
  )
)
```

9.2.29 Funciones recientes

Este grupo contiene funciones utilizadas recientemente. Dependiendo del contexto de su uso (selección de características, calculadora de campo, genérico), las expresiones aplicadas recientemente se agregan a la lista correspondiente (hasta diez expresiones), ordenadas de más a menos recientes. Esto facilita recuperar y volver a aplicar rápidamente expresiones utilizadas anteriormente.

CAPÍTULO 10

La librería Style

10.1 El administrador de Estilo

10.1.1 El diálogo del Administrador de Estilo

El :guilabel: *Administrador de Estilo* es el lugar donde puede administrar y crear elementos de estilo genéricos. Estos son símbolos, rampas de color, formatos de texto o configuraciones de etiquetas que se pueden utilizar para simbolizar entidades, capas o diseños de impresión. Se almacenan en la base de datos `symbology-style.db` en el activo *user profile* y compartido con todos los archivos de proyecto abiertos con ese perfil. Los elementos de estilo también se pueden compartir con otros gracias a las capacidades de exportación/importación del diálogo *Administrador de Estilo*.

Puede abrir este cuadro de diálogo no modal:

- desde el menú *Ajustes* ►  *Administrador de Estilo…*
- con el botón  desde la barra de herramientas del Proyecto
- o con el botón  desde un menú vector *Propiedades de Capa* ► (si bien *configuring a symbol* o *formatting a text*).

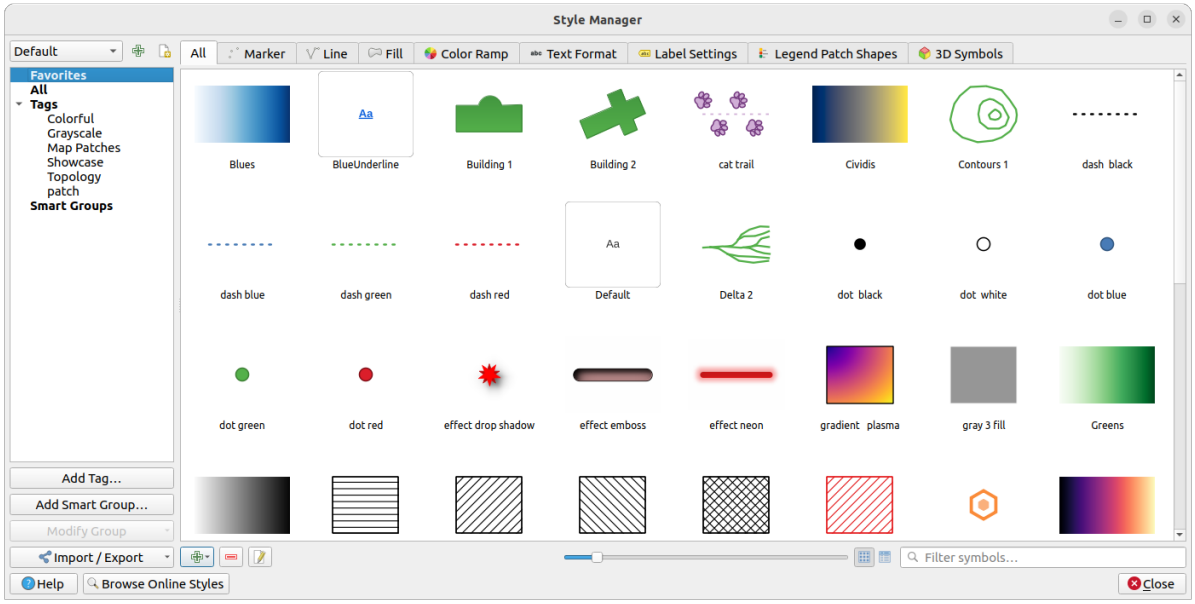


Figura 10.1: El administrador de Estilo

Organizando elementos de estilo

En la esquina superior izquierda del cuadro de diálogo *Administrador de estilo*, dentro del menú desplegable, puede elegir la base de datos de estilos a la que desea conectarse. Si elige *Predeterminado* se conectará a la base de datos de estilos por defecto donde podrá encontrar todos los elementos de estilo por defecto disponibles y los que haya guardado en esta base de datos. Si elige *Estilos del proyecto* se conectará a la base de datos de estilos del proyecto donde podrá encontrar sólo los símbolos que haya guardado en esta base de datos. Si ha creado más bases de datos de estilos, aparecerán en el menú desplegable. También hay opciones para Añadir base de datos de estilos existente al proyecto y Crear nueva base de datos de estilos (ver más *Base de datos de estilos*).

Para cada base de datos de estilos, puede organizar los elementos en distintas categorías, enumeradas en el panel de la izquierda:

- **Favoritos**: mostrado de manera predeterminada al configurar un elemento, muestra un conjunto extensible de elementos;
- **Todo**: enumera todos los elementos disponibles para el tipo activo;
- **Etiquetas**: muestra una lista de etiquetas que puede usar para identificar los elementos. Un artículo se puede etiquetar más de una vez. Seleccione una etiqueta en la lista y las pestañas se actualizan para mostrar solo los elementos que le pertenecen. Para crear una nueva etiqueta que luego pueda adjuntar a un conjunto de elementos, use el botón *Agregar etiqueta …* o seleccione el *Agregar etiqueta …* desde cualquier menú contextual de etiquetas;
- **Grupo inteligente**: un grupo inteligente recupera dinámicamente sus símbolos de acuerdo con las condiciones establecidas (ver p.ej. , Figura 10.2). Haga click en el botón *Agregar grupo inteligente…* para crear grupos inteligentes. El cuadro de diálogo le permite ingresar una expresión para filtrar los elementos a seleccionar (tiene una etiqueta particular, tiene una cadena en su nombre, etc.). Cualquier símbolo, rampa de color, formato de texto o configuración de etiqueta que satisfaga las condiciones ingresadas se agrega automáticamente al grupo inteligente.

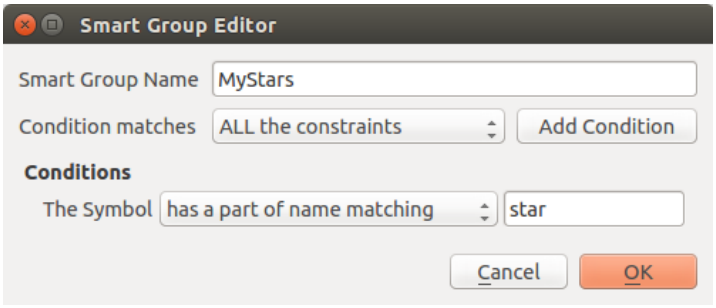


Figura 10.2: Creando un grupo inteligente

Las etiquetas y los grupos inteligentes no son mutuamente excluyentes: son simplemente dos formas diferentes de organizar sus elementos de estilo. A diferencia de los grupos inteligentes que obtienen automáticamente sus elementos pertenecientes en función de las restricciones de entrada, el usuario llena las etiquetas. Para editar cualquiera de esas categorías, puede:

- seleccione los elementos, haga click con el botón derecho y elija *Agregar a etiqueta* -> y luego seleccione el nombre de la etiqueta o cree una nueva etiqueta;
- seleccione la etiqueta y presione *Modificar grupo …* -> *Adjuntar etiqueta seleccionada a símbolos*. Aparece una casilla de verificación junto a cada elemento para ayudarlo a seleccionarlo o anular su selección. Cuando finalice la selección, presione: menuselection: *Modificar grupo …* -> *Finalizar etiquetado*.
- seleccione el grupo inteligente, presione *Modificar grupo …* -> *Editar grupo inteligente …* y configure un nuevo conjunto de restricciones en el diálogo: guilabel: *Editor de grupo inteligente*. Esta opción también está disponible en el menú contextual del grupo inteligente.

Para eliminar una etiqueta o un grupo inteligente, haga click con el botón derecho en él y seleccione el botón *Eliminar*. Tenga en cuenta que esto no elimina los elementos agrupados en la categoría.

El diálogo *Administrador de Estilo* muestra en su centro un marco con elementos previsualizados organizados en pestañas:

- *Todo* para una colección completa de símbolos de puntos, lineales y de superficie y configuraciones de etiquetas, así como rampas de colores y formatos de texto predefinidos;
- *Marcador* solo para símbolos de puntos;
- *Línea* solo para símbolos lineales;
- *Relleno* solo para símbolos de superficie;
- *Rampa de color*;
- abc *Formato de Texto* para administrar *text formats*, que almacenan la fuente, el color, los búferes, las sombras y los fondos de los textos (es decir, todas las partes de formato de la configuración de la etiqueta, que por ejemplo se pueden usar en diseños);
- *Ajustes de etiquetado* para administrar *label settings*, que incluyen los formatos de texto y algunas configuraciones específicas de tipo de capa, como ubicación de etiquetas, prioridad, rótulos, renderizado …
- *Formas de parches de leyenda* para gestionar las formas de parche de leyenda personalizadas, que incluyen las geometrías *Marcador*, *Línea* y *Rellenar*.
- *Símbolos 3D* para configurar con *propiedades 3D* (extrusión, sombreado, altitud, …) para los objetos a representar en una *vista de Mapa 3D*

Puede ordenar los Estilos en *Vista de iconos* o en *Vista de lista* en la parte inferior derecha. En ambas vistas la información de herramienta muestra una instancia más grande del estilo. El control deslizante del tamaño de las

miniaturas a la izquierda de los iconos le ayuda a ajustar el tamaño real de las miniaturas en el diálogo, ¡para una mejor previsualización de los símbolos!

Agregar, editar o eliminar un elemento

Como se vio anteriormente, los elementos de estilo se enumeran en diferentes pestañas cuyo contenido depende de la categoría activa (etiqueta, grupo inteligente, favoritos …). Cuando una pestaña está habilitada, puede:

- Agregar nuevos elementos: presione el botón  Agregar elemento y configurar el elemento de la siguiente manera *symbols*, *color ramps* o *text format and label*
- Modificar un elemento existente: seleccione un elemento y presione el botón  Editar elemento y configurar como se mencionó anteriormente.
- Eliminar elementos existentes: para eliminar un elemento que ya no necesita, selecciónelo y haga click en  : sup:*Eliminar elemento* (también disponible haciendo click derecho). El elemento se eliminará de la base de datos local.

Tenga en cuenta que la pestaña *Todos* proporciona acceso a estas opciones para cada tipo de elemento.

Hacer click derecho sobre una selección de elementos también le permite:

- *Añadir a Favoritos*;
- *Borrar de Favoritos*;
- *Añadir a etiqueta* ► y seleccione la etiqueta apropiada o cree una nueva para usar; se verifican las etiquetas asignadas actualmente;
- *Limpiar Etiquetas*: separar los símbolos de cualquier etiqueta;
- *Borrar elemento(s)*;
- *Editar elemento*: se aplica al elemento sobre el que hace click con el botón derecho;
- *Copiar elemento*;
- *Pegar elemento…*: pegar en una de las categorías del administrador de estilo o en cualquier otro lugar de QGIS (botones de símbolo o color)
- *Exportar Símbolo(s) Seleccionados como PNG…* (solo disponible con símbolos);
- *Exportar Símbolo(s) Seleccionados como SVG…* (solo disponible con símbolos);

Compartiendo elementos de estilo

La herramienta  *Importar/Exportar*, en la parte inferior izquierda del cuadro de diálogo Administrador de Estilo, ofrece opciones para compartir fácilmente símbolos, rampas de color, formatos de texto y configuraciones de etiquetas con otros. Estas opciones también están disponibles haciendo clic derecho sobre los elementos.

Esportando elementos

Puede exportar un conjunto de elementos a un archivo `.XML`:

1. Expanda el menú desplegable  *Importar/Exportar* y seleccione  *Exportar elemento(s)…*
2. Elija los elementos que le gustaría integrar. La selección se puede hacer con el ratón o usando una etiqueta o un grupo previamente establecido.
3. Presione *Exportar* cuando esté listo. Se le pedirá que indique el destino del archivo guardado. El formato XML genera un solo archivo que contiene todos los elementos seleccionados. Este archivo se puede importar en la biblioteca de estilos de otro usuario.

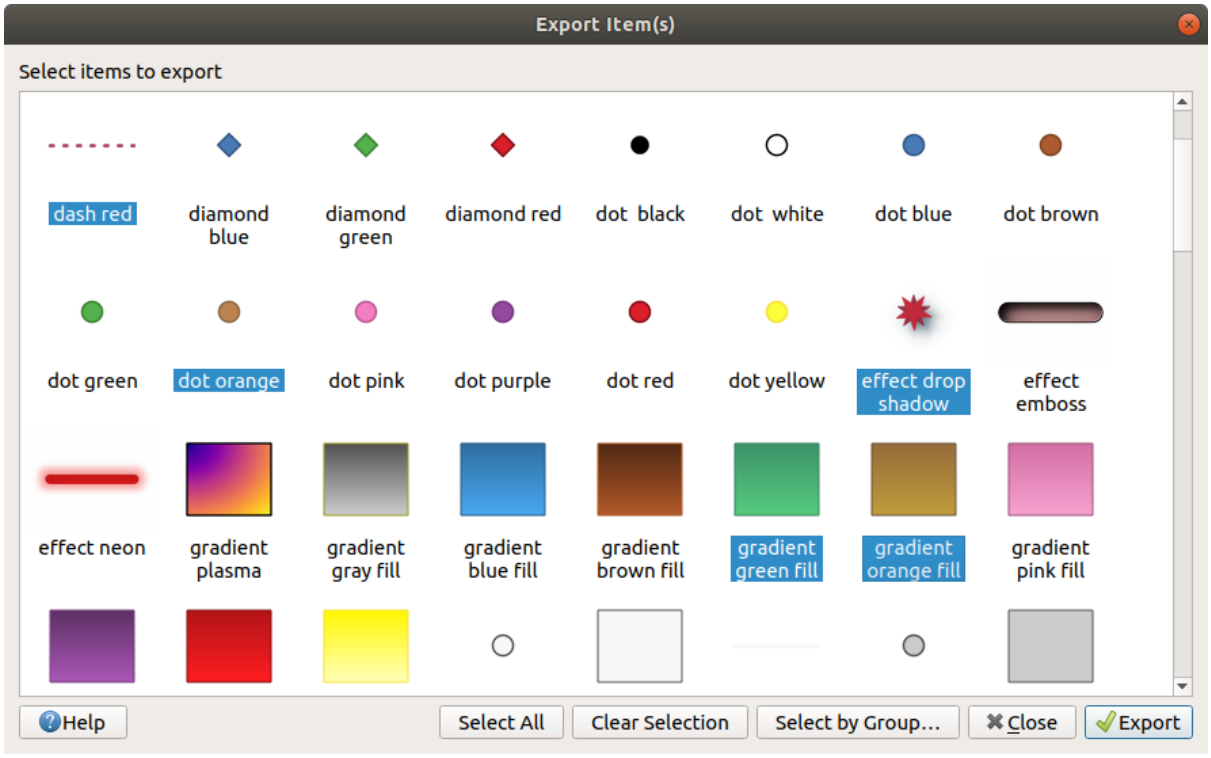


Figura 10.3: Exportando elementos de estilo

Cuando se seleccionan símbolos, también puede exportarlos a `.PNG` o `.SVG`. La exportación a `.PNG` o `.SVG` (ambos no disponibles para otros tipos de elementos de estilo) crea un archivo para cada símbolo seleccionado en una carpeta determinada. La carpeta SVG se puede agregar a *Trayectorias SVG* en el menú *Configuración -> Opciones -> Sistema* de otro usuario, lo que le permite el acceso directo a todos estos símbolos.

Importando elementos

Puede ampliar su librería de estilo importando nuevos elementos:

1. Expanda el menú desplegable *Importar/Exportar* y seleccione *Importar elemento(s)* en la esquina inferior izquierda del diálogo.
2. En el nuevo cuadro de diálogo, indique la fuente de los elementos de estilo (puede ser un archivo `.xml` en el disco o una url).
3. Establecer si *Agregar a favoritos* los elementos a importar.
4. Marque *No importar etiquetas incrustadas* para evitar la importación de etiquetas asociadas a los artículos que se importan.
5. Dé el nombre de cualquier :guilabel:`Etiqueta adicional(es)”para aplicar a los nuevos elementos.
6. Seleccione de la vista previa los símbolos que desea agregar a su biblioteca.
7. Y presione *Importar*.

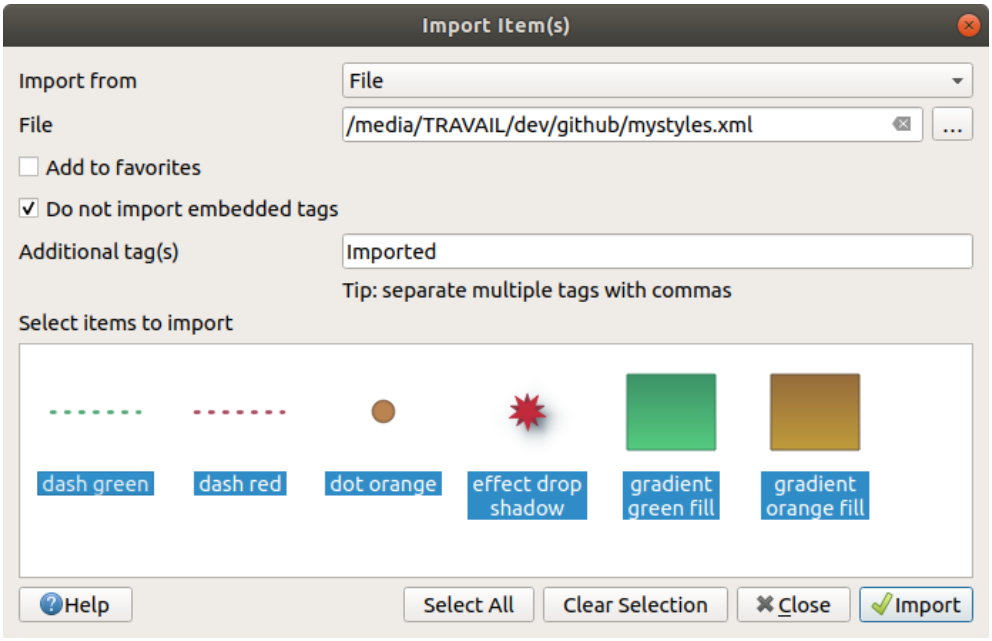


Figura 10.4: Importando elementos de estilo

Usando el panel Navegador

También es posible importar elementos de estilo en la base de datos de estilo de perfil de usuario activo directamente desde el panel *Navegador*:

1. Seleccione el archivo de estilo `.xml` en el navegador
2. Arrástrelo y suéltelo sobre el lienzo del mapa o haga click con el botón derecho y seleccione *Importar estilo …*
3. Rellene el siguiente diálogo *Importar elementos* *Importando elementos*
4. Presione :guilabel:`Importar “y los elementos de estilo seleccionados se agregan a la base de datos de estilos

Al hacer doble click en el archivo de estilo en el navegador, se abre el diálogo :guilabel:Administrador de Estilo` que muestra los elementos del archivo. Puede seleccionarlos y presionar *Copiar al estilo predeterminado …* para importarlos a la base de datos de estilo activa. Las etiquetas pueden asignarse a elementos. También disponible haciendo click derecho, comando *Abrir estilo …*.

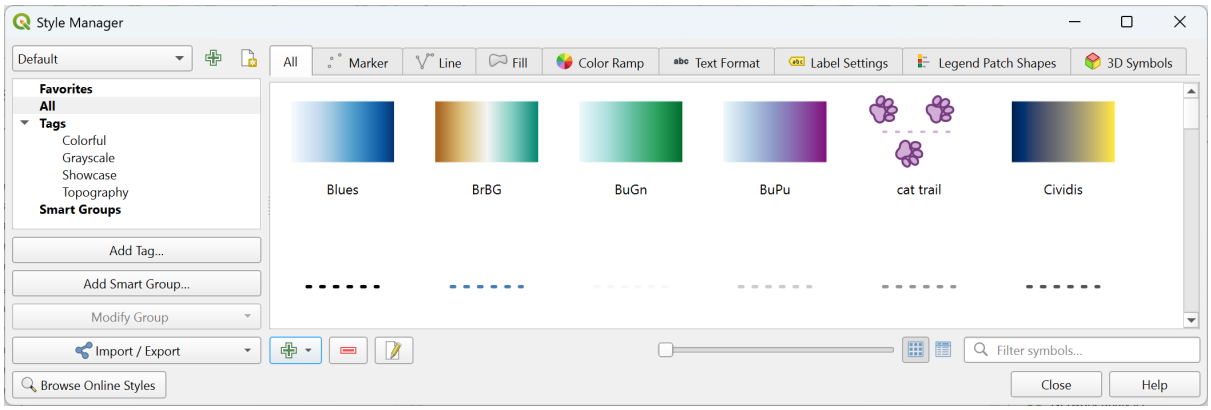


Figura 10.5: Abriendo un archivo de elementos de estilo

El cuadro de diálogo también permite exportar símbolos individuales como `.PNG` o `.SVG`.

Uso del repositorio en línea

El proyecto QGIS mantiene un repositorio con una colección de estilos compartidos por los usuarios de QGIS. Está disponible en https://hub.qgis.org/styles y se puede acceder a él desde el cuadro de diálogo *Gestor de estilos*, pulsando el botón  *Examinar estilos en línea* de la parte inferior.

Desde ese repositorio, puede:

1. Explore y busque cualquier elemento de estilo, según su tipo o nombre
2. Descarga el archivo de estilo y descomprímelo
3. Cargue el archivo basado en `.xml` en su base de datos de estilo en QGIS, utilizando cualquiera de los métodos de importación mencionados anteriormente.

10.1.2 Estableciendo una Rampa de Color

La pestaña Rampa de color en el diálogo *Administrador de estilo* le ayuda a obtener una vista previa de diferentes rampas de color según la categoría seleccionada en el panel izquierdo.

Para crear una rampa de color personalizada, active la pestaña Rampa de color y haga clic en el botón  Agregar elemento. El botón revela una lista desplegable para elegir el tipo de rampa:

- *Gradient*: given a start and end colors, generates a color ramp which can be **continuous** or **discrete**. With double-clicking the ramp preview, you can add as many intermediate color stops as you want. Click on the color stop indicator and under *Gradient stop* you can:
  - ajuste su *Posición relativa* desde el inicio de la rampa de color. También es posible arrastrar el indicador con el ratón, o pulsando las teclas de flecha (combinar con `Mayúsculas` para un movimiento mayor)
  - especifique el modelo de color que se utilizará al interpolar entre colores: puede ser *RGB*, *HSL* o *HSV*. En algunas circunstancias, esta opción puede ayudar a evitar los tonos medios desaturados, lo que resulta en degradados visualmente más agradables.
  - defina la dirección que debe seguir la interpolación para el componente **Hue** de una especificación de color *HSL* o *HSV*. Puede ser *Sentido horario* o *Sentido antihorario*.
  - establece las propiedades *color*
  - eliminar la parada de color pulsando *Borrar parada* o `DEL`.

  El grupo *Gráficos* proporciona otra forma gráfica de diseñar la rampa de color, cambiando la posición o los componentes de opacidad y HSL de los topes de color.

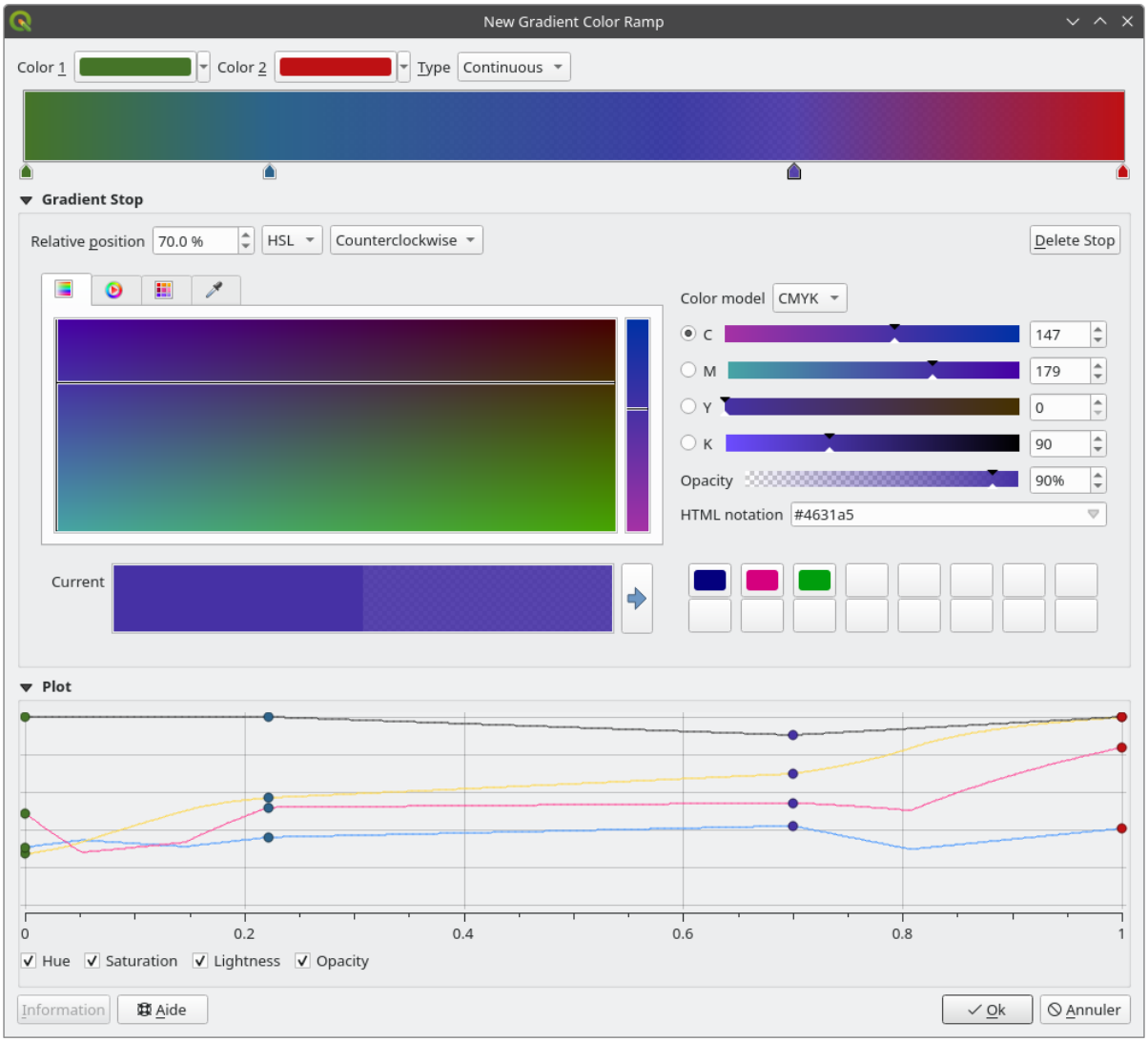


Figura 10.6: Ejemplo de rampa de color degradado personalizado con múltiples paradas

> **Consejo**
>
> Arrastrar y soltar un color desde un punto de color a la vista previa de la rampa de degradado añade una nueva parada de color.

- *preajustes de color*: permite crear una rampa de color que consiste en una lista de colores seleccionados por el usuario;
- *Aleatorio*: crea un conjunto aleatorio de colores basado en el rango de valores para *Matiz*, *Saturación*, *Valor* y *Opacidad* y varios colores (:guilabel:`Clases `);
- *Catálogo: ColorBrewer*: un conjunto de gradientes de color discretos predefinidos que puede personalizar el número de colores en la rampa;
- o *Catálogo: cpt-city*: un acceso a todo un catálogo de gradientes de color a nivel local *guardar como gradiente estándar*. La opción cpt-city abre un nuevo diálogo con cientos de temas incluidos «listos para usar».

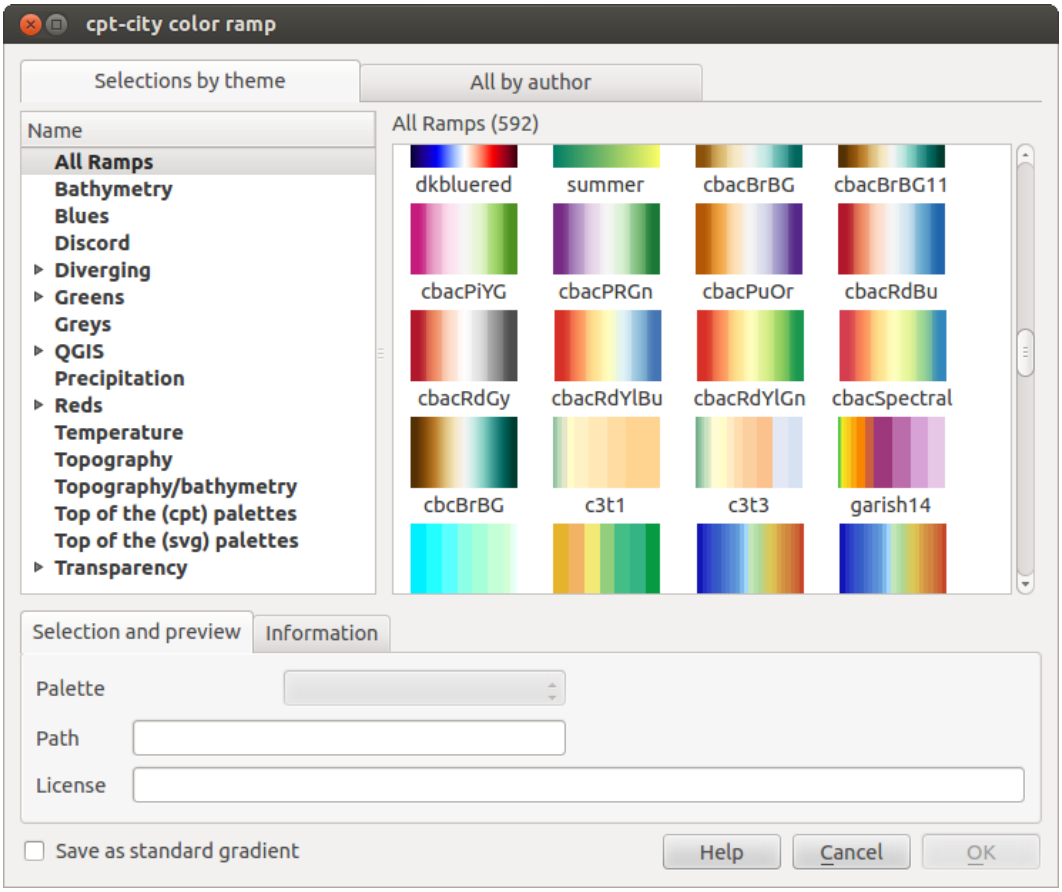


Figura 10.7: diálogo cpt-city con cientos de rampas de color

10.1.3 Creación de una forma de parche de leyenda

Para crear una nueva forma de parche de leyenda, active la pestaña *Formas de parche de leyenda* y haga clic en el botón  Añadir elemento. El botón revela una lista desplegable para elegir el tipo de geometría:

- *Leyenda del marcador Forma del parche…*: para usar con geometrías de puntos.
- *Leyenda de Línea Forma de Parche…*: para usar con geometrías de línea.
- *Llenar la forma de parche de leyenda…*: para usar con geometrías poligonales.

Las tres opciones mostrarán el mismo diálogo.

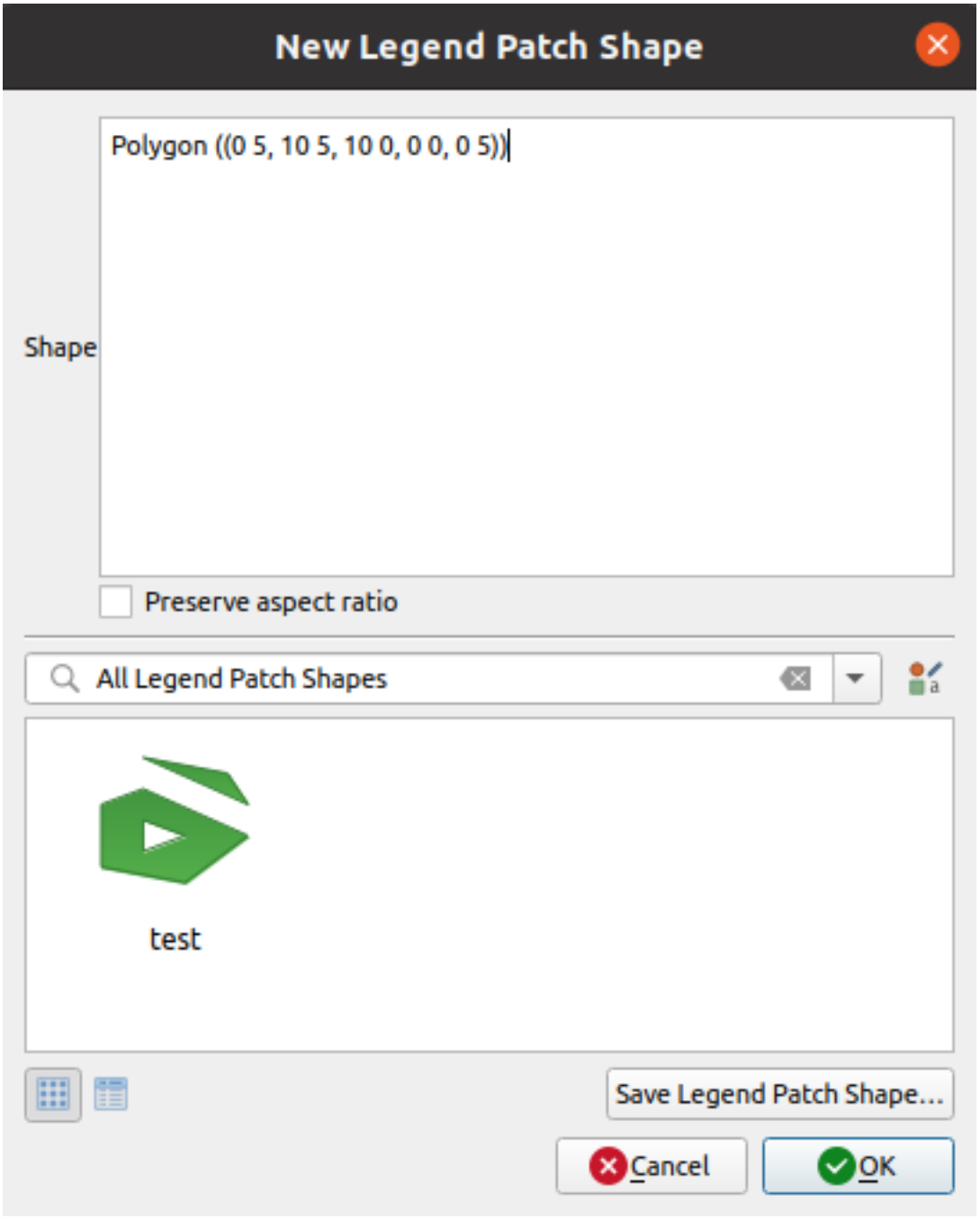


Figura 10.8: Crear una nueva forma de parche de leyenda

Sólo el tipo de forma y las formas de los parches de la leyenda que se muestren diferirán con respecto al tipo de geometría elegido. Las siguientes opciones estarán disponibles:

- *Forma*: define la forma del parche de la leyenda como una cadena WKT. Se pueden utilizar geometrías de una o varias partes, pero no GeometryCollection.
- *Preservar la relación de aspecto*
- *Vista de iconos* o *Vista de lista* de las formas de parches de leyenda disponibles, filtradas por etiquetas.

Cuando la nueva forma está definida, puede *Guardar forma de parche de leyenda…* o pulsar *Aceptar*, que le llevarán al mismo diálogo.

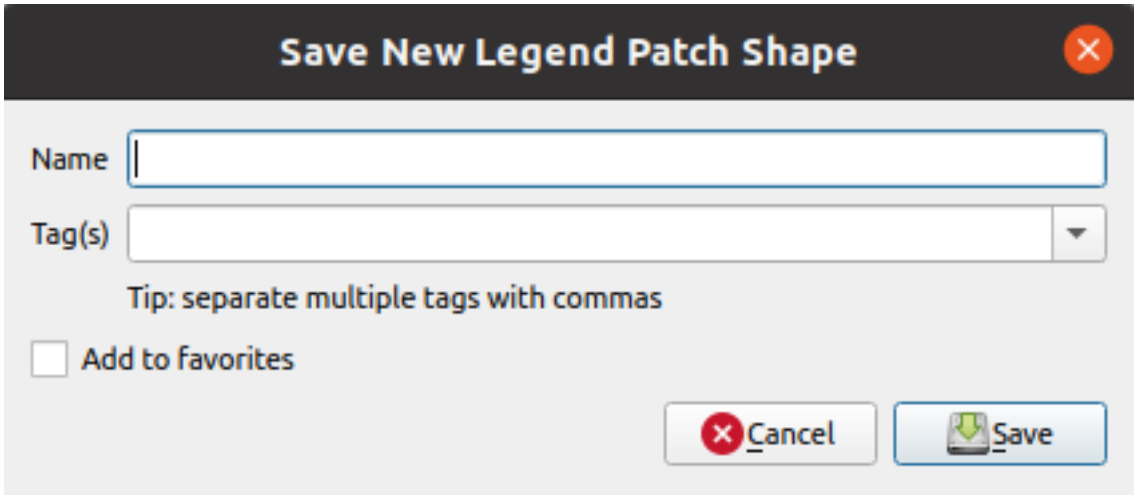


Figura 10.9: Guardar una nueva forma de parche de leyenda

Aquí tiene que elegir un nombre, etiquetas para describir la forma y si debe ser añadido a los favoritos.

Si pulsa *Guardar…*, la forma se añade a la lista y se le dirige de nuevo al diálogo *Nueva forma de parche de leyenda* para seguir creando nuevas formas.

10.2 EL selector de símbolo

EL selector de símbolo en el diálogo principal paa diseñar un símbolo. Puede crear o editar Marcadores, Líneas o Símbolos Rellenos

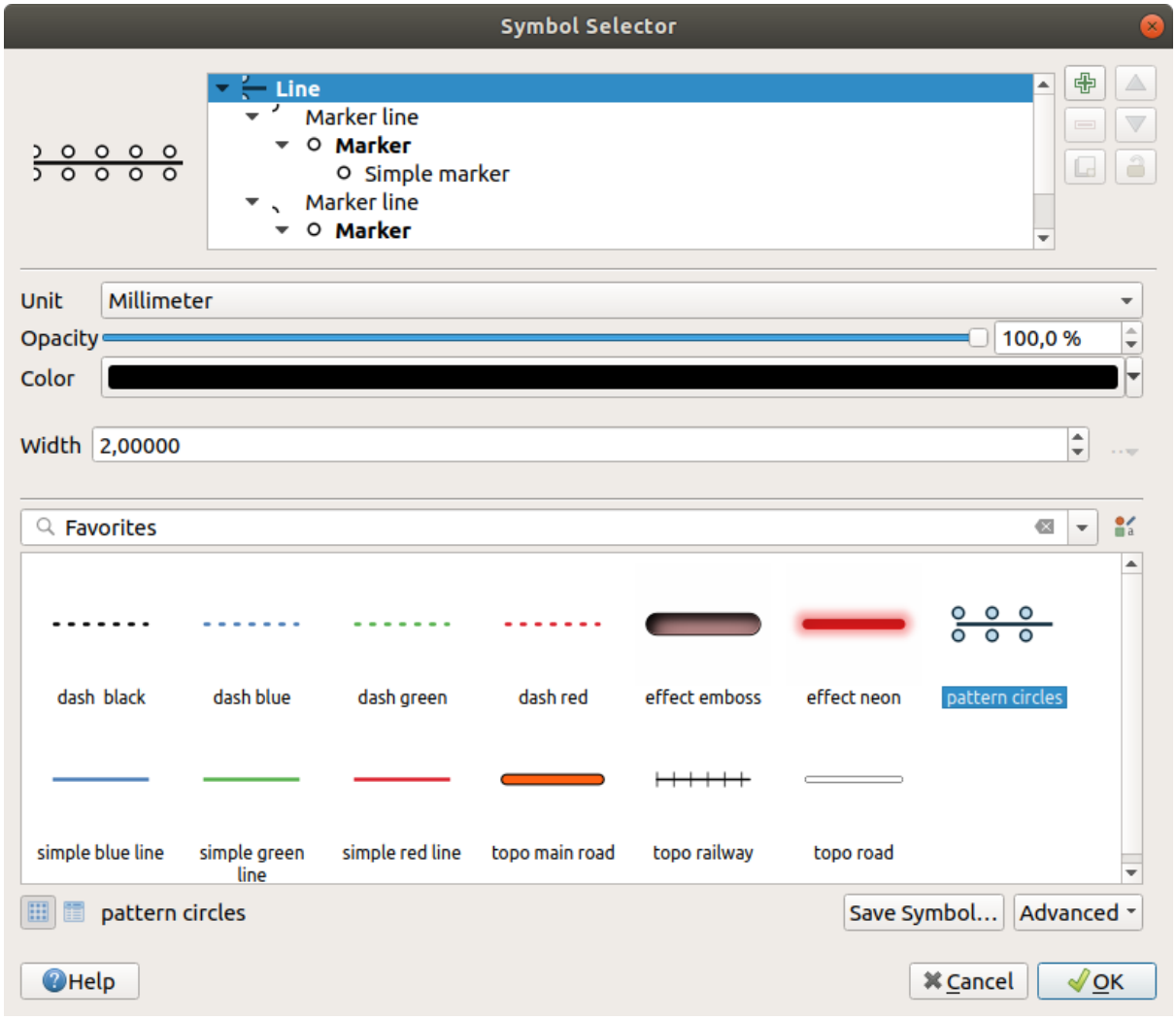


Figura 10.10: Diseñando un símbolo lineal

Dos componentes principales estructuran el diálogo de selector de símbolo:

- el arbol de símbolos, mostrando capas de símbolos que son combinados luego para dar forma a un nuevo símbolo global
- y configuraciones para configurar la capa de símbolo seleccionada en el árbol.

10.2.1 El árbol de símbolo de capa

Un símbolo puede constar de varias *Capas de símbolos*. El árbol de símbolos muestra la superposición de estas capas de símbolos que se combinan luego para formar un nuevo símbolo global. Además, una representación dinámica de símbolo se actualiza tan pronto como cambian las propiedades del símbolo.

Dependiendo del nivel seleccionado en los elementos del árbol de símbolos, hay varias herramientas disponibles para ayudarlo a administrar el árbol:

- añade nuevo símbolo de capa: puede apilar tantos símbolos como quiera
- borrar el símbolo de capa seleccionado
- colores de bloqueo de la capa de símbolo: a el color bloqueado permanece sin cambios cuando el usuario cambia el color en el nivel de símbolo global (o superior)

-  duplicar un (grupo de) capas(s) de símbolo
- mover arriba y abajo una capa de símbolo

10.2.2 Configurando un símbolo

En QGIS, la configuración de un símbolo se hace en dos pasos: el símbolo y despues la capa de símbolo.

El símbolo

En el nivel superior del árbol, depende de la geometría de la capa y puede ser del tipo **Marcador**, **Línea** o **Relleno**. Cada símbolo puede incrustar uno o más símbolos (incluidos, de cualquier otro tipo) o capas de símbolos.

Puede configurar algunos parámetros a aplicar al símbolo global:

- *Unidad*: puede ser **Milímetros**, **Puntos**, **Píxeles**, **Metros a escala**, **Unidades de mapa** o **Pulgadas** (ver *Selector de Unidad* for more details)
- *Opacidad*
- *Color*: cuando el usuario cambia este parámetro, su valor se repite en todos los colores de los símbolos secundarios desbloqueados
- *Tamaño* y *Rotación* para símbolos de marcador
- *Anchura* para símbolos de línea

> **Truco**
>
> Use las propiedades *Tamaño* (para símbolos de marcador) o la *Anchura* (para símbolos de línea) propiedades a nivel de símbolo para redimensionar proporcionálmente todas las dimensiones incrustadas *symbol layers*.

> **Nota**
>
> El botón *Data-defined override* de al lado de los parámetros anchura, tamaño o rotación está inactivo al configurar el símbolo desde el cuadro de diálogo Administrador de estilos. Cuando el símbolo está conectado a una capa de mapa, este botón te ayuda a crear *proportional or multivariate analysis* de representación.

- A preview of the *symbols library*: Se muestran símbolos del mismo tipo y, a través de la lista desplegable editable justo arriba, se pueden filtrar por texto de forma libre o por *categories*. También puede actualizar la lista de símbolos utilizando el botón  Administrador de Estilo y abra el cuadro de diálogo del mismo nombre. Allí, puede usar cualquier capacidad como se expone en la sección *El administrador de Estilo*.

  Los símbolos se muestran:

  - en una lista de iconos (con miniatura, nombre y etiquetas asociadas) usando el botón  Vista de lista debajo del marco;
  - o como vista previa de icono usando el botón  Vista de icono.

- Pulse el botón *Guardar símbolo* para abrir el cuadro de diálogo *Guardar nuevo símbolo*. Aquí puede elegir el *Destino* donde desea añadir el símbolo que está editando, darle un *Nombre* y añadir *Etiqueta(s)*. También tiene la opción de marcar Añadir a favoritos su nuevo símbolo.
- Con la opción *Avanzado*  , puedes:

- para símbolos de línea y relleno, *Recortar entidades a la extensión del lienzo*.
- para símbolos de relleno, :guilabel:`Forzar orientación por la regla de la mano derecha ": permite forzar a los símbolos de relleno representados a seguir la «regla de la mano derecha «estándar para la orientación del anillo (es decir, polígonos donde el anillo exterior está en el sentido de las agujas del reloj, y los anillos interiores son todos en sentido anti-horario).

  La corrección de orientación se aplica solo mientras se procesa, y la geometría de la entidad original no cambia. Esto permite la creación de símbolos de relleno con una apariencia consistente, independientemente del conjunto de datos que se representa y la orientación del anillo de las entidades individuales.
- para los símbolos de marcador, la opción *Configuración del búfer…* permite añadir un efecto de halo alrededor del símbolo de marcador con el fin de que sea más legible sobre diferentes fondos. El búfer se calcula y dibuja utilizando la forma de TODAS las capas de símbolos del marcador. Puede ajustar el *Tamaño*, el *Estilo de unión* y el *Símbolo* de visualización.
- for all symbol types, enable and configure a distance for the map canvas *Extent buffer*, or the *Animation settings*.
- Depending on the *symbology* of the layer the symbol is being applied to, additional settings are available in the *Advanced* menu:
  - *Symbol levels…* para definir el orden de la representación de los símbolos
  - *Data-defined Size Legend*
  - *Coincidir con símbolos guardados …* y *Coincidir con símbolos del archivo…* a automáticamente *assign symbols to classes*

La capa de símbolo

En un nivel inferior del árbol, puede personalizar las capas de símbolos. Los tipos de capa de símbolo disponibles dependen del tipo de símbolo superior. Puedes aplicar en la capa de símbolo  *paint effects* para mejorar su representación.

Debido a que no sería posible describir todas las opciones de todos los tipos de capas de símbolos, a continuación solo se mencionan las particulares y significativas.

Parámentros comunes

Algunas opciones y widgets comunes están disponibles para construir una capa de símbolo, independientemente de que sea un subtipo de marcador, línea o relleno:

- el widget *color selector* para facilitar la manipulación del color
- *Unidades*: pueden ser **Milímetros**, **Puntos**, **Píxeles**, **Metros a escala**, **Unidades de mapa** o **Pulgadas** (ver *Selector de Unidad* para mas detalles)
- el widget de  :sup:"anulación definida por datos" cerca de casi todas las opciones, amplía las capacidades de personalizar cada símbolo (ver *Configuración de anulación definida por datos* para más información)
- la opción  *Activar capa de símbolos* controla la visibilidad de la capa de símbolos. La capa de símbolos deshabilitadas no se dibujan al renderizar el símbolo pero guardada en el símbolo. Poder ocultar capas de símbolos es conveniente cuando se busca el mejor diseño de su símbolo, ya que no es necesario eliminar ninguna para probar. La anulación definida por datos permite ocultar o mostrar diferentes capas de símbolos según expresiones (utilizando, por ejemplo, atributos de objetos).
- el botón  *Dibujar Efectos* para *effects rendering*.

Nota

Si bien la descripción a continuación asume que el tipo de capa de símbolo está vinculado a la geometría de la entidad, tenga en cuenta que puede incrustar capas de símbolo en las demás. En ese caso, el parámetro de la capa de símbolo de nivel inferior (ubicación, desplazamiento …) podría estar vinculado al símbolo de nivel superior y no a la geometría de la entidad en sí.

Símbolos de Marcador

Apropiado para las características de geometría de puntos, los símbolos de marcador tienen varios *Tipos de capa de símbolo*:

Tabla 10.1: List of marker symbol layer types

Simple marker	*Animated marker*	*Ellipse marker*
Filled marker	*Font marker*	*Geometry generator*
Mask	*Raster image marker*	*Vector Field marker*
SVG marker		

- **Marcador Simple** (predeterminado)

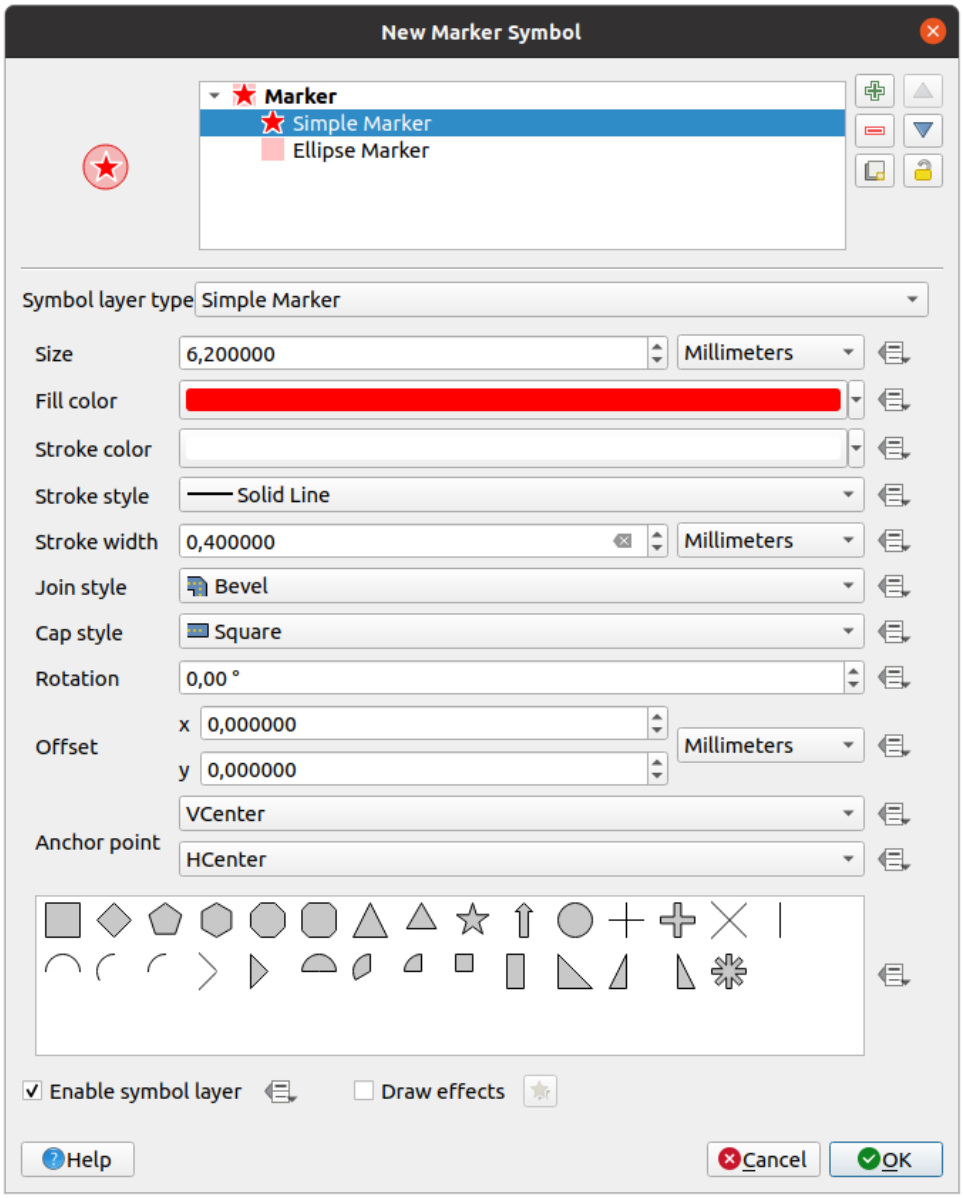


Figura 10.11: Diseñando un Símbolo de Marcador Simple

El tipo de capa de símbolo de marcador simple tiene las siguientes propiedades:

- *Tamaño* en varias unidades soportadas
- *Color de relleno*
- *Color del trazo*, *Estilo del trazo* de una lista predefinida y *Tamaño del trazo*
- *Estilo de unión*: puede ser **Bisel**, **Inglete** o **Redondo**
- *Estilo de capa*: puede ser **Cuadrado**, **Plano** o **Redondo**
- *Rotación*
- *Compensación* en *X* y *Y* direcciones desde la característica
- *Punto de anclaje*: definiendo el punto cuadrante sobre el símbolo a asentar como origen de colocación. Este es el punto sobre el que se aplica la *Compensación*.

- **Marcador animado** (véase *Marcador animado*)
- **Marcador Elipse**: una capa de símbolo de marcador simple, con ancho y alto personalizables

- **Marcador relleno**: similar a la capa de símbolo de marcador simple, excepto que usa un *fill sub symbol* para representar el marcador. Esto permite el uso de todos los estilos de relleno (y trazo) QGIS existentes para representar marcadores, p.ej. rellenos de degradado o explosión de forma.
- **Marcador de Fuente**: similar al marcador simple de capa de símbolos, excepto que usa las fuentes instaladas para representar el marcador. Sus propiedades adicionales son:
  - *Familia de Fuente*
  - *Estilo de Fuente*
  - *Caracter(es)*, representa el texto a mostrar como símbolo. Puede escribirse o seleccionarlo del control de colección de carácteres de fuente y puede hacer una *Previsualización* instantánea de ellos con la configuración seleccionada.
- **Generator de Geometría** (ver *El generador de Geometría*)
- **Máscara**: su sub-símbolo define una forma de máscara cuya propiedad de color será ignorada y solo se usará la opacidad. Esto es conveniente cuando el símbolo del marcador se superpone con etiquetas u otros símbolos cuyos colores están cerca, lo que dificulta su descifrado. Más detalles en *Propiedades de Máscaras*.
- **Marcador de imagen de trama**: usar una imagen (`PNG`, `JPG`, `BMP` ...) como símbolo marcador. La imagen puede ser un archivo en el disco, una URL remota incrustada en la base de datos de estilo (*más detalles*) o puede estar codificada como una cadena base64. El ancho y el alto de la imagen puede sr establecido independientemente o usando la  Relación de aspecto de bloqueo. El tamaño se puede establecer usando cualquiera de las *unidades comunes* o como un porcentaje del tamaño original de la imagen (en escala con el ancho).
- **Marcador de campo vectorial** (ver *El Marcador de Campo Vectorial*)
- **Marcador SVG**: le proporciona imágenes desde sus rutas SVG (establecidas en el menú *Ajustes* ► *Opciones...* ► *Sistema*) para representar como símbolo de marcador. El ancho y el alto del símbolo puede ser establecido usando la  Relación de aspecto de bloqueo. También se pueden adaptar el color y el trazo de cada archivo SVG. La imagen puede ser un archivo en el disco, una URL remota incrustada en la base de datos de estilo (*más detalles*) o puede estar codificada como una cadena base64..

  El símbolo también se puede configurar con *Parámetros SVG dinámicos*. Consulte la sección *SVG parametrizable* para parametrizar un símbolo SVG.

> **Nota**
>
> Requerimientos de version SVG
>
> QGIS renderiza archivos SVG que siguen el perfil SVG Tiny 1.2, destinado a la implementación en una variedad de dispositivos, desde teléfonos celulares y PDA hasta computadoras portátiles y de escritorio, y por lo tanto incluye un subconjunto de las características incluidas en SVG 1.1 Full, junto con nuevas prestaciones para ampliar las capacidades de SVG.
>
> Es posible que algunas entidades no incluidas en estas especificaciones no se representen correctamente en QGIS.

Símbolos de Línea

Apropiada para entidades de geometría lineal, los símbolos lineales tienen los siguientes tipos de capa de símbolos:

Tabla 10.2: List of line symbol layer types

Simple line	*Arrow*	*Filled line*
Geometry generator	*Hashed line*	*Interpolated line*
Linear referencing	*Lineburst*	*Marker line*
Raster line		

- **Línea simple** (predeterminado)

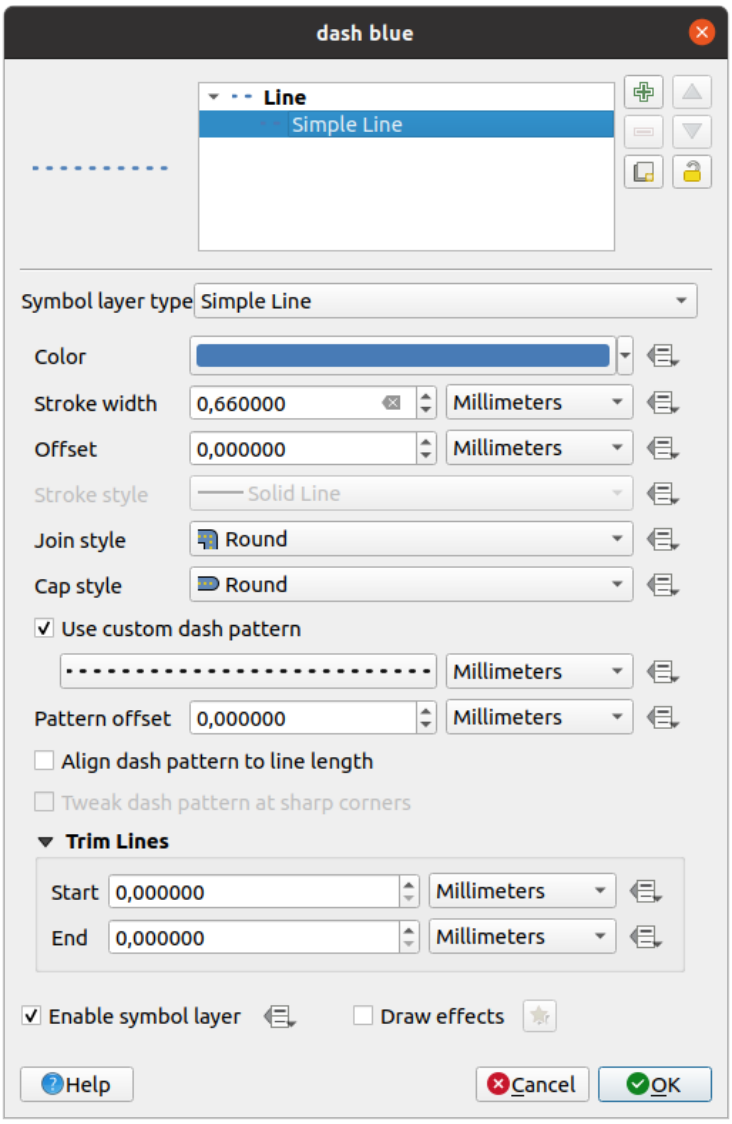


Figura 10.12: Diseñar un símbolo de línea simple

El tipo de capa de símbolo de línea simple tiene muchas de las mismas propiedades que el *simple marker symbol*, y además:

- *Usar patrón de guión personalizado*: anula la configuración de *Estilo de trazo* con un guión personalizado. Tendrá que definir la longitud de los guiones y espacios consecutivos que conforman el modelo, en la unidad elegida. La longitud total del patrón se muestra en la parte inferior del diálogo.
- *Desplazamiento de patrón*: la posición de los guiones/espacios en la línea se puede modificar, de modo que se puedan colocar en posiciones más agradables para tener en cuenta las esquinas de la línea (también se puede usar potencialmente para «alinear» el guión adyacente bordes del patrón)
- *Alinear patrón de guiones a longitud de la línea*: el patrón de guiones será ajustado hasta que finalice la línea con un elemento de guión entero, en lugar de un hueco.
- *Ajustar el patrón de guiones en las esquinas afiladas*: ajusta dinámicamente la ubicación del patrón de guiones para que las esquinas afiladas estén representadas por un elemento de guión completo que entra y sale de la esquina afilada. Depende de *Alinear el patrón de trazos con la longitud de la línea*.

  - *Recortar líneas* de *Inicio* y/o *Fin*: permite que el renderizado de líneas recorte los primeros x mm y los últimos y mm de la cadena de líneas real al dibujar la línea. Admite un rango de *unidades*, incluyendo el porcentaje de la longitud total de la línea, y puede definirse con datos para un control extra. La distancia de recorte inicial/final puede utilizarse, por ejemplo, al crear símbolos complejos en los que una capa de línea no debe superponerse a las capas de símbolos marcadores colocadas al principio y al final de la línea.

- **Flecha**: dibuja líneas como curvas (o no) con cabeza simple o doble con configuración (y definida por datos):
  - *Tipo de cabeza*
  - *Tipo de Flecha*
  - *Anchura de Flecha*
  - *Anchura de Flecha en Origen*
  - *Longitud de Cabeza*
  - *Espesor de la Cabeza*
  - *Desplazamiento*

  Es posible crear *flechas Curvas* (el objeto lineal debe tener al menos tres vértices), y *Repetir flecah en cada segmento*. También usa un *símbolo relleno* así como gradientes o estallidos de forma para representar el cuerpo de la flecha. Combinado con el generador de geometría, este tipo de símbolo de capa le ayuda a representar mapas de flujo.

- **Generator de Geometría** (ver *El generador de Geometría*)
- **Línea interpolada**: permite renderizar una línea cuyo *Ancho del trazo* y/o *Color* puede ser constante (dado un *Ancho fijo* y *Color único* parámetros) o varían a lo largo de la geometría. Cuando varían, las entradas necesarias son:
  - *Valor inicial* y *Valor final*: Valores que se utilizarán para la interpolación en los extremos de la geometría de las características. Pueden ser valores fijos, atributos de características o basados en una expresión.
  - *Mín. valor* y *Máx. valor*: Valores entre los que se realiza la interpolación. Pulse el botón Cargar para llenarlos automáticamente en función de los valores de inicio/final mínimos y máximos aplicados a la capa.
  - Solo disponible para la opción de trazo:
    - *Mín. ancho* y *Máx. ancho*: define el rango del ancho variable. *Mín. ancho* se asigna a *Mín. valor* y *Máx. ancho* al *Máx. valor*. Se puede asociar un *unit*
    - *Usar valor absoluto*: considerar sólo el valor absoluto para la interpolación (los valores negativos se usan como positivos).
    - *Ignore out of range*: by default, when the `[start value – end value]` range of a feature is not included in the `[min. value – max. value]` range, the out-of-bounds parts of the feature's geometry are rendered with the min or max width. Check this option to not render them at all.
  - Para variar el color, puede utilizar cualquiera de los métodos de interpolación de *color ramp classification*

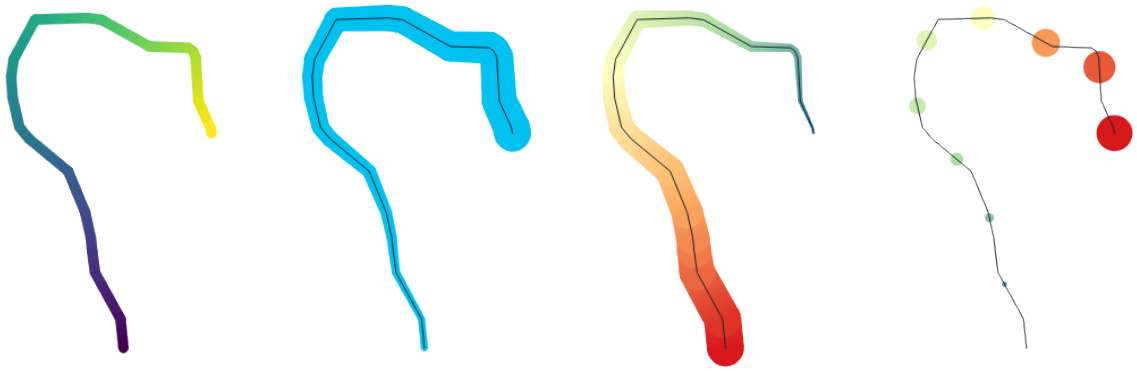


Figura 10.13: Ejemplos de líneas interpoladas

- **Marcador lineal**: repite un *marker symbol* a lo largo de una línea.
  - La *Colocación de la marca* puede establecerse utilizando una configuración de intervalo regular (empezando por el primer vértice) y/o la propiedad de geometría de la línea (en el primer o último vértice, en los vértices interiores, en el punto central de la línea o de cada segmento, o en cada punto de la curva).
  - Cuando se activa la colocación en el primer o último vértice, la opción *Colocar en cada extremidad de la parte* hará que los marcadores se representen también en el primer o último vértice para cada parte de las geometrías multiparte.
  - *Desplazamiento a lo largo de la línea*: la colocación de los marcadores también puede tener un desplazamiento a lo largo de la línea, en la ref:*unidad <unit_selector>* de su elección (milímetros, puntos, unidad cartográfica, metros a escala, porcentaje, …):
    - Un valor positivo desplaza los símbolos marcadores en la dirección de la línea (con las colocaciones *En primer vértice* y *Con intervalo*) y hacia atrás (con la colocación *En último vértice*).
    - Un valor negativo en una línea no cerrada provocará que no haya desplazamiento (para las colocaciones *En el primer vértice* y *En el último vértice*) o que los símbolos se desplacen hacia atrás (desde el último vértice).
    - Con un anillo cerrado, QGIS trata el desplazamiento como si continuara haciendo un bucle alrededor del anillo (hacia delante o hacia atrás). Por ejemplo, si se establece un desfase del 150% (resp. -10% o -110%), el desfase se trata como el 50% (resp. 90%) de la longitud del anillo cerrado.
  - La opción *Rotar marcador para seguir la dirección de línea* establece si cada símbolo de marcador debe orientarse en relación con la dirección de la línea o no.

    Debido a que una línea es a menudo una sucesión de segmentos de diferentes direcciones, la rotación del marcador se calcula promediando una distancia específica a lo largo de la línea. Por ejemplo, establecer la propiedad *Ángulo promedio* en``4 mm “” significa que los dos puntos a lo largo de la línea que son `2 mm` antes y después de la colocación del símbolo se utilizan para calcular el ángulo de línea para ese símbolo marcador Esto tiene el efecto de suavizar (o eliminar) cualquier pequeña desviación local de la dirección general de la línea, lo que da como resultado orientaciones visuales mucho más agradables de los símbolos de línea de marcador.
  - *Desplazamiento de línea*: los símbolos de marcador también se pueden desplazar de la entidad de línea.
- **Línea de trazos**: repite un segmento de línea (un trazo) a lo largo de un símbolo de línea, con un sub-símbolo de línea utilizado para representar cada segmento individual. En otras palabras, una línea discontinua es como una línea de marcador en la que los símbolos de marcador se reemplazan con segmentos. Como tal, las líneas de trazos tienen el *same properties* que los marcadores de símbolos, junto con:
  - *longitud de Trazo*
  - *rotación de Trazo*

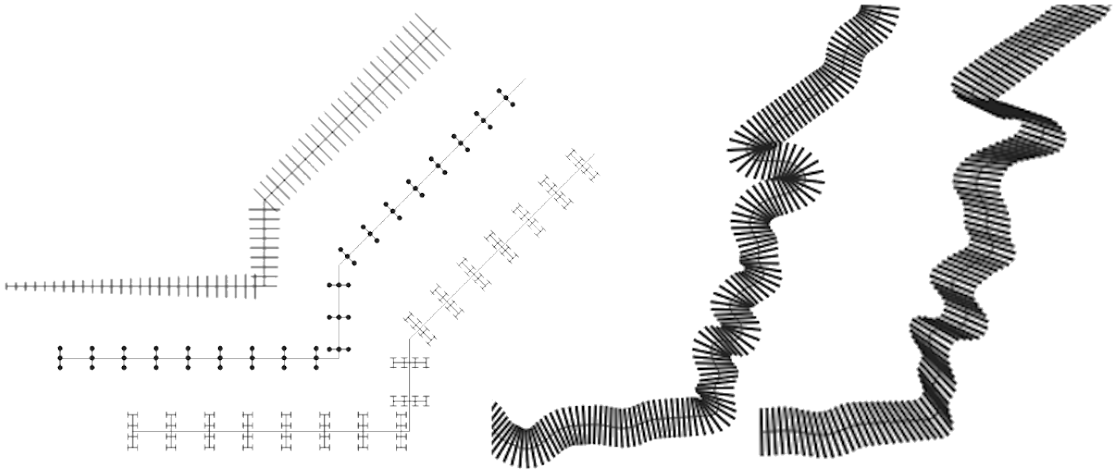


Figura 10.14: Ejemplos de líneas de trazos

- **Línea ráster**: representa y repite una imagen de trama siguiendo la longitud de una línea. Se puede ajustar *Anchura de trazo*, *Desplazamiento*, *Estilo de unión*, *Estilo de tapa* y *Opacidad*.

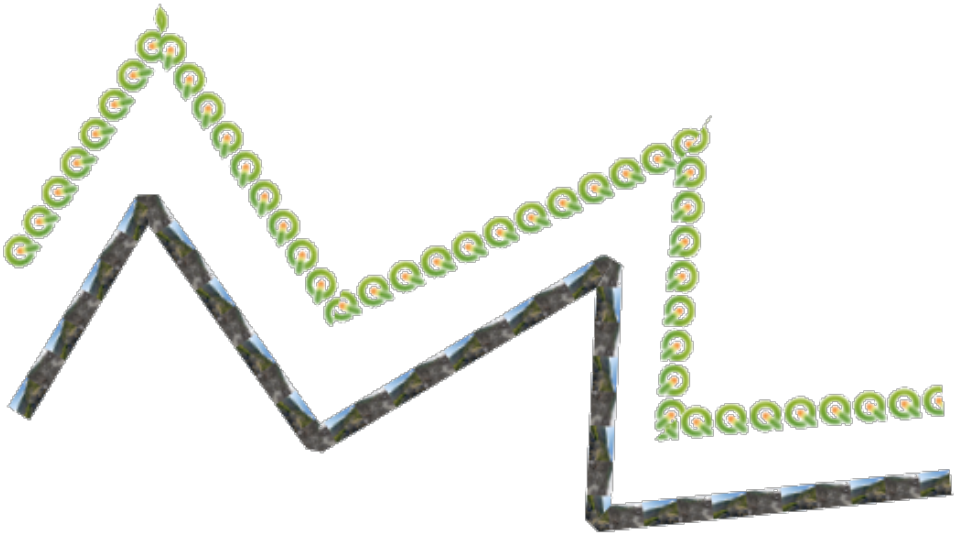


Figura 10.15: Ejemplos de líneas raster

- **Referenciación lineal**: permite colocar etiquetas de texto a intervalos regulares a lo largo de una línea, o en posiciones correspondientes a vértices existentes. Las posiciones se pueden calcular utilizando distancias cartesianas o interpoladas a partir de valores z/m. Los parámetros incluyen:
  - *Colocación de medidas*: Las etiquetas se pueden colocar a distancias cartesianas 2D regulares, a intervalos regulares interpolados linealmente calculados utilizando los valores Z o M en geometrías, o en vértices existentes. Se proporciona un *Intervalo* fijo o definido por datos.
  - *Cantidad*: La etiqueta que se muestra en cada posición representa la distancia recorrida a lo largo de la línea, o el valor Z o M interpolado linealmente. Las etiquetas se representan utilizando todo el rango de funcionalidades de formateo de *texto* y *número* disponibles en QGIS.

> **Nota**
>
> Cuando se utiliza la colocación basada en la distancia o las etiquetas, las distancias se calculan utilizando solo 2D. Los cálculos cartesianos se basan en el SRC de la capa original.

- *Omitir múltiplos de*: Si se establece, se omitirán las etiquetas que sean múltiplos de este valor. Esto permite la creación de etiquetas de referencia complejas, por ejemplo, cuando un símbolo tiene dos capas de símbolos de referencia lineal, una configurada para etiquetar cada 100 m con una fuente pequeña, omitiendo los múltiplos de 1000, y otra configurada para etiquetar cada 1000 m con una fuente grande y en negrita.
- *Ángulo promedio sobre*: Las etiquetas se representan utilizando un ángulo calculado a partir del promedio de la cadena de líneas, de modo que los pequeños bordes dentados no provocan una rotación antiestética de las etiquetas.
- *Mostrar símbolos de marcador*, en los puntos referenciados de la característica de línea, utilizando un símbolo de marcador completo de QGIS. Esto permite, por ejemplo, mostrar un sombreado cruzado en el punto etiquetado, para una línea de estilo «regla».

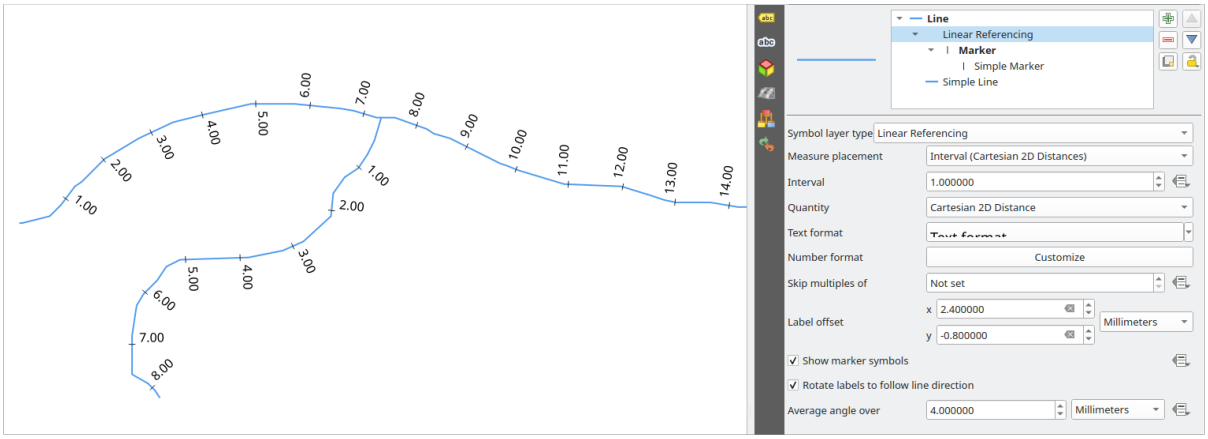


Figura 10.16: Dibujar etiquetas lineales a lo largo de una característica lineal

- **Línea de ráfaga**: representa un degradado a lo ancho de una línea. Puede elegir entre *Dos colores* o *Rampa de colores* y puede ajustar el *Ancho de trazo*, *Desplazamiento*, *Estilo de unión*, *Estilo de tapa*.



Figura 10.17: Ejemplos de líneas de ráfaga

- **Línea rellena**: renderiza el interior de las líneas utilizando un símbolo de relleno que permite líneas rellenas con degradados, rayados, etc. Se pueden ajustar los parámetros *Ancho del trazo*, *Desplazamiento*, *Estilo de unión* y *Estilo de extremo*.

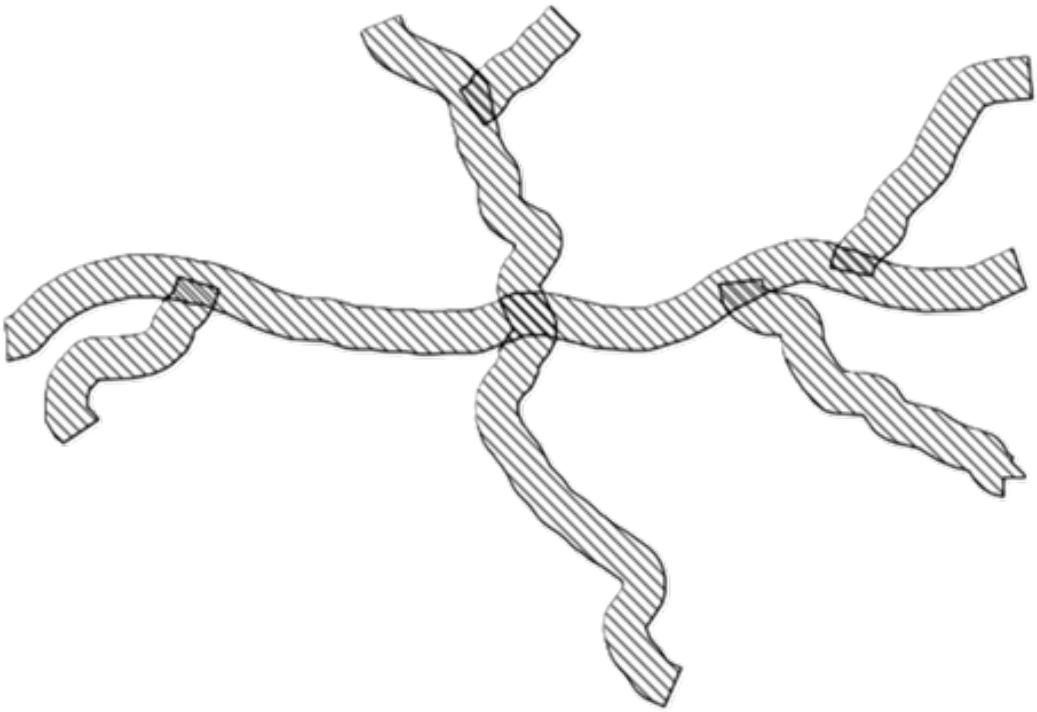


Figura 10.18: Ejemplos de líneas rellenas

Símbolos rellenos

Apropiado para las entidades de geometría poligonal, los símbolos de relleno también tienen varios tipos de capas de símbolos:

Tabla 10.3: List of fill symbol layer types

For polygon interior	For polygon ring
Simple fill	*Outline: Arrow*
Centroid fill	*Outline: Filled line*
Geometry generator	*Outline: Hashed line*
Gradient fill	*Outline: Interpolated line*
Line pattern fill	*Outline: Linear referencing*
Point pattern fill line	*Outline: Lineburst*
Random marker fill	*Outline: Marker line*
Raster image	*Outline: Raster line*
SVG fill	*Outline: Simple line*
Shapeburst fill	

- **Relleno simple** (predeterminado): llena un polígono con un color uniforme

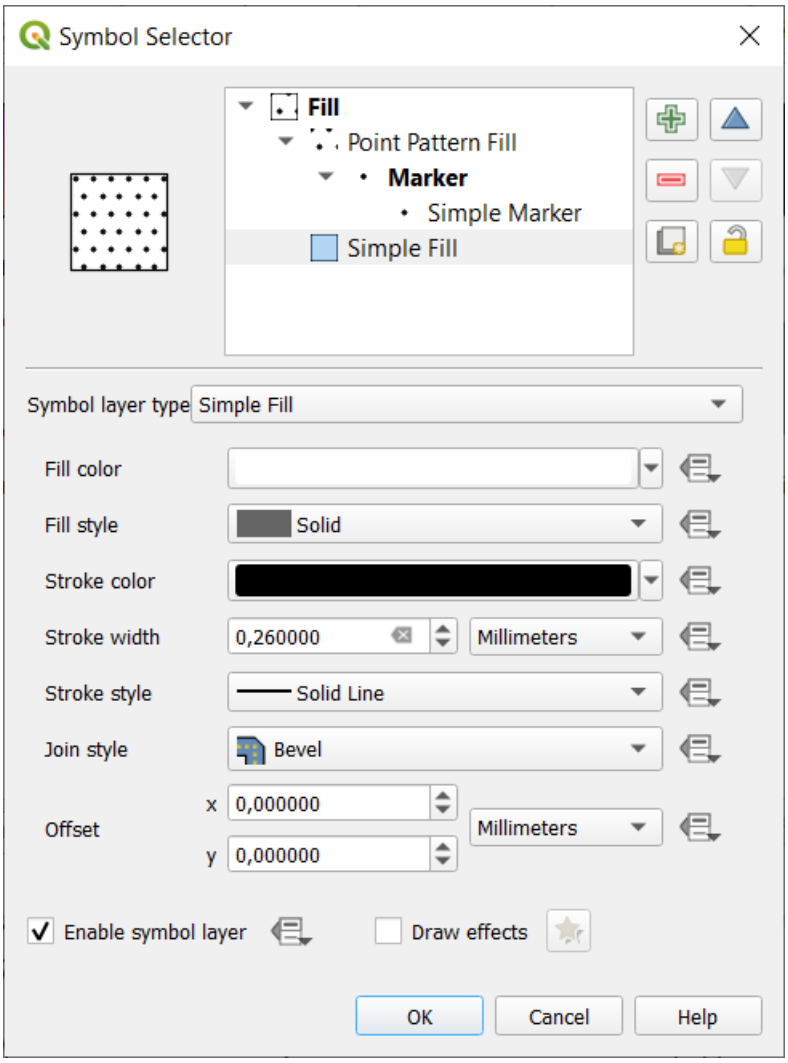


Figura 10.19: Diseñar un símbolo de relleno simple

- **Centroide relleno**: ubica un *marcador de símbolo* en el centroide la la entidad visible. La posición del marcador puede no ser el centroide real del objeto, porque el cálculo tiene en cuenta el polígono(s) recortado en el área visible del lienzo del mapa a renderizar e ignora los huecos.Use the *generador de geometría de símbolo* si quiere el centroide exacto.

  Puede:

  - *Forzar colocación de marcadores dentro de polígonos*
  - *Dibuje marcadores en cada parte de entidades de varias partes* o coloque el punto solo en su parte más grande
  - muestra el marcador de símbolo(s) en todo o en una parte, manteniendo partes superpuestas a la geometría de objeto actual (*Cortar marcadores en límite del polígono*) o la parte de la geometría a la que pertenece el símbolo (*Cortar marcadores solo en límites de la parte actual*)

- **Generator de Geometría** (ver *El generador de Geometría*)
- **Relleno de degradado**: utiliza un degradado radial, lineal o cónico, basado en degradados simples de dos colores o en una predefinida *rampa de colores de degradado* para rellenar polígonos. El gradiente puede girarse y aplicarse a un elemento concreto o a toda la extensión del mapa. Los puntos inicial y final pueden establecerse mediante coordenadas o utilizando el centroide (del objeto espacial o del mapa). Puede definirse un desplazamiento definido por los datos.
- **Relleno de patrón de línea**: rellena el polígono con un patrón de sombreado de *capa de símbolo de línea*. Puede definir:
  - *Alineación*: define cómo se posiciona el patrón en relación con lo(s) objeto(s) espacial(es):
    - *Alinear el patrón al objeto espacial*: las líneas se representan dentro de cada objeto espacial
    - *Alinear el patrón a la extensión del mapa*: un patrón se representa sobre toda la extensión del mapa, permitiendo que las líneas se alineen bien a través de los objetos espaciales
  - *Rotación* de las líneas, en sentido contrario a las agujas del reloj
  - *Espaciado*: distancia entre líneas consecutivas
  - *Compensar* distancia de las líneas desde el límite del objeto espacial
  - *Recorte*: permite controlar cómo deben recortarse las líneas del relleno a la forma del polígono. Las opciones son:
    - *Recortar sólo durante el renderizado*: las líneas se crean cubriendo todo el cuadro delimitador de la característica y luego se recortan mientras se dibujan. Los extremos de la línea (principio y final) no serán visibles.
    - *Recortar líneas antes de renderizar*: las líneas se recortan a la forma exacta del polígono antes de renderizar. Los extremos de la línea (incluyendo estilos de tapa, objetos de línea de marcador de inicio/fin, …) serán visibles, y a veces pueden extenderse fuera del polígono (dependiendo de la configuración del símbolo de línea).
    - *Sin recorte*: no se realiza ningún recorte - las líneas cubrirán todo el cuadro delimitador del elemento

  Nota

  When exporting maps to vector formats such as PDF or SVG, or when the line symbol in the pattern has data-defined properties, QGIS renders the line pattern fill using a line-by-line vector-based approach. This results in smaller output file sizes, higher quality exports, and improved editability of the pattern lines in external vector graphics applications.

- **Relleno de patrón de puntos**: rellena el polígono con un patrón de cuadrícula de *símbolo de marcador*. Puede definir:
  - *Alineación*: define cómo se posiciona el patrón en relación con lo(s) objeto(s) espacial(es):
    - *Alinear el patrón al objeto espacial*: las líneas de marca se representan dentro de cada objeto espacial

    * *Alinear el patrón a la extensión del mapa*: un patrón se representa sobre toda la extensión del mapa, permitiendo que los marcadores se alineen bien a través de los objetos espaciales

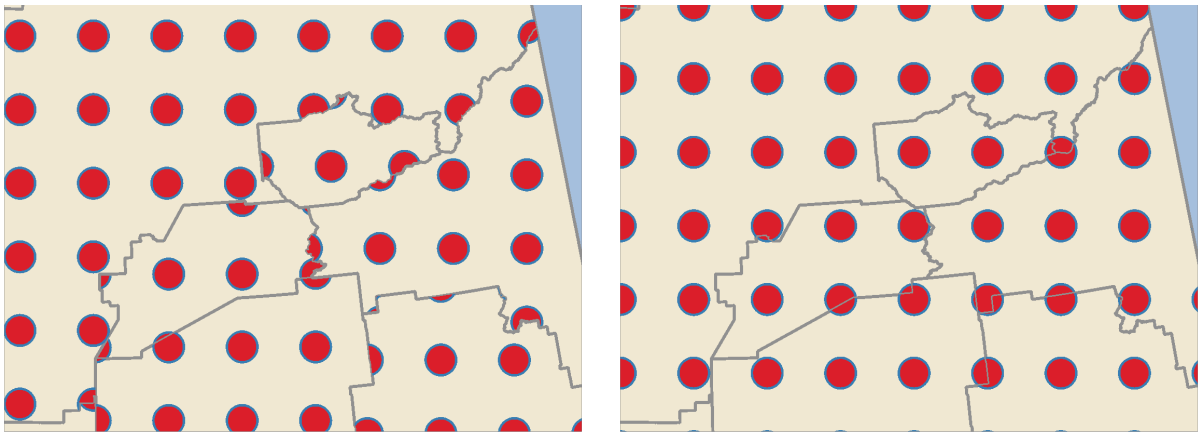


Figura 10.20: Alineación del patrón de puntos con el elemento (izquierda) y con la extensión del mapa (derecha)

- *Distancia*: *Horizontal* y *Vertical* distancias entre marcadores consecutivos
- *Desplazamiento*: un *Horizontal* (resp. *Vertical*) desplazamiento de alineación entre marcadores consecutivos en una columna (resp. en una fila)
- *Desplazamiento*: distancias *Horizontal* y *Vertical* desde el límite del objeto espacial
- *Recorte*: permite controlar cómo deben recortarse los marcadores en el relleno a la forma del polígono. Las opciones son:
    * *Recorte a la forma*: los marcadores se recortan para que sólo sean visibles las partes situadas dentro del polígono
    * *Centroide del marcador dentro de la forma*: sólo se dibujan los marcadores cuyo centro cae dentro del polígono, pero estos marcadores no se recortarán hacia el exterior del polígono.
    * *Marcador completamente dentro de la forma*: sólo se muestran los marcadores que caen completamente dentro del polígono
    * *Sin recorte*: cualquier marcador que se cruce con el polígono se renderizará por completo (en sentido estricto es «se cruce con el cuadro delimitador del marcador»)

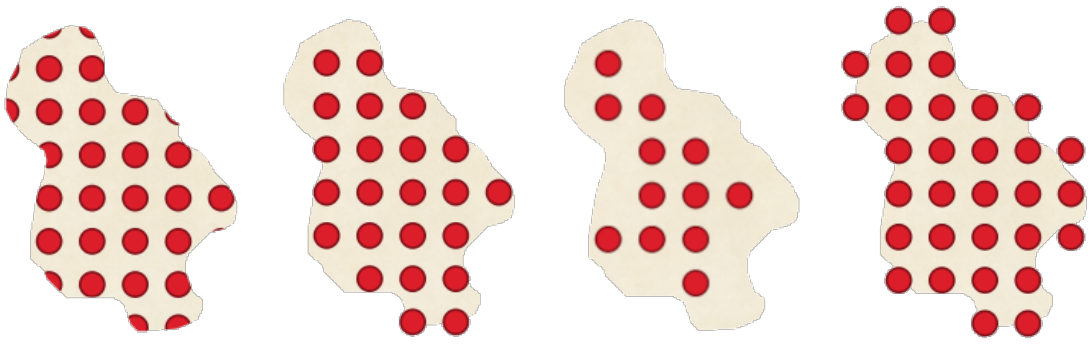


Figura 10.21: Recorte de marcadores en el relleno - De izquierda a derecha: Recorte a la forma, Centroide del marcador dentro de la forma, Marcador completamente dentro de la forma, Sin recorte

- *Rotación* de todo el patrón, en el sentido de las agujas del reloj
- El ajuste de grupo *Randomizar patrón* permite que cada punto de un relleno de patrón de puntos se desplace aleatoriamente hasta la distancia máxima especificada *Horizontalmente* o *Verticalmente*. Puede

especificar el desplazamiento máximo en cualquier unidad admitida, como milímetros, puntos, unidades de mapa o incluso «porcentaje» (donde porcentaje es relativo a la anchura o altura del patrón).

Puede establecer una semilla de número aleatorio opcional para evitar que los patrones de símbolos «salten» entre actualizaciones del mapa. También se admiten anulaciones definidas por datos.

> **Nota**
>
> La principal diferencia entre el tipo de símbolo *Aleatorizar patrón* y el *relleno de marcador aleatorio* es que el desplazamiento aleatorio con un patrón de puntos permite una colocación casi «regular» de los marcadores, ya que los puntos del patrón están limitados a una cuadrícula, lo que permite crear rellenos semialeatorios sin áreas vacías ni marcadores superpuestos. (A diferencia del relleno de marcador aleatorio, que siempre colocará los puntos de forma completamente aleatoria… lo que a veces da lugar a agrupaciones visuales de puntos o áreas vacías no deseadas).

- **Relleno de marcador aleatorio**: rellena el polígono con un *marcador de símbolo* ubicado en posiciones aleatorias dentro de los límites del polìgono. Puede establecer:
  - *Método de recuento*: si el número de símbolos de marcador a renderizar se considera como un recuento absoluto o basado en la densidad
  - *Cuento de puntos*: el número de símbolos de marcadores que hay que representar,
  - un número aleatorio opcional *semilla*, para dar una colocación consistente
  - *Área de densidad*: en el caso del método de recuento basado en la densidad, garantiza que la densidad de relleno de los marcadores siga siendo la misma en diferentes niveles de escala / zoom de los marcadores cada vez que se actualizan los mapas (también permite la colocación aleatoria para jugar bien con el servidor QGIS y la representación basada en teselas)
  - *Recortar los marcadores a los límites del polígono*: si los marcadores representados cerca de los bordes de los polígonos deben ser recortados al límite del polígono o no
- **Relleno con imagen rasterizada**: rellena el polígono con mosaicos de una imagen rasterizada (`PNG` `JPG`, `BMP` …). La imagen puede ser un archivo en el disco, una URL remota o un archivo incrustado codificado como una cadena (*más detalles*). Las opciones incluyen (datos definidos) opacidad, tamaño, modo de coordenadas (objeto o ventana gráfica), rotación y desplazamiento. En la opción Tamaño, puede ajustar de forma independiente la anchura y la altura del patrón de relleno, lo que permite rellenos rasterizados estirados en dirección horizontal o vertical. La anchura y la altura de la imagen se pueden establecer utilizando cualquiera de las *unidades comunes* o como porcentaje del tamaño original.
- **Relleno SVG**: rellena el polígono utilizando *marcadores SVG* de un tamaño determinado (*Anchura de la textura*).
- **Relleno de degradado**: rellena un polígono con un degradado que se dibuja desde el borde del polígono hacia el centro del mismo. Los parámetros configurables incluyen la distancia desde el límite hasta la sombra, el uso de rampas de color o gradientes simples de dos colores, el desenfoque opcional del relleno y los desplazamientos.
- **Outline: Arrow**: uses a line *arrow symbol* layer to represent the polygon boundary (*Rings*) which can be the interior rings only, the exterior ring only or all the rings. The settings for the outline arrow are the same as for arrow line symbols.
- **Outline: Filled line**: uses a *filled line symbol* layer to represent the polygon boundary (*Rings*) which can be the interior rings only, the exterior ring only or all the rings. The settings are the same as for filled line symbols.
- **Outline: Hashed line**: uses a *hash line symbol* layer to represent the polygon boundary (*Rings*) which can be the interior rings only, the exterior ring only or all the rings. The other settings are the same as for hashed line symbols.
- **Outline: Interpolated line**: uses an *interpolated line symbol* layer to represent the polygon boundary (*Rings*) which can be the interior rings only, the exterior ring only or all the rings. The other settings are the same as for interpolated line symbols.

- **Outline: Linear referencing**: uses a *linear referencing symbol* layer to represent the polygon boundary (*Rings*) which can be the interior rings only, the exterior ring only or all the rings. The other settings are the same as for linear referencing symbols.
- **Outline: Lineburst**: uses a *lineburst symbol* layer to represent the polygon boundary (*Rings*) which can be the interior rings only, the exterior ring only or all the rings. The other settings are the same as for lineburst symbols.
- **Outline: Marker line**: uses a *marker line symbol* layer to represent the polygon boundary (*Rings*) which can be the interior rings only, the exterior ring only or all the rings. The other settings for the outline marker line are the same as for marker line symbols.
- **Outline: Raster line**: uses a *raster line symbol* layer to represent the polygon boundary (*Rings*) which can be the interior rings only, the exterior ring only or all the rings. The other settings for the outline hashed line are the same as for hashed line symbols.
- **Outline: Simple line**: uses a *simple line symbol* layer to represent the polygon boundary (*Rings*) which can be the interior rings only, the exterior ring only or all the rings. The *Draw line only inside polygon* option displays the polygon borders inside the polygon and can be useful to clearly represent adjacent polygon boundaries. The other settings for the outline simple line are the same as for simple line symbols.

Nota

Cuando el tipo de geometría es polígono, puede optar por desactivar el recorte automático de líneas/polígonos en la extensión del lienzo. En algunos casos, este recorte da como resultado una simbología desfavorable (por ejemplo, el centroide varía donde el centroide siempre debe ser el centroide de la entidad real).

SVG parametrizable

Tiene la posibilidad de cambiar los colores de un *marcador SVG*. Tiene que añadir los marcadores de posición `param(fill)` para el color de relleno, `param(fill-opacity)` para la opacidad de relleno, `param(outline)` y `param(outline-opacity)` para el color y la opacidad del trazo respectivamente, y `param(outline-width)` para el ancho del trazo. Estos marcadores de posición pueden ir seguidos opcionalmente de un valor predeterminado, por ejemplo:

```
<svg width="100%" height="100%">
<rect fill="param(fill) #ff0000" fill-opacity="param(fill-opacity) 1" stroke=
↪"param(outline) #00ff00" stroke-opacity="param(outline-opacity) 1" stroke-width=
↪"param(outline-width) 10" width="100" height="100">
</rect>
</svg>
```

Más generalmente, SVG se puede parametrizar libremente usando `param(param_name)`. Este parámetro puede usarse como valor de atributo o como texto de nodo:

```
<g stroke-width=".265" text-anchor="middle" alignment-baseline="param(align)">
  <text x="98" y="147.5" font-size="6px">param(text1)</text>
  <text x="98" y="156.3" font-size="4.5px">param(text2)</text>
</g>
```

Los parámetros se pueden definir como expresiones en la tabla *Parámetros SVG dinámicos*.

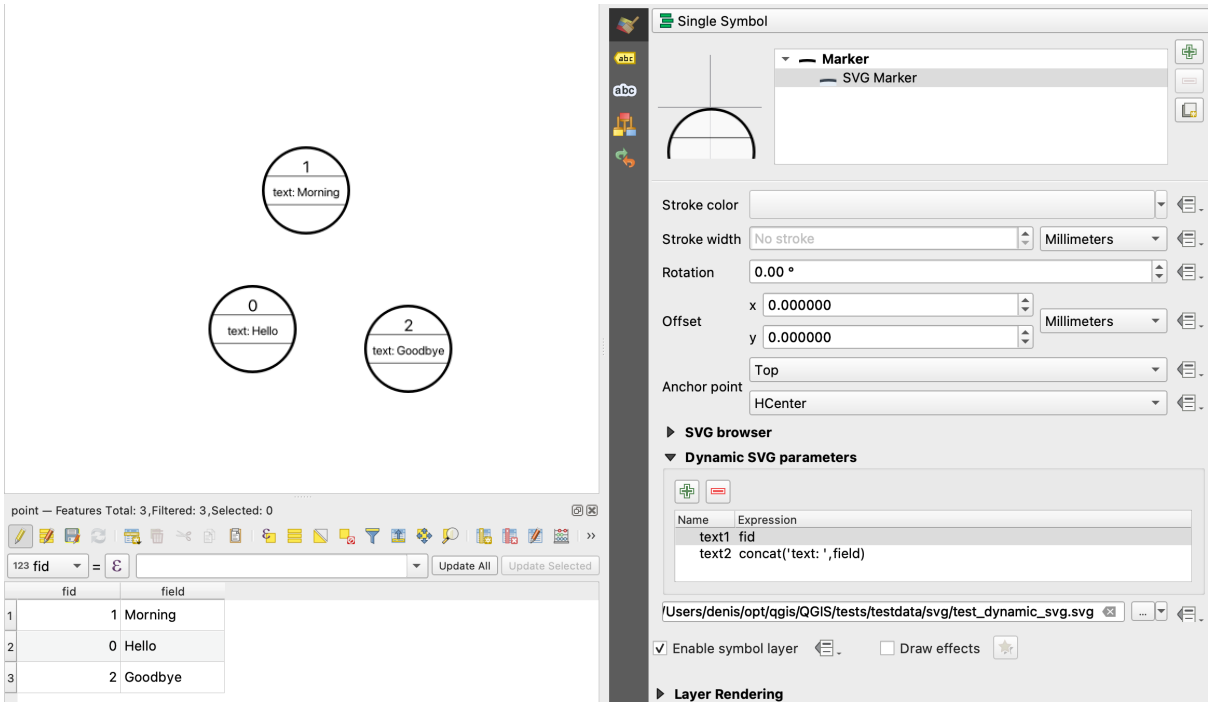


Figura 10.22: Tabla de parámetros SVG dinámicos

Nota

QGIS busca un nodo SVG completo. Así que si su parámetro se encuentra dentro de un nodo más complejo, es necesario inyectar el nodo completo con la expresión. Por ejemplo, `transform=«rotate(param(angle)»` no funcionará. En su lugar, necesitas hacer `transform=«param(rotation)»` y el parámetro `rotation` se definirá con la expresión `'rotate(' || coalesce(my_field, 0) || ')'`.

El generador de Geometría

Disponible con todos los tipos de símbolos, la capa de símbolos *generador de geometría* permite utilizar *sintaxis de expresión* para generar una geometría sobre la marcha durante el proceso de renderizado. La geometría resultante no tiene por qué coincidir con la original *Tipo de geometría* y puede añadir varias capas de símbolos modificados de forma diferente unas sobre otras.

Se puede establecer una propiedad *Unidades*: cuando el símbolo del generador de geometría no se aplica a una capa (por ejemplo, se usa en un elemento de diseño), esto permite un mayor control sobre la salida generada.

Algunos ejemplos:

- Renderizar el símbolo como el centroide de un objeto espacial

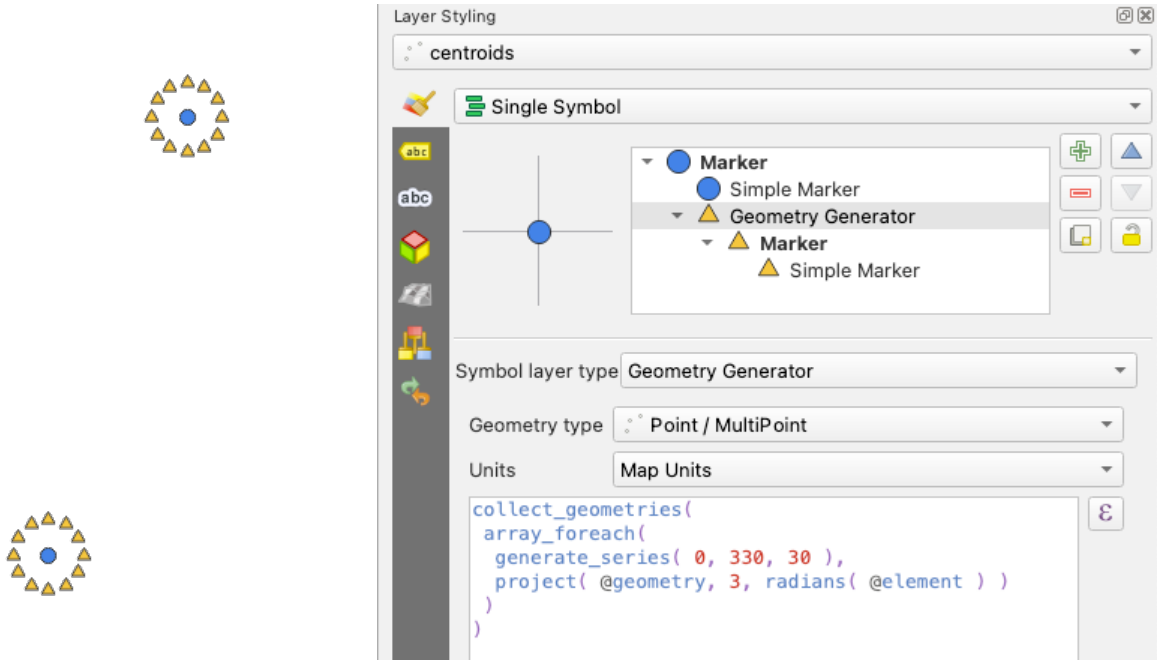
```
centroid( $geometry )
```

- Superponer visualmente los elementos situados a una distancia de 100 unidades cartográficas de un elemento puntual, es decir, generar un búfer de 100 m alrededor del punto.

```
buffer( $geometry, 100 )
```

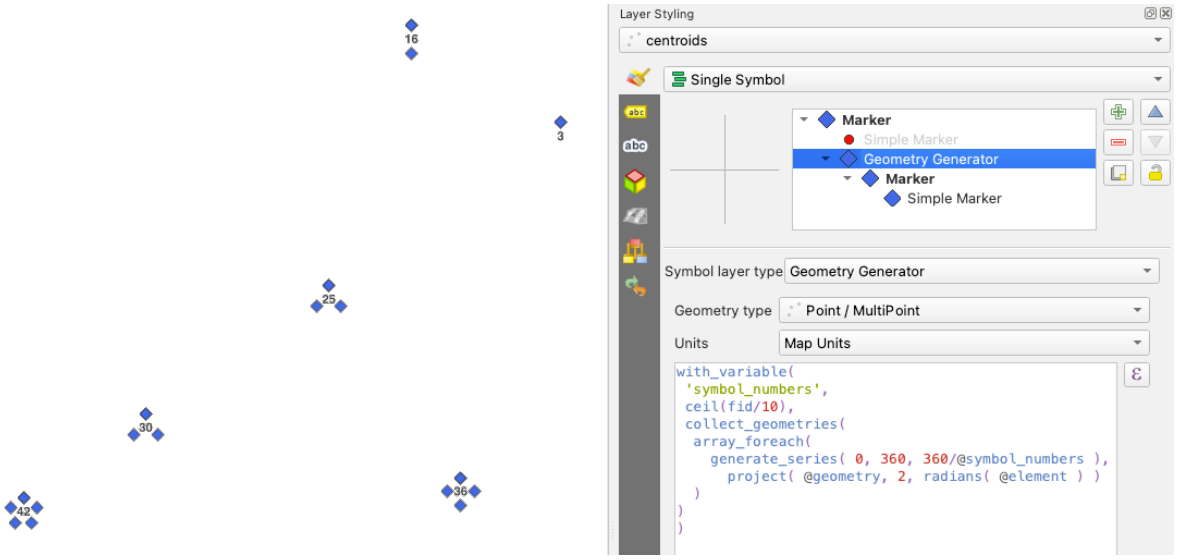
- Crea un efecto radial de puntos que rodean el punto de objeto espacial central cuando se utiliza como generador de geometría MultiPunto.

```
collect_geometries(
  array_foreach(
    generate_series( 0, 330, 30 ),
    project( $geometry, 3, radians( @element ) )
  )
)
```



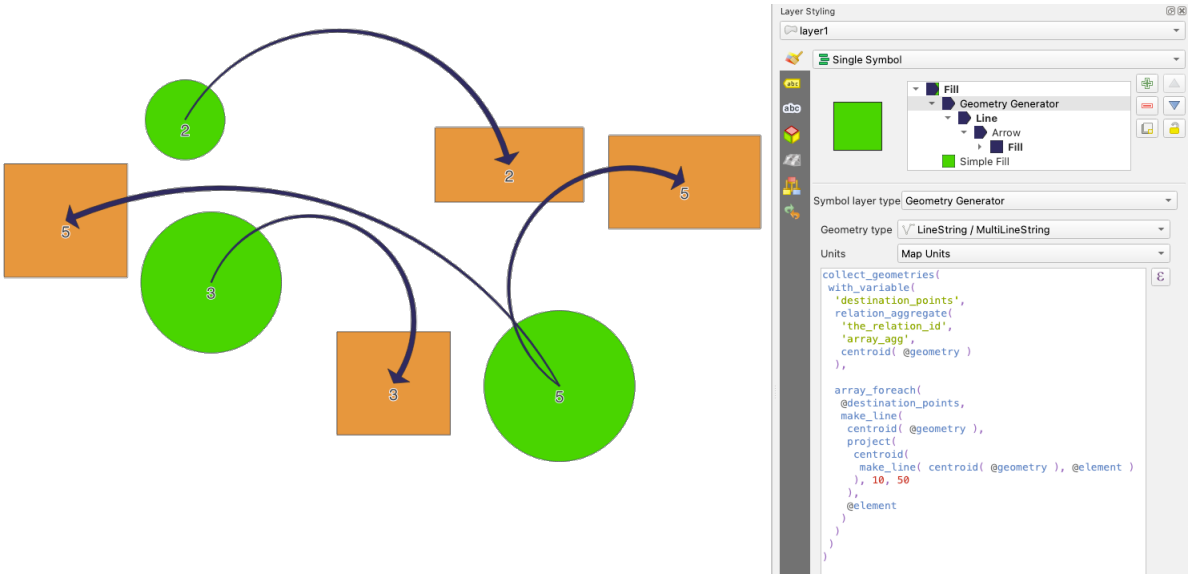
- Crea un efecto radial de puntos que rodean el punto central del objeto espacial. El número de puntos varía en función de un campo.

```
with_variable(
  'symbol_numbers',
  ceil(fid/10),
  collect_geometries(
    array_foreach(
      generate_series( 0, 360, 360/@symbol_numbers ),
      project( $geometry, 2, radians( @element ) )
    )
  )
)
```



- Crear una línea de flecha curva que conecte objetos espaciales de dos capas basándose en su *relación*

```
collect_geometries(
  with_variable(
    'destination_points',
    relation_aggregate( 'the_relation_id', 'array_agg', centroid( $geometry )
↪),
    array_foreach(
      @destination_points,
      make_line(
        centroid( @geometry ),
        project(
          centroid( make_line( centroid( @geometry ), @element ) ),
          10,
          50
        ),
        @element
      )
    )
  )
)
```



El Marcador de Campo Vectorial

El marcador de campo vectorial se usa para mostrar datos de campo vectorial tales como deformación de la tierra, flujos de marea y similares. Muestra los vectores como líneas (preferiblemente flechas) que están escaladas y orientadas de acuerdo con los atributos seleccionados de los puntos de datos. Solo se puede usar para representar datos de puntos; Las capas de línea y polígono no se dibujan con esta simbología.

El campo vectorial está definido por atributos en los datos, que pueden representar el campo ya sea por:

- componentes **cartesianas** (componentes `x` e `y` del campo)
- o coordenadas **polares**: en este caso, los atributos definen `Longitud` y `Ángulo`. El ángulo puede medirse en sentido horario desde el norte o en sentido antihorario desde el este, y puede ser en grados o radianes.
- o como datos **de altura solamente**, que muestra una flecha vertical escalada utilizando un atributo de los datos. Esto es apropiado para mostrar el componente vertical de la deformación, por ejemplo.

La magnitud del campo se puede ampliar o reducir a un tamaño apropiado para ver el campo.

Marcador animado

El tipo de símbolo de marcador animado le permite utilizar un archivo de animación `.GIF`, `.WebP`, `.MNG`, etc. para representar puntos en su mapa. Puede especificar:

- *Archivo* ruta,
- *Tasa de fotogramas*: número de pasos que se muestran por segundo, indicando la velocidad a la que se reproduce la animación,
- *Tamaño* en cualquier *unidad admitida*,
- *Opacidad*,
- *Rotación*,
- *Desplazamiento* en las direcciones *x* y *y* desde la posición del marcador,
- *Punto de anclaje*

Hay dos formas de manejar los símbolos animados:

- **Cuando su mapa no está configurado como una animación** (es decir, es un proyecto QGIS estándar sin animaciones), el marco para los marcadores animados será determinado únicamente por la marca de tiempo actual.
- **Cuando su mapa está** *configurado como una animación*, los marcadores animados se sincronizarán con la línea de tiempo de la animación. Esto significa que los marcadores animados se pausarán cuando la animación se detenga, progresarán con la animación, etc. El mapa también se redibujará según la velocidad de fotogramas establecida para la animación temporal. Este modo también se aplica al exportar una animación utilizando el controlador temporal.

10.3 Ajustando una etiqueta

Labels are textual information you can display on maps. They add details you could not necessarily represent using symbols, and may refer to vector features, raster cells, mesh elements, or simple annotations on the map… Two types of text-related items are available in QGIS:

- *Formato de Texto*: define la pariencia del texto, incluyendo *fuente, tamaño, colores*, *sombra*, *fondo*, *buffer*, …

  Pueden ser usadas para representar textos sobre el mapa (diseño(título de mapa, decoraciones, barra de escala,…), generalmente mediante el control *fuente* .

  Para crear un elemento de *Formato de Texto*:

1. Abrir el diálogo *Administrador de Estilo*
2. Activar la pestaña *Formato de Texto*

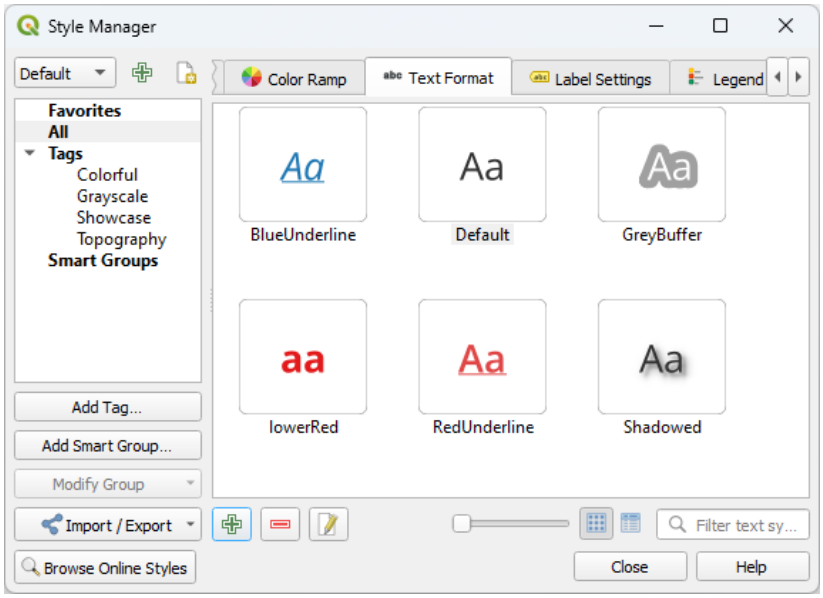


Figura 10.23: Formatos de texto en el cuadro de diálogo Administrador de estilos

3. Presione el botón Añadir elemento. El diálogo *Formato de Texto* se abre para *configurar*. Como es habitual, estas propiedades son *definibles por datos*.

- *Label Settings*: extend the text format settings with properties related to the location or the interaction with other texts or features (*callouts*, *placement*, *overlay, scale visibility*, *mask* …).

  Se utilizan para configurar el etiquetado inteligente de las capas vectoriales y de malla a través de la pestaña *Etiquetas* del diálogo *Propiedades de Capa* o del panel *Estilos de Capa* o utilizando el botón Opciones de Etiquetado de Capa de la *barra de herramientas Etiquetas*.

  Para crear un elemento *Configuración Etiquetas*:

  1. Abrir el diálogo *Administrador de Estilo*
  2. Activa la pestaña *Configuración Etiquetas*

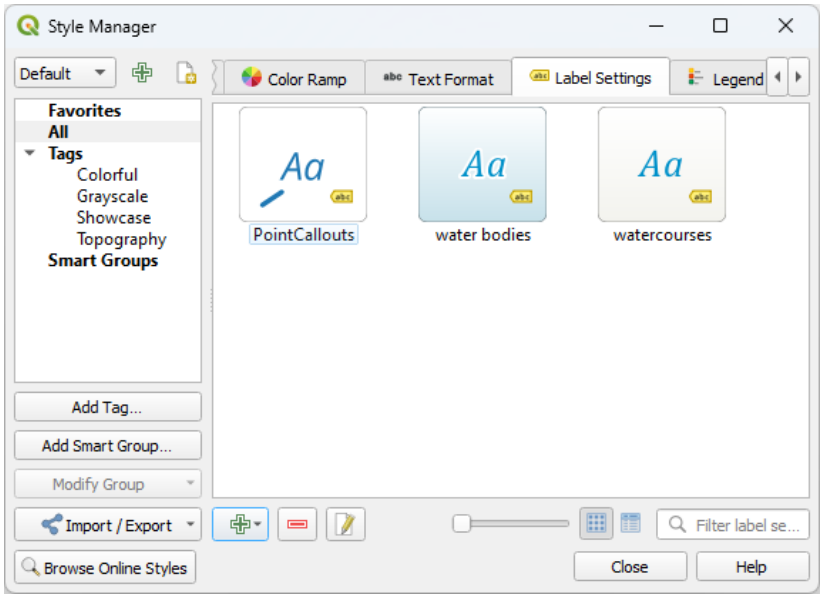


Figura 10.24: Configuración de etiquetas en el cuadro de diálogo Administrador de estilos

3. Presione el menú Agregar elemento y seleccione la entrada correspondiente al tipo de geometría de las entidades que desea etiquetar.

El cuadro de diálogo *Configuración Etiquetas* se abre con las siguientes propiedades. Como es habitual, esas propiedades son *definibles por datos*.

10.3.1 Formateando la etqueta de texto

Whether you are configuring a *Text Format* or *Label Settings* item, and depending on the type of layer you are configuring, you will be given the following options:

Tabla 10.4: Label properties by data type

Pestaña de propiedades	Formato de texto	Configuración de etiqueta	Vector or mesh layer	raster layer
Texto	✓	✓	✓	✓
Formateado	✓	✓	✓	✓
Buffer	✓	✓	✓	✓
Máscara		✓	✓	
Fondo	✓	✓	✓	✓
Sombra	✓	✓	✓	✓
Aviso		✓	✓	
Colocación		✓	✓	✓
Representación		✓	✓	✓

Atención

While for legibility, «feature» is the name used below to indicate the item being labeled, depending on the underlying layer type, it can be replaced by «pixel», «face» or «vertex».

Pestaña Texto

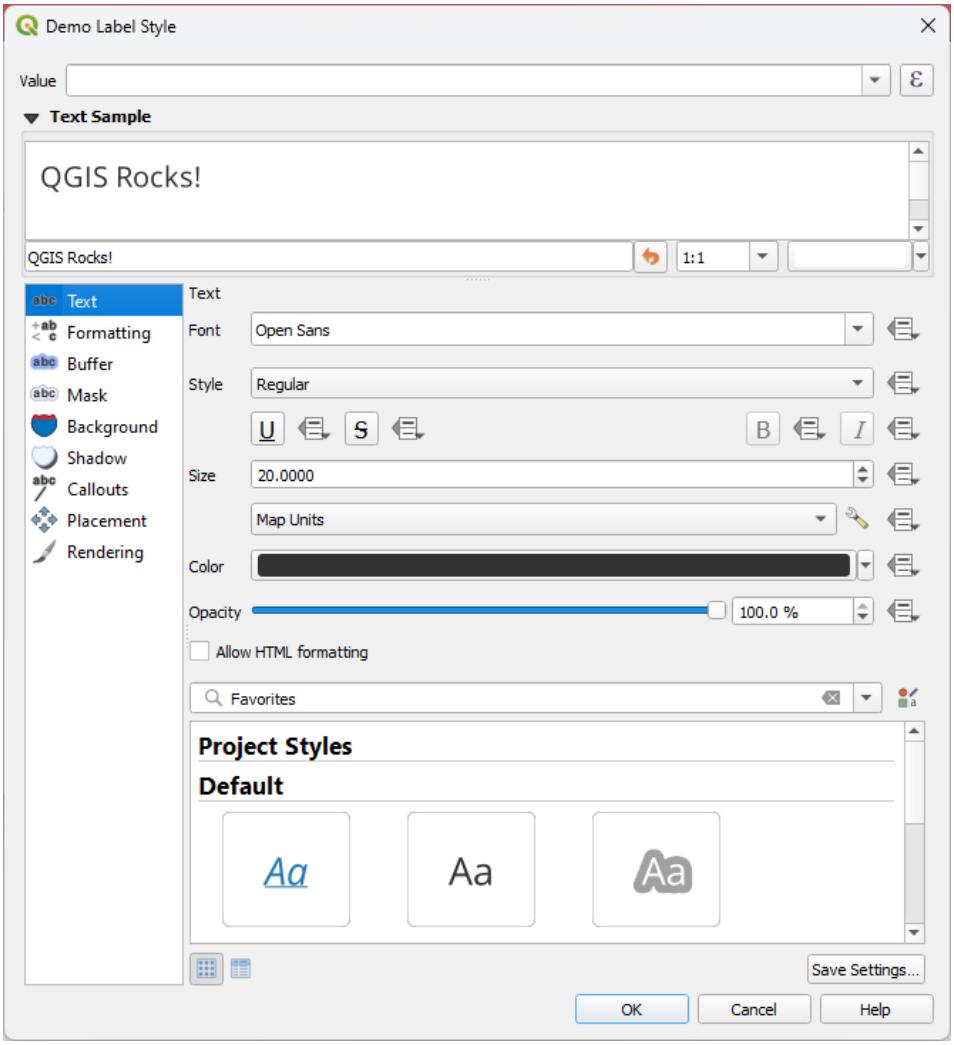


Figura 10.25: Configuración Etiqueta - Pestaña Texto

En la pestaña abc *Texto*, puede:

- la *Fuente*, de las que están disponibles en su sistema
- el *Estilo*: junto con los estilos comunes de la fuente, puede establecer si el texto debe ser subrayado o marcado
- el :Tamaño` en cualquier *supported unit*
- El *Color*
- la *Opacidad*
- y *Permitir formato HTML* permite utilizar un subconjunto de etiquetas HTML y reglas CSS para personalizar la etiqueta.

En la parte inferior de la pestaña, un widget muestra una lista filtrable de elementos compatibles almacenados en su :ref:administrador de estilo de base de datos <vector_style_manager>`. Esto le permite configurar fácilmente el formato de texto actual o la configuración de la etiqueta en función de uno existente, y también guardar un nuevo elemento en la base de datos de estilos: Presione *Guardar formato…* o *Guardar configuración…* y proporcione un nombre y etiquetas.

> **Nota**
>
> Al configurar *Configuración Etiqueta* de un elemento, los elementos de formato de texto también están disponibles en este control. Seleccione uno para sobrescribir rápidamente las actuales *propiedades textuales* de la etiqueta. Del mismo modo, puede crear/sobrescribir un formato de texto desde allí.

Permitir formateo HTML

Con *Permitir formateo HTML* activado, debe proporcionar el código HTML en el campo *Valor*. Utilice espacios en blanco en lugar de tabuladores para cualquier tipo de sangría. La expresión se analiza y cualquier etiqueta HTML soportada anula su correspondiente ajuste en las propiedades de las etiquetas. Dado que es imposible listar y detallar cada etiqueta HTML y propiedad CSS que QGIS soporta actualmente, te invitamos a explorar y probar en tus etiquetas las soportadas por la librería Qt subyacente.

Ejemplos de etiquetas HTML compatibles:

- Formateo de texto, como cursiva o negrita, por ejemplo:

```
<i>QGIS</i> <b>rocks!</b>
```

- Superíndice y subíndice, donde el texto se alineará verticalmente $^{\text{super}}$ o $_{\text{sub}}$ y se dimensionará automáticamente a 2/3 del tamaño de la fuente padre. También puede establecer un tamaño de fuente fijo para el superíndice/subíndice incluyendo reglas css, por ejemplo

```
<sup style="font-size:33pt">my superscript text</sup>
```

- Alineación horizontal del texto, utilizando en HTML el atributo `align=«xxx»` o la etiqueta `center`. Para compatibilidad con HTML5, prefiere usar la propiedad CSS `text-align`.

> **Atención**
>
> La alineación horizontal no puede utilizarse para etiquetas curvas.

- Estructurar un bloque de texto mediante etiquetas de encabezado (como `h1`, `h2`, …), o párrafos (con `p`, `div`, `br`):

```
<div class="myDiv">
  <h2>QGIS always rocks!!</h2>
  <p align="center">Let's dive into details of its nice features.</p>
</div>
```

- Inserción de imágenes: cualquier formato de imagen legible por QGIS puede utilizarse en el contenido de etiquetas HTML. Se puede servir desde rutas de archivos locales, enlaces HTTP o contenido embebido en base64, utilizando el atributo `src=«xxx»`. El tamaño de las imágenes puede especificarse mediante los atributos `width=«##»` y `height=«##»`, en la unidad `points`. Si no se especifica la anchura o la altura, se calculará automáticamente a partir del tamaño original de la imagen. Las imágenes sólo se colocan en línea, no como imágenes flotantes ni sobre etiquetas de texto curvadas.

```
<img src="qgis.png" width=40 height=60>
```

Ejemplos de propiedades CSS compatibles:

- Propiedades de fuente (`color`, `family`, `font-size`, `font-weight`, `font-style`, `word-spacing`). Ten en cuenta que `word-spacing` siempre utilizará puntos unitarios.
- Decoraciones de texto como subrayado, sobrelineado y salto de línea (`text-decoration`)
- Alineación del texto (`vertical-align`, `text-align`). La alineación horizontal no puede utilizarse para etiquetas curvas.

- Line height, in `points` or `percent` unit, e.g. «line-height: 40pt» or «line-height: 40%»
- Background properties such as `background-color` and `background-image`. They are supported for block type items (e.g. `div`) or inline items (e.g. `span`). For images, the CSS should be formatted as `background-image: url(xx)` and supports local file paths, HTTP links, or base64 embedded content.

> **Atención**
>
> Backgrounds are not supported for curved text and are always rendered above any background shape for the label, and below drop shadows/buffers properties.

- Margin properties, available for block type items only, such as `div`, `p`, `h1`,... They can be specified in either the longhand or shorthand way, in `points` unit only. Negative values can be set for the bottom margin.

```
<div class="myDiv">
  <h2 style="margin-left: 5pt; margin-right: 10pt">QGIS still rocks...</h2>
  <p style="margin: 5pt 0pt -10pt 0pt">Thanks to you!!</p>
</div>
```

Las propiedades CSS pueden establecerse en etiquetas HTML con el atributo `style`. La etiqueta HTML `span` no aplica ningún formato al texto por sí misma y es ideal si sólo quieres aplicar estilos CSS. El nombre de una propiedad CSS y su valor van separados por dos puntos (`:`). Las propiedades CSS múltiples se separan con punto y coma (`;`), por ejemplo:

```
<span style="text-decoration:underline;text-align:center;color:blue;word-spacing:20
↪">I will be displayed as blue underlined and centered text with increased space␣
↪between words</span>
```

A continuación se muestra un ejemplo de expresión y representación basada en HTML (aplica diferentes colores y subrayado a la misma etiqueta):

```
format(
  '<span style="color:blue">%1</span> ( <span style="color:red"><u>%2 ft</u></span>
↪ )',
  title( lower( "Name" ) ),
  round($length)
)
```

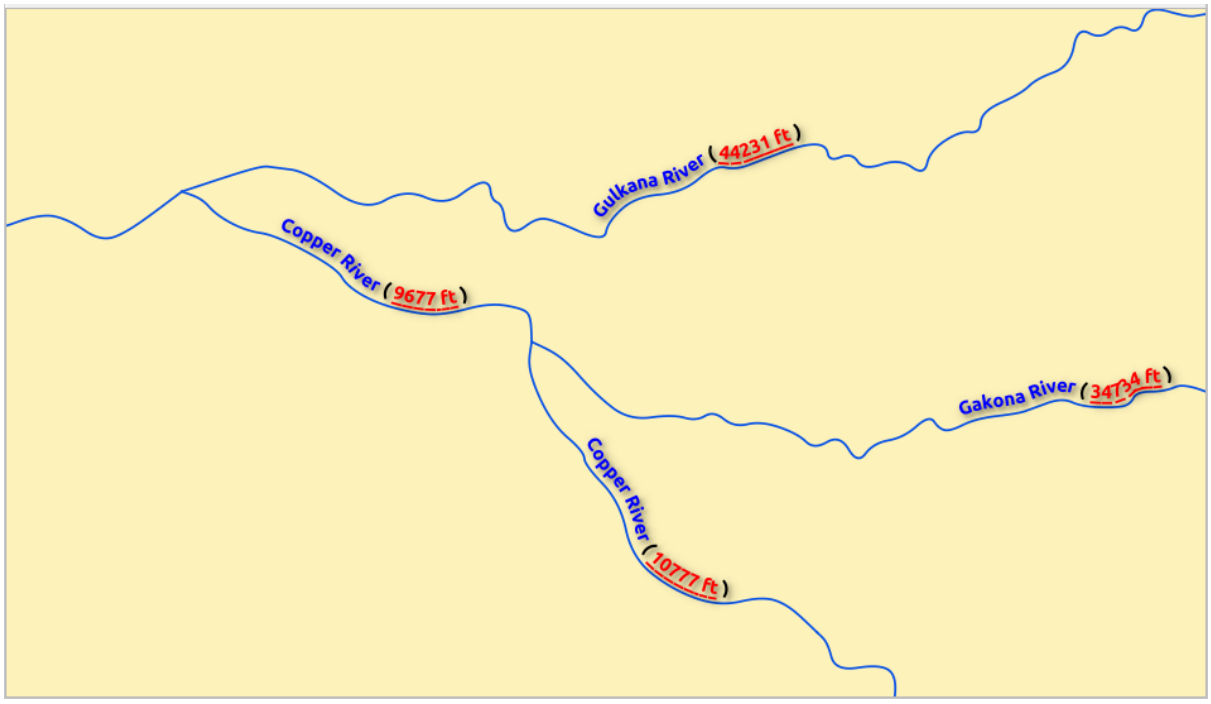


Figura 10.26: Etiquetado con formato HTML habilitado

Pestaña Formato

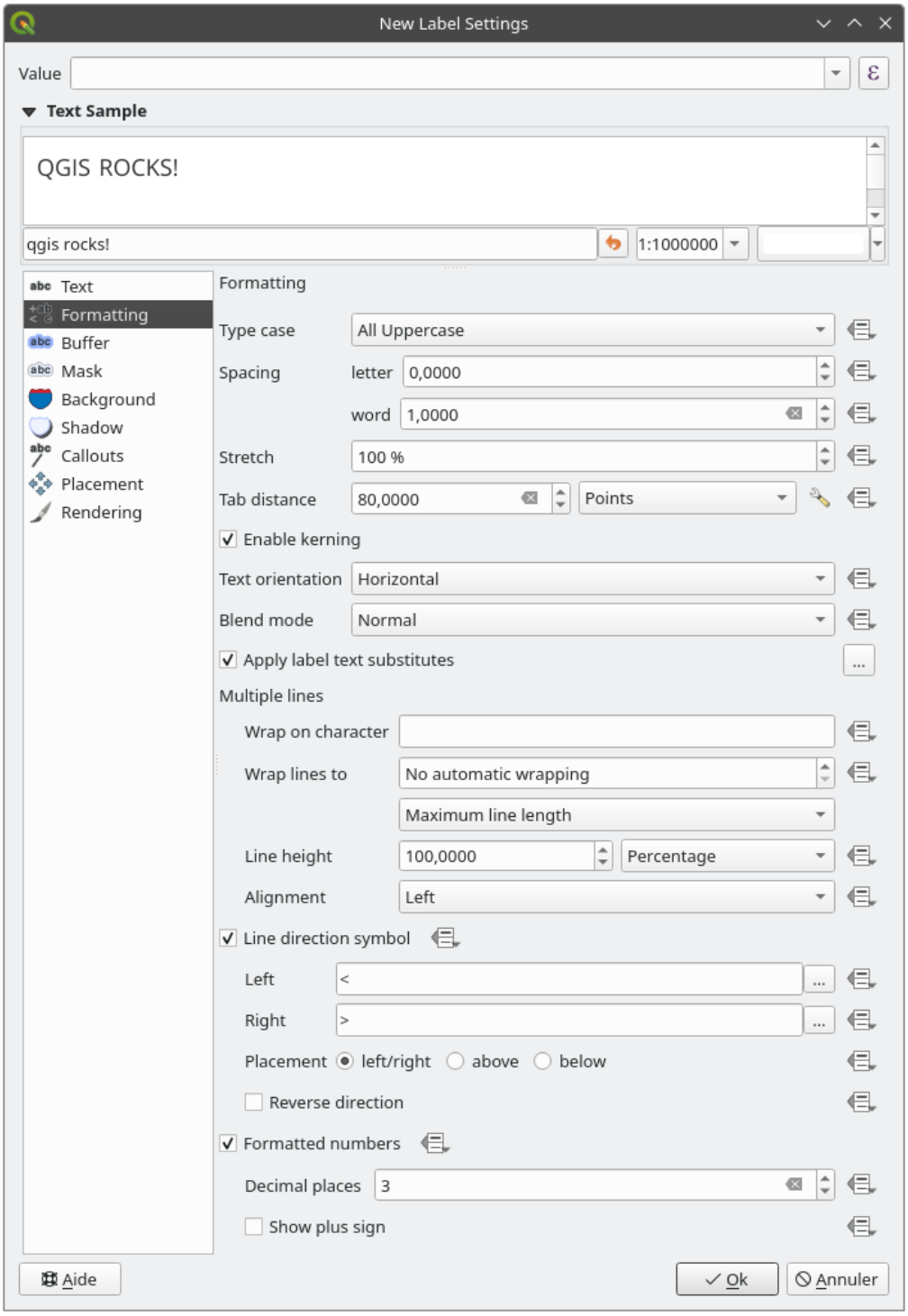


Figura 10.27: Configuración de Etiqueta - pestaña Formato

En la pestaña *Formato* ,puede:

- Usar la opción *Tipo de envoltura* para cambiar el estilo de uso de mayúsculas del texto. Tiene la posibilidad de renderizar el texto como:
  - *Sin cambios*

  - *Todo Mayúsculas*
  - *Todo Minúsculas*
  - *Small Caps*: renders lowercase characters as small caps.
  - *All Small Caps*: renders all characters as small caps regardless of their original case.
  - *Cajón de Título*: modifica la primera letra de cada palabra en mayúscula y convierte las otras letras en minúsculas si el texto original usa un solo tipo de letra. En el caso de casos de tipos mixtos en el texto, las otras letras se dejan intactas.
  - *Forzar la primera letra a mayúsculas*: modifica la primera letra de cada palabra en mayúscula y deja intactas las demás letras del texto.

- Debajo en *Espaciado*, cambie el espacio entre palabras y entre letras individuales.
- Con *Distancia de tabulación* puede ajustar el espaciado de sustitución para los caracteres `Tab` de la etiqueta, en cualquier unidad admitida. Esto permite, por ejemplo, separar correctamente o alinear verticalmente los elementos de un texto de leyenda de diseño.

  Pressing the Configure tab stops button on the right, you can add a list of tab positions, instead of a single distance. This allows e.g. the creation of table-like multiline labels, where texts are split over different «columns» and lined up nicely, based on the `Tab` characters.
- *Estirar* ratio: permite estirar o condensar horizontalmente el texto según un factor. Resulta útil para ajustar la anchura de las fuentes para que quepa un poco más de texto en las etiquetas.
- *Habilitar kerning* de la fuente de texto
- Ajuste la *Orientación de Texto* que puede ser *Horizontal* o *Vertical*. También puede ser *Basado en rotación* al configurar una etiqueta (p.ej. etiquetar correctamente las entidades de línea en modo de emplazamiento *paralelo*).
- Use la opción *Modo de fusión* para determinar cómo se mezclarán sus etiquetas con las entidades del mapa debajo de ellas (más detalles en *Modos de Mezcla*).
- La opción *Aplicar sustitutos de texto de etiqueta* le permite especificar una lista de textos para sustituir a los textos en etiquetas de características (por ejemplo, abreviando tipos de calles). Los textos de reemplazo se usan cuando se muestran etiquetas en el mapa. Los usuarios también pueden exportar e importar listas de sustitutos para facilitar la reutilización y el intercambio.
- Configurar *Multiples líneas*:
  - Establezca un carácter que forzará un salto de línea en el texto con la opción *Ajustar al carácter*
  - Establezca un tamaño de línea ideal para el ajuste automático utilizando la opción *Ajustar líneas a*. El tamaño puede representar ya sea *Longitud máxima de línea* o *Longitud mínima de línea*.
  - Decidir la *Altura de línea*: los valores pueden establecerse en *Milímetros*, *Puntos*, *Píxeles*, *Porcentaje* o *Pulgadas*. Cuando la altura de línea se establece en porcentaje, es el porcentaje del espaciado de línea de texto predeterminado de esa familia de fuentes. Normalmente es de 1,2 a 1,5 veces el tamaño del texto.
  - Formatear la *Alineación*: valores típicos disponibles son *Izquierdo*, *Derecho*, *Justificado* y *Centrado*.

    Al establecer las propiedades de las etiquetas de puntos, la alineación del texto también puede ser *Seguir la ubicación de la etiqueta*. En ese caso, la alineación dependerá de la colocación final de la etiqueta en relación con el punto. Por ejemplo, si la etiqueta se coloca a la izquierda del punto, la etiqueta se alineará a la derecha, mientras que si se coloca a la derecha, se alineará a la izquierda.

> **Nota**
>
> El formato de:guilabel:*Multiple lines* aún no es compatible con la *label placement* basada en curvas. A continuación, las opciones se desactivarán.

- Para las etiquetas de línea puede incluir *Símbolo de dirección de línea* para ayudar a determinar las direcciones de línea, con símbolos que se usarán para indicar *Izquierda* o *Derecha*. Funcionan particularmente bien cuando se usan con las opciones de colocación *curva* o *Paralela* de la pestaña *Colocación*. Hay opciones para establecer la posición de los símbolos y *Dirección inversa*.
- Utilice la opción *Números formateados* para formatear textos numéricos. Puede establecer el número de *lugares decimales*. Por defecto, se usarán 3 decimales. Utilicee *Mostrar signo más* si desea mostrar el signo más para números positivos.

Pestaña Buffer

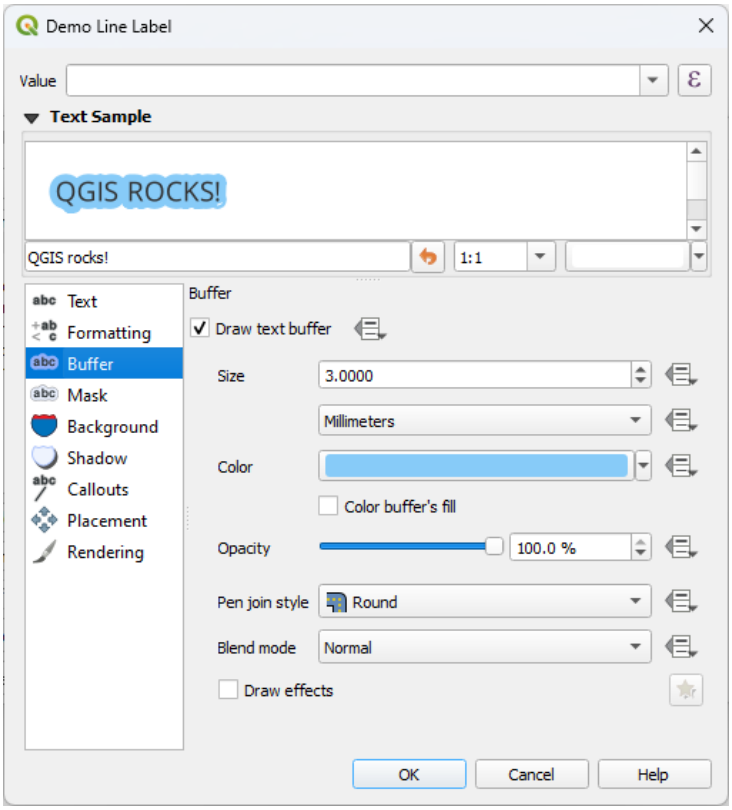


Figura 10.28: Configuración Etiquetas - pestaña Buffer

Para crear un búfer alrededor de la etiqueta, active la casilla de verficación *Dibujar buffer de texto* en la pestaña *Buffer*. Entonces tú puedes:

- Establecer el buffer *Tamaño* en cualquier *supported unit*
- Seleccione *Color* del buffer
- *Solor de relleno del buffer*: el buffer se expande desde el contorno de la etiqueta, por lo tanto, si la opción está activada, el interior de la etiqueta se llena. Esto puede ser relevante cuando se usan etiquetas parcialmente transparentes o con modos de fusión no normales, lo que permitirá ver detrás del texto de la etiqueta. Desmarcar la opción (mientras usa etiquetas totalmente transparentes) le permitirá crear etiquetas de texto resumidas.
- Definir la *Opacidad* de buffer
- Apliar un *Estilo de unión con bolígrafo*: puede ser :guilabel:` Redondo`, *Inglete* o: guilabel:` Biselado`
- Use la opción *Modo de fusión* para determinar cómo se mezclará el búfer de su etiqueta con los componentes del mapa debajo de ellos (más detalles en *Modos de Mezcla*).

- Marque *Dibujar Efectos* para añadir *paint effects* avanzados para mejorar la legibilidad del texto, por ejemplo, a través de brillos y desenfoques externos.

Pestaña de Telón de Fondo

La pestaña *Fondo* le permite configurar una forma que se mantiene debajo de cada etiqueta. Para agregar un fondo, active la casilla de verificación *Dibujar fondo* y seleccione el tipo de *Forma*. Puede ser:

- una forma regular como *Rectangle*, *Square*, *Circle* o *Ellipse* usando las propiedades completas de un *fill symbol*
- un símbolo *SVG* de un archivo, una URL o incrustado en la base de datos del proyecto o estilo (*more details*)
- o un *Marker Symbol* que puede crear o seleccionar desde *symbol library*.

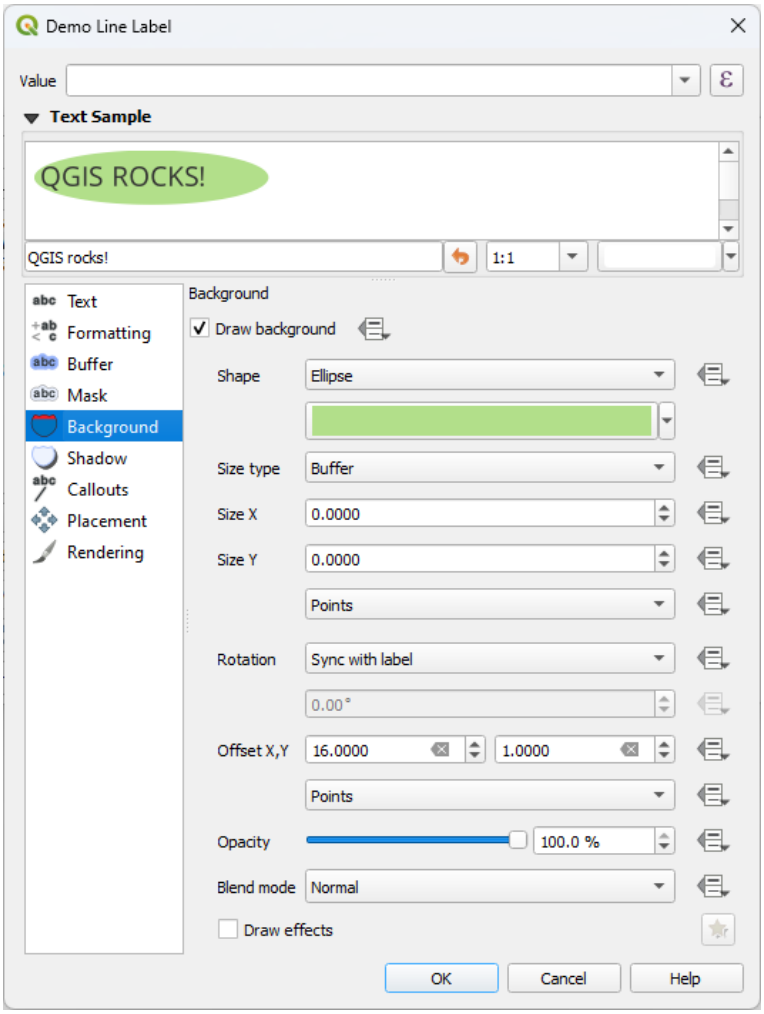


Figura 10.29: Configuración Etiquetas - pestaña Fondo

Dependiendo de la forma seleccionada, debe configurar algunas de las siguientes propiedades:

- El *Tipo de tamaño* del marco, que puede ser:
  - *Fijado*: usando el mismo tamaño para todas las etiquetas, independientemente del tamaño del texto
  - o un *Sombreado* sobre el cuadro delimitador del texto
- El *Tamaño* del marco en direcciones X e Y, usando cualquier *supported units*

- Una *Rotación* del fondo, entre *Sincronizar con etiqueta*, *Desplazamiento de etiqueta* y *Fijo*. Los dos últimos requieren un ángulo en grados.
- Un *Desplazamiento X,Y* para cambiar el fondo del elemento en las direcciones X y/o Y
- Un *Radio X,Y* para redondear las esquinas de la forma de fondo (solo aplicable a formas rectangulares o cuadradas)
- Una *Opacidad* del fondo
- Un *modo Combinar* para mezclar el fondo con los otros elementos en la representación (Ver *Modos de Mezcla*).
- Para el símbolo SVG, puede usar sus propiedades predeterminadas (*Load symbol parameters*) o establecer un *Fill color*, *Stroke color* y *Stroke width* personalizado.
- *Dibujar efectos* para añadir *efectos de pintura* avanzados para mejorar la legibilidad del texto, por ejemplo a través de brillos y difuinaciones externas.

Sección Sombra

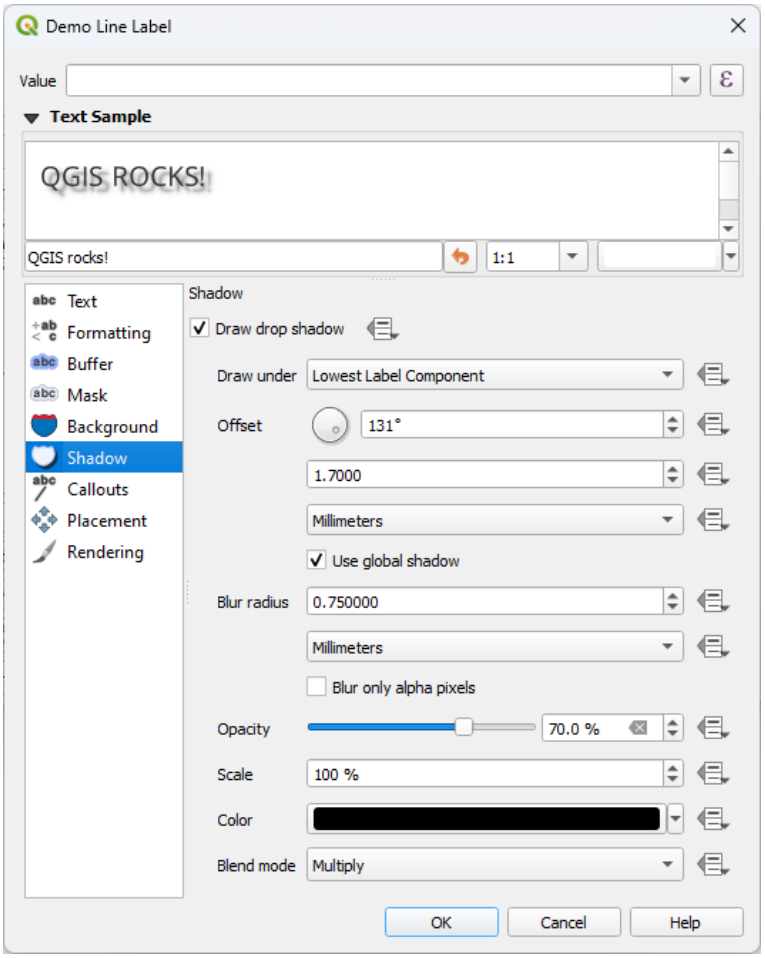


Figura 10.30: Configuración Etiquetas - pestaña Sombra

Para añadir una sombra al texto, activa la pestaña *Sombra* y activa *Dibujar sombra paralela*. Entonces puede:

- Indique el elemento utilizado para generar la sombra con *Dibujar debajo*. Puede ser *Componente de etiqueta más bajo* o un componente particular como *Texto* en sí, el *Buffer* o el *Fondo*.
- Establece el *Desplazamiento* de la sombra con respecto al elemento sombreado:

  - El ángulo: giro horario, depende de la orientación del elemento subyacente
  - La distancia de desplazamiento desde el elemento sombreado
  - Las unidades de desplazamiento

  Si marca el *Usar sombra global*, entonces el punto cero del ángulo siempre está orientado hacia el norte y no depende de la orientación del elemento de la etiqueta.
- Influir en la apariencia de la sombra con *Radio de desenfoque*. Cuanto mayor sea el número, más suaves serán las sombras, en las unidades que elija.
- Define la *Opacidad* de la sombra
- Reescale el tamaño de la sombra usando el factor de *Escala*
- Selecciona el :guilabel:`Color`de la sombra
- Utilice la opción *modo Blend* para determinar cómo se mezclará la sombra de su etiqueta con los componentes del mapa bajo ella (más detalles en *Modos de Mezcla*).

10.3.2 Configurando la interacción con etiquetas

Además de la configuración de formato de texto expuesta anteriormente, también puede establecer cómo las etiquetas interactúan entre sí o con las funciones.

Pestaña máscara

La pestaña *Máscara* le permite definir un área de máscara alrededor de las etiquetas. Esta prestación es muy útil cuando se superponen símbolos y etiquetas con colores similares, y se desea que las etiquetas sean visibles. Una máscara de etiqueta impide que los objetos espaciales especificados se dibujen dentro del límite establecido para la máscara. Por ejemplo, puede configurar una máscara de etiqueta para que una capa específica no dibuje a menos de 2 mm de la etiqueta, pero permitir que los objetos espaciales de otra capa sigan mostrándose. Las máscaras de etiquetas son similares a las memorias intermedias de etiquetas en el sentido de que permiten controlar la legibilidad de las etiquetas que cubren otros objetos espaciales. La memoria intermedia de etiquetas dibuja encima de los objetos espaciales subyacentes, mientras que la máscara de etiquetas impide de forma selectiva que se dibujen otras capas.

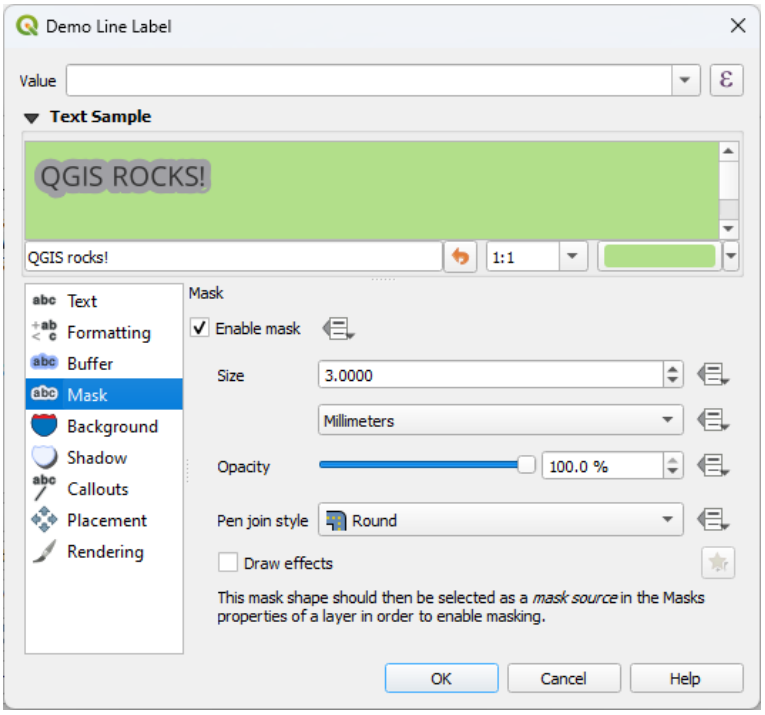


Figura 10.31: Configuración de etiquetas - pestaña Máscara (con la muestra de texto mostrando un fondo verde que representa otra capa excluida)

Para crear efectos de máscara en etiquetas:

1. Activa el checkbox *Habilitar Máscara* en la pestaña .
2. Luego puede ajustar:
   - El *Tamaño* de máscara en las *unidades soportadas*
   - la *Opacidad* del área de la máscara alrededor de la etiqueta
   - un *Estilo de Unión de Lápiz*
   - *efectos de pintura* con el checkbox *efectos de Dibujo*.
3. Seleccione esta forma de máscara como fuente de máscara en las propiedades de la capa superpuesta *pestaña Máscara* (ver *Propiedades de Máscaras*).

Sección Callouts

Una práctica común al colocar etiquetas en un mapa lleno de gente es usar **leyendas**: las etiquetas que se colocan fuera (o desplazadas) de su entidad asociada se identifican con una línea dinámica que conecta la etiqueta y la entidad. Si se mueve uno de los dos extremos (la etiqueta o la entidad), se vuelve a calcular la forma del conector.



Figura 10.32: Etiquetas con varias configuraciones de Callouts

Para añadir un Callout a una etiqueta, activa la pestaña *Callouts* y activa el *Draw Callouts*. Ahora puedes;

1. Selecciona el *Estilo* del conector, uno de:
   - *líneas simples*: una línea recta, la ruta mas corta
   - *Estilo Manhattan*: una línea quebrada a 90°
   - *Curved lines*: una línea curva
   - *Balloons*: una burbuja de diálogo que rodea la etiqueta y apunta a la característica. Puede tener esquinas redondeadas.
2. Establezca las propiedades de la llamada. La siguiente tabla muestra las diferentes propiedades, con descripción y estilo de conector compatible.

Tabla 10.5: Propiedades de las etiquetas de llamada

Propiedad	Estilo de llamada	Descripción
Estilo de relleno	Globos	Un *símbolo de relleno* con todas las funciones de visualización, incluidos efectos de capa, ajustes definidos por datos, … para dibujar la forma del globo.
Radio de la esquina		:guilabel:`Radio de la esquina`Radio de la esquina de la burbuja de diálogo
Ancho del borde		Establece el tamaño de la burbuja de conexión de voz con el puntero de la función
Márgenes		Márgenes alrededor del texto de la etiqueta, en la unidad que elija
Estilo de línea	Todo menos globos	Un *símbolo de línea* con todas las funciones de visualización, incluidos efectos de capa, ajustes definidos por datos, … para dibujar la línea del conector.

continúe en la próxima página

Tabla 10.5 – proviene de la página anterior

Propiedad	Estilo de llamada	Descripción
Curvatura	Líneas curvas	El porcentaje de curvatura de la línea de conexión
Orientación		Orientación, partiendo de la etiqueta hacia la característica. Puede ser *Sentido horario*, *Sentido antihorario*, o *Automático* (determinando una orientación óptima para cada etiqueta).
Longitud mínima	Todo menos globos	Longitud mínima de la línea de conexión
Dibujar líneas a todas las partes del objeto espacial		En el caso de un elemento con varias partes, indica si debe trazarse una línea de conexión desde la etiqueta hasta cada parte de la geometría.
Punto de anclaje de la etiqueta.		Controla dónde debe unirse la línea del conector con el texto de la etiqueta. Opciones disponibles: • *Punto mas Cercano* • *Centroide* • Posición fija en el borde (*Arriba izquierda*, *Arriba centro*, *Arriba derecha*, *Medio izquierda*, *Medio derecha*, *abajo izquierda*, *Abajo centro* and *Abajo derecha*).
Desplazamiento desde el área de la etiqueta		Controla la distancia desde el punto de anclaje de la etiqueta (donde termina la línea de llamada). De este modo se evita trazar líneas pegadas al texto.
Desplazamiento del objeto espacial	Todo	Controla la distancia desde el objeto espacial (o su punto de anclaje si es un polígono) donde terminan las líneas de llamada. De este modo, por ejemplo, se evita trazar líneas en los bordes de los elementos.
Punto de anclaje del objeto espacial		Dónde termina la línea del conector en el objeto espacial (polígono). Opciones disponibles: • *Polo de inaccesibilidad* • *Punto en el exterior* • *Punto en la superficie* • *Centroide*
Marcador de punto final	Globos	Un *símbolo de marcador* con capacidades de visualización completas, incluidos efectos de capa, definidos por datos, … para representar un símbolo de marcador debajo del punto final de la llamada del globo.
Modo de mezcla	Todo	Controla la *mezcla* de la llamada.

En el grupo *Data defined placement*, se pueden controlar las coordenadas de los puntos *Origin* (en el lado de la etiqueta) y/o *Destination* (en el lado de la característica) de la llamada. Las llamadas también se pueden controlar manualmente mediante Move Label, Diagram or Callout en la herramienta *Labeling Toolbar*. Los puntos inicial y final de cada llamada se pueden mover de esta manera. Los nodos deben resaltarse cuando el puntero del mouse está cerca. Si es necesario, se puede mantener presionada la tecla `Shift` durante el movimiento. Esto ajustará el punto de manera que el ángulo entre los dos puntos de llamada se incremente en 15 grados.

Pestaña Ubicación

Elija la pestaña  *Colocación* para configurar la colocación de etiquetas y la prioridad de etiquetado. Tenga en cuenta que las opciones de colocación difieren según el tipo de capa vectorial o de malla, a saber, punto, línea o polígono, y se ven afectadas por el global *ajuste PAL*.

Ubicación para capas de puntos

Los modos de colocación para etiquetas de puntos disponibles son:

- *Cartografico*: las etiquetas de puntos son generadas con una mejor relación visual con el objeto puntual, siguiendo las reglas ideales de ubicación cartográfica. Las etiquetas se pueden colocar:
  - a una *Distancia* establecida en *unidades soportadas*, ya sea desde la propia entidad puntual o desde los límites del símbolo utilizado para representar la entidad (ajustada en *Desplazamiento de distancia desde*). La última opción es especialmente útil cuando el tamaño del símbolo no es fijo, p. Ej. si está establecido por un tamaño definido por datos o cuando se usan diferentes símbolos en un renderizador *categorizado*.
  - dentro de una *Distancia máxima* del elemento, que es un parámetro opcional que permite controlar la distancia a la que puede colocarse una etiqueta del elemento que está etiquetando. Este parámetro funciona junto con *Distancia* para crear un rango para la colocación de etiquetas, añadiendo flexibilidad para colocar las etiquetas de forma más eficaz, especialmente en mapas con mucho tráfico, asegurando que se ajustan perfectamente alrededor de sus elementos correspondientes.
  - utilizando la opción *Priorizar colocación*, que decide qué es más importante a la hora de colocar las etiquetas. Hay dos opciones:
    - *Preferir etiquetas más cercanas*: Por defecto, las etiquetas se mantienen cerca del objeto espacial.
    - *Preferir orden de posición*: La etiqueta intentará permanecer en una posición específica (como arriba a la izquierda o arriba a la derecha), incluso si está un poco más lejos de la característica. La etiqueta sólo se mueve a otras posiciones si no hay espacio dentro de la distancia máxima en su posición preferida.
  - siguiendo una *Prioridad de posición* que dicta los candidatos de colocación para las etiquetas de anclaje alrededor y (centradas) sobre la característica de punto, y el orden en que se prueban las posiciones. El orden predeterminado, basado en las guías de Krygier y Wood (2011) y otros libros de texto cartográficos, es el siguiente:
    1. superior derecha
    2. superior izquierda
    3. inferior derecha
    4. inferior izquierda
    5. centro derecha
    6. centro izquierda
    7. superior, ligeramente a la derecha
    8. inferior, ligeramente a la izquierda

    Utilizando el botón  Sustitución definida por datos, puede proporcionar una lista separada por comas de colocaciones por orden de prioridad. Esto también permite que sólo se utilicen ciertas colocaciones, sólo para ciertos objetos espaciales, por ejemplo, para los objetos espaciales costeros puede evitar que las etiquetas se coloquen sobre la tierra.
- *Alrededor del punto*: las etiquetas se colocan en un círculo alrededor del objeto con el mismo radio establecido en *Distancia*. Además, puede definir *Distancia máxima* del elemento, para controlar la distancia a la que puede colocarse una etiqueta del elemento que está etiquetando. La prioridad de colocación es en el sentido de las

agujas del reloj desde la «parte superior derecha». La posición puede limitarse utilizando la opción *Cuadrante* definida por los datos.

- *Desplazamiento desde punto*: Las etiquetas se colocan a una distancia de *Desplazamiento X, Y* desde la entidad puntual, en varias unidades, o preferiblemente sobre la entidad. Puede utilizar un *Cuadrante* definido por datos para restringir la ubicación y puede asignar una *Rotación* a la etiqueta.

Emplazamiento para capas de líneas

Modos de Etiqueta para capas lineales incluyen:

- *Paralela*: dibuja la etiqueta paralela a una línea generalizada que representa la entidad, con preferencia para colocarla sobre porciones más rectas de la línea. Puede definir:
  - *Posiciones Permitidas*: *SObre línea*, *En la línea*, *Bajo la línea* and *Posición dependiente de la Orientación de la línea* (colocando la etiqueta a la izquierda o a la derecha de la línea). Es posible seleccionar varias opciones a la vez. En ese caso, QGIS buscará la posición óptima de la etiqueta.
  - *Distancia* entre la etiqueta y la línea
- *Curvada*: dibuja la etiqueta siguiendo la curvatura de la entidad lineal. Además de los parámetros disponibles con el modo *Paralelo*, puede configurar *Ángulo máximo entre caracteres curvos*, ya sea dentro o fuera.
- *Horizontal*: dibuja etiquetas horizontalmente a lo largo de la entidad de línea.

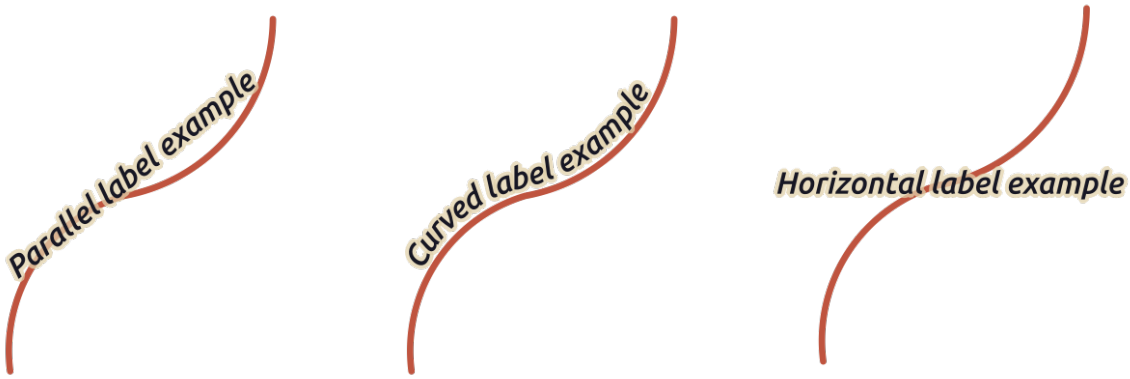


Figura 10.33: Ejemplos de colocación de etiquetas para líneas

Junto a los modos de ubicación, puede configurar:

- *Etiquetas repetidas Distancia* para mostrar varias veces la etiqueta a lo largo de la entidad. La distancia puede ser en `Milímetros`, `Puntos`, `Píxeles`, `Metros a escala`, `Unidades de mapa` y `Pulgadas`.
- Una *Rebasamiento de etiqueta Distancia* (no disponible para el modo horizontal): especifica la distancia máxima permitida que una etiqueta puede recorrer más allá del final (o comienzo) de las entidades de línea. El aumento de este valor puede permitir que se muestren etiquetas para entidades de línea más cortas.
- *Anclado de Etiqueta*: controla la ubicación de las etiquetas a lo largo de la característica de línea a la que hacen referencia. Haga clic en *Configuración…* para elegir:
  - la posición a lo largo de la línea (como una proporción) a la que se colocarán las etiquetas cerca. Puede estar definido por datos y los valores posibles son:
    -  *Centro de la línea*
    -  *Comienzo de la línea*
    -  *Fin de la línea*

    - or  *Personalización…*.

- *Clipping*: Determina cómo se calcula la ubicación de la etiqueta en una línea. De forma predeterminada, solo se usa la extensión visible de la línea, pero se puede usar toda la extensión para obtener resultados más consistentes.
- *Texto de anclaje*: controla qué parte del texto (inicio, centro o final) se alineará con el punto de anclaje. El uso de *Anclaje automático* significa que:
    - Para las etiquetas ancladas cerca del inicio de la línea (0-25%), la colocación del ancla será el **inicio** del texto de la etiqueta
    - Para las etiquetas ancladas cerca del final de la línea (75-100%), la colocación del ancla será el **final** del texto de la etiqueta
    - Para las etiquetas ancladas cerca del centro de la línea (25-75%), la colocación del ancla será el **centro** del texto de la etiqueta.
- *Comportamiento de ubicación*: use *Sugerencia de ubicación preferida* para tratar el ancla de etiqueta solo como una sugerencia para la ubicación de la etiqueta. Al elegir *Estricto*, las etiquetas se colocan exactamente en el ancla de la etiqueta.

Ubicación para capas de polígonos

Puede elegir uno de los siguientes modos para ubicar etiquetas de polígonos:

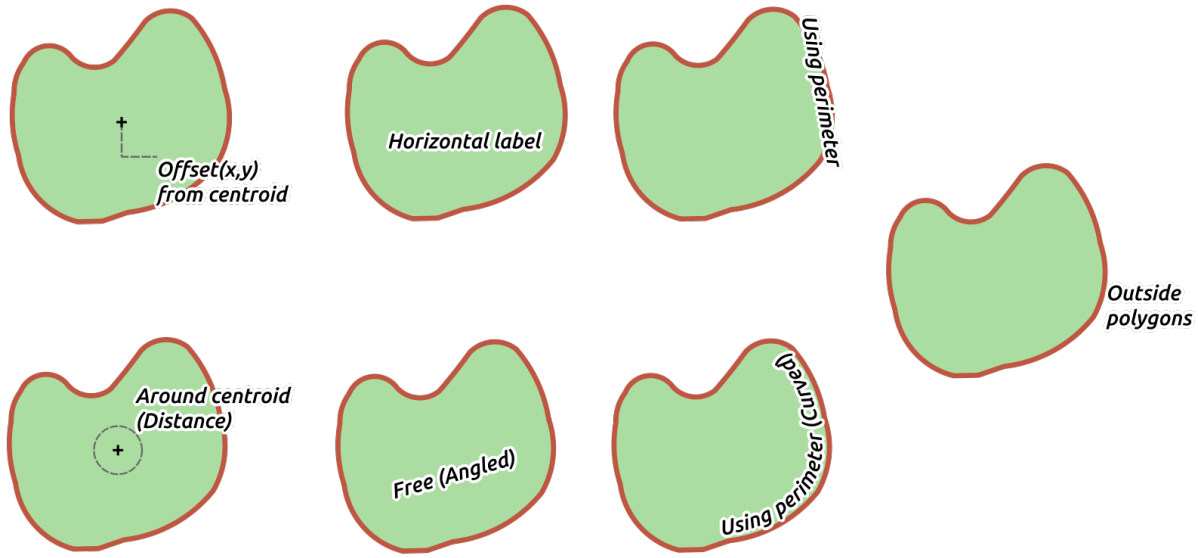


Figura 10.34: Ejemplos de emplzamiento de etiquetas para polígonos

- *Desplazamiento del centroide*: las etiquetas se colocan sobre el centroide de la entidad o en una distancia fija *Desplazamiento X, Y* (en *unidades admitidas*) del centroide. El centroide de referencia se puede determinar en función de la parte del polígono representada en el lienzo del mapa (*polígono visible*) o el *polígono completo*, sin importar si puede verlo. Tú también puedes:
    - forzar que el punto del centroide caiga dentro de su polígono
    - ubicar la etiqueta dentro de un cuadrante específico
    - asignar una rotación
    - *Permitir colocar etiquetas fuera de polígonos* cuando no sea posible colocarlas dentro del polígono. Gracias a las propiedades definidas por los datos, esto hace posible permitir etiquetas externas, prevenir etiquetas externas o forzar etiquetas externas entidad por entidad.

- *En torno a Centroide*: coloca la etiqueta dentro de una distancia preestablecida alrededor del centroide, con preferencia por la ubicación directamente sobre el centroide. Nuevamente, puede definir si el centroide es el del *polígono visible* o el *polígono completo*, y si forzar el punto del centroide dentro del polígono.
- *Horizontal*: coloca en la mejor posición una etiqueta horizontal dentro del polígono. La ubicación preferida es más alejada de los bordes del polígono. Es posible *Permitir colocar etiquetas fuera de polígonos*.
- *Libre (Angulado)*: coloca en la mejor posición una etiqueta rotada dentro del polígono. La rotación respeta la orientación del polígono y la ubicación preferida es más alejada de los bordes del polígono. Es posible *Permitir colocar etiquetas fuera de polígonos*.
- *Usando Perímetro*: dibuja la etiqueta paralela a una línea generalizada que representa el límite del polígono, con preferencia por las porciones más rectas del perímetro. Puede definir:
  - *Posiciones permitidas*: *Encima de la línea*, *En la línea*, :guilabel:` Debajo de la línea` y *Posición dependiente de la orientación de la línea* (colocando la etiqueta a la izquierda o al a la derecha del límite del polígono). Es posible seleccionar varias opciones a la vez. En ese caso, QGIS buscará la posición óptima de la etiqueta.
  - *Distancia* entre etiqueta y el borde del polígono
  - la *Etiquetas repetidas Distancia* para mostrar varias veces la etiqueta a lo largo del perímetro.
- *Usando Perímetro (Curvado)*: dibuja la etiqueta siguiendo la curvatura del límite del polígono. Además de los parámetros disponibles con *Usando el modo Perímetro*, puede configurar un *Ángulo máximo entre polígono de caracteres curvos*, ya sea dentro o fuera.
- *Polígonos exteriores*: siempre coloca etiquetas fuera de los polígonos, a una *Distancia* establecida

Configuración de ubicación común

Algunas configuraciones de ubicación de etiquetas están disponibles para todos los tipos de geometría de capa:

Label Spacing

The *Label Spacing* option allows you to set the margin around the label, preventing all other labels from being placed within that area. You can dynamically control *Margin around label* using the Data-defined override button. Choose unit from the unit selector, and there you can also *adjust scaling range*.

Atención

This option is only available on QGIS installed with GEOS >= 3.10 (see *Help* ► *About* menu).

Duplicate Labels

Use *Avoid duplicate labels* to improve label appearance in situations like road networks, where features such as dual carriageways or service roads can result in many identical labels appearing close together. When active, it prevents placing labels with the exact same text (case-sensitive) closer than the minimum distance defined in the *Settings…* option. The setting applies across layers, so duplicate text from any layer will be considered.

Atención

This option is only available on QGIS installed with GEOS >= 3.10 (see *Help* ► *About* menu).

Generador de geometría

La sección *Geometry Generator* permite al usuario modificar la geometría subyacente utilizada para colocar y representar la etiqueta, mediante el uso de *expressions*. Esto puede ser útil para realizar el desplazamiento de la geometría de forma dinámica o para convertirla a otra geometría (tipo).

Para utilizar el generador de geometría:

1. Marque la opción *Geometry generator*
2. Introduzca la expresión que genera la geometría en la que confiar
3. Si es relevante, seleccione el tipo de geometría de la salida de la expresión: la configuración basada en la geometría de la etiqueta, como la ubicación o la representación, se actualiza para coincidir con las nuevas capacidades del tipo de geometría.

Algunos casos de uso incluyen:

- Usar una geometría que está guardada en otro campo «label_position»
- Use la *generated geometry* de la simbología también para el etiquetado
- Utilice la variable @map_scale para calcular distancias/tamaños independientemente del nivel de zoom.
- Combinado con el modo de colocación curvo, crea una etiqueta circular alrededor de una entidad de punto:

```
exterior_ring(make_circle($geometry, 20))
```

- Agregue una etiqueta al principio y al final de una característica de línea:

```
collect_geometries( start_point($geometry), end_point($geometry) )
```

- Confíe en una línea suavizada de un río para obtener más espacio para la colocación de etiquetas:

```
smooth( $geometry, iterations:=30, offset:=0.25, min_length:=10 )
```

Definida por Datos

El grupo *Definido por Datos* proporciona control directo sobre la ubicación de las etiquetas, entidad por entidad. Se basa en sus atributos o en una expresión para establecer:

- la coordenada *X* e *Y*
- la alineación del texto sobre la posición personalizada establecida arriba:
  - *Horizontal*: pude ser a la **Izquierda**, **Centro** o **Derecha**
  - el texto *Vertical*: puede ser **Abajo**, **Base**, **Mitad**, **Cima** o **Arriba**
- el texto *Rotación*. La rotación se define como un ángulo en el sentido de las agujas del reloj con 0° apuntando en dirección Este para el texto con orientación `Horizontal` y con 0° apuntando en dirección Norte para el texto con orientación `Vertical`. Se pueden definir diferentes unidades para la rotación del etiquetado (por ejemplo, `grados`, `minutos de arco`, `vueltas`). QGIS prioriza los ángulos de rotación de la pantalla por defecto y siempre rota el texto para que sea más legible en la pantalla, por lo que los ángulos de rotación «al revés» se anulan durante el renderizado. Para cambiar este comportamiento y forzar a QGIS a renderizar los ángulos de rotación exactos definidos por los datos, existe una opción en la pestaña *Renderizado*, en *Mostrar etiquetas al revés*. Marque la opción *Preservar valores de rotación de datos* si desea mantener el valor de rotación en el campo asociado y aplicarlo a la etiqueta, tanto si está fijada como si no. Si no está marcada, al desanclar la rotación de la etiqueta se restablece y su valor se borra de la tabla de atributos.

> **Nota**
>
> La rotación definida por datos con entidades poligonales actualmente solo se admite con el modo de ubicación *En torno a Centroide*.

> **Nota**
>
> Las expresiones no se pueden usar en combinación con las herramientas de mapa de etiquetas (es decir, las herramientas *Rotar etiqueta* y *Mover etiqueta*) a emplzaamiento de etiquetas *definida por datos*. El control se restablecerá al correspondiente *campo de almacenamiento auxiliar*.

Prioridad

En la sección *Prioridad* puede definir el rango de prioridad de ubicación de cada etiqueta, es decir, si hay diferentes diagramas o etiquetas candidatas para la misma ubicación, se mostrará el elemento con mayor prioridad y los demás podrían quedar fuera .

El rango de prioridad también se usa para evaluar si una etiqueta podría omitirse debido a una mayor ponderación de *entidad de obstáculo*.

Obstáculos

En algunos contextos (por ejemplo, etiquetas de alta densidad, entidades superpuestas…), la ubicación de las etiquetas puede provocar que las etiquetas se coloquen sobre entidades no relacionadas.

Un obstáculo es una entidad sobre la cual QGIS evita colocar etiquetas o diagramas de otras entidades . Esto se puede controlar desde la sección *Obstáculos* :

1. Active la opción *Los objetos actúan como obstáculos* para decidir que los objetos de la capa deben actuar como obstáculos para cualquier etiqueta y diagrama (incluidos elementos de otras entidades en la misma capa).

   En lugar de la capa completa, puede seleccionar un subconjunto de entidades para usar como obstáculos, usando el control anulación definida por datos junto a la opción.

2. Use el botón *Configuración* para ajustar la ponderación del obstáculo.

   - Para cada entidad de obstáculo potencial, puede asignar un *Peso del obstáculo*: cualquier *etiqueta* o *diagrama* cuyo rango de prioridad de ubicación sea mayor que este valor, se puede ubicar. Las etiquetas o diagramas con rango inferior se omitirán si no es posible otra ubicación.

     Esta ponderación también se puede definir por datos, de modo que dentro de la misma capa, es más probable que se cubran determinados objetos que otras.

   - Para las capas de polígono, puede elegir el tipo de obstáculo en el que se encuentra la entidad:

     – **sobre el interior de la entidad**: evita colocar etiquetas sobre el interior del polígono (prefiere colocar etiquetas totalmente fuera o solo ligeramente dentro del polígono)

     – o **sobre el límite de la entidad**: evita colocar etiquetas sobre el límite del polígono (prefiere colocar etiquetas fuera o completamente dentro del polígono). Esto puede ser útil para capas donde las características cubren toda el área (unidades administrativas, coberturas categóricas, …). En este caso, es imposible evitar colocar etiquetas dentro de estas entidades, y se ve mucho mejor cuando se evita colocarlas sobre los límites entre entidades.

Pestaña Renderizado

En la pestaña *Renderizado*, puede ajustar cuándo se pueden representar las etiquetas y su interacción con otras etiquetas y funciones.

Opciones de Etiqueta

En *opciones de Etiqueta*:

- Encuentras las configuraciones de visibilidad *basada en escala* y la *basada en tamaño de pixel*.
- El *Label z-index* determina el orden en el que se renderizan las etiquetas, así como en relación con otras etiquetas de entidades en la capa (usando una expresión de superposición definida por datos), como con las etiquetas de otras capas. Las etiquetas con un índice Z más alto se representan encima de las etiquetas (de cualquier capa) con un índice Z más bajo.

  Adicionálmente, sla lógica se ha ajustado para que si dos etiquetas tienen índices Z coincidentes, entonces:

  - si son de la misma capa, la etiqueta más pequeña se dibujará sobre la etiqueta más grande
  - si son de diferentes capas, las etiquetas se dibujarán en el mismo orden que sus propias capas (es decir, respetando el orden establecido en la leyenda del mapa).

  Nota

  Esta configuración no hace que las etiquetas se dibujen debajo de las entidades de otras capas, solo controla el orden en el que se dibujan las etiquetas encima de todas las entidades de las capas.

- *Permitir colocaciones inferiores*: Por defecto QGIS intenta mostrar las etiquetas en su mejor ubicación, siguiendo su configuración. Marque este modo para permitir que las características se coloquen en peores opciones cuando no hay otra opción, por ejemplo, cuando una línea es demasiado corta para adaptarse a un texto de etiqueta curva, la etiqueta puede colocarse horizontalmente justo sobre el punto central de la característica.
- Con expresiones definidas por datos en *Mostrar etiqueta* y :guilabel:` Mostrar siempre` puede ajustar qué etiquetas se deben representar.
- Permitir *Mostrar etiquetas al revés*: las alternativas son **nunca**, **cuando se defina la rotación** o **siempre**.
  - **nunca** - configuración predeterminada, se da prioridad a la legibilidad de la pantalla,
  - **cuando se define la rotación** - la rotación de la etiqueta debe definirse en la pestaña *Placement*, dentro del grupo :guilabel:`Data Defined
  - **siempre** - se permiten las etiquetas al revés

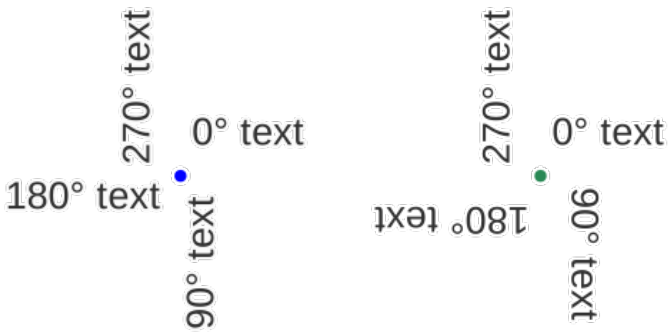


Figura 10.35: Rotación de etiquetas definida por datos con la opción de mostrar etiquetas al revés establecida en: «nunca» (izquierda), “cuando se define la rotación” (derecha)

- El grupo *Etiquetas superpuestas* permite controlar si se permiten etiquetas superpuestas para los objetos espaciales de la capa y cómo debe tratarse cada una de ellas:
  - *Nunca solapar*: nunca colocar etiquetas solapadas para la capa, aunque eso signifique que faltarán algunas etiquetas
  - *Permitir solapamientos en caso necesario*: si la etiqueta no puede colocarse de otro modo, dibuja una etiqueta superpuesta. Este modo hará que la etiqueta se mueva a una colocación menos ideal si es posible, por ejemplo, mover la etiqueta más lejos del centro de una línea o polígono, SI al hacerlo se evita que las etiquetas se solapen. Pero si no hay otras posiciones posibles, entonces dibuje la etiqueta solapada.
  - *Permitir solapamientos sin penalización*: No importa en absoluto si la etiqueta se solapa con otras etiquetas u obstáculos, eso está bien hacerlo y siempre se debe utilizar la mejor colocación (por ejemplo, la colocación más central), incluso si es posible una colocación alternativa posterior que evite por completo los solapamientos.

  Si se permite la superposición de etiquetas y las opciones de colocación alternativa, se garantizará que todas las características de la capa estén etiquetadas… ¡y no necesariamente en su mejor representación!

Opciones de Entidad

En *opciones de Entidad*:

- Puede elegir *Etiquetar cada parte de una entidad de varias partes* y *Limitar el número de entidades a etiquetar*.
- Tanto la capa de línea como la de polígono ofrecen la opción de establecer un tamaño mínimo para las entidades a etiquetar, usando *Suprimir el etiquetado de entidades menores que*.
- En el caso de las entidades poligonales, también puede filtrar las etiquetas para que se muestren según se ajusten completamente a su entidad o no.
- Para entidades de línea, puede elegir *Fusionar líneas conectadas para evitar etiquetas duplicadas*, lo que representa un mapa bastante aireado junto con las opciones *Distancia* o *Repetir* en la pestaña *Ubicación*.

10.4 Creando Símbolos 3D

El *Administrador de estilo* le ayuda a crear y almacenar símbolos 3D para cada tipo de geometría a renderizar en la vista de mapa *3D*.

En cuanto a los demás elementos, habilitar la pestaña  *Símbolos 3D* y expanda el menú del botón  para crear:

- *símbolos de punto 3D*
- *símbolos lineales 3D*
- *símbolos poligonales 3D*

10.4.1 Capas de puntos

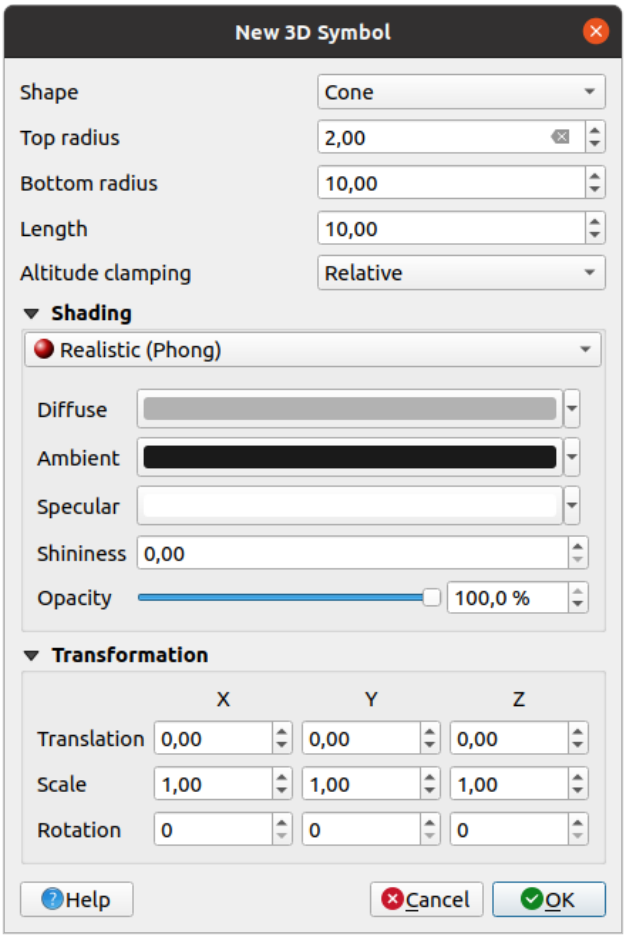


Figura 10.36: Propiedades de un símbolo de punto 3D

- Puede definir diferentes tipos de *Forma* 3D para usar con símbolos de puntos. Se definen principalmente por sus dimensiones cuya unidad se refiere al SRC del proyecto. Los tipos disponibles son:
  - *Esfera* definida por un *Radio*
  - *Cilindro* definido por un *Radio* y *Longitud*
  - *Cubo* definido por un *Tamaño*
  - *Cono* definido por un *Radio superior*, un *Radio inferior* y un *Longitud*
  - *Plano* definido por un *Tamaño*
  - *Toroide* definido por un *Radio* y un *Radio menor*
  - *Modelo 3D*, utilizando un archivo de modelo 3D: los formatos soportados incluyen wavefront `.obj`, `.glTF` y `.fbx`. Los modelos pueden ser un archivo en disco, una URL remota o *incorporado en el proyecto.*. Los modelos creados por la comunidad se comparten en el QGIS Hub.
  - *Valla publicitaria*, definida por la *Altura de la valla publicitaria* y el *Símbolo de la valla publicitaria* (generalmente basado en un:ref:*marker symbol* <*vector_marker_symbols*>). El símbolo tendrá un tamaño estable. Conveniente para visualizar formas de nubes de puntos 3D.
- El *Sujeción de altitud* se puede configurar en *Absoluto*, *Relativo* o *Terreno*. La configuración *Absoluto* se puede usar cuando los valores de altura de los vectores 3d se proporcionan como medidas absolutas desde 0. *Relativo* y *Terreno* agregan valores de elevación dados a la elevación del terreno subyacente.

- Las propiedades *sombreado* se pueden definir.
- Bajo el cuadro *Transformaciones*, puede aplicar una transformación afín al símbolo:
  - *Translación* para mover objetos en los ejes x, y y z.
  - *Escala* para cambiar el tamaño de las formas 3D
  - *Rotación* alrededor del eje x, y y z.

10.4.2 Capas lineales

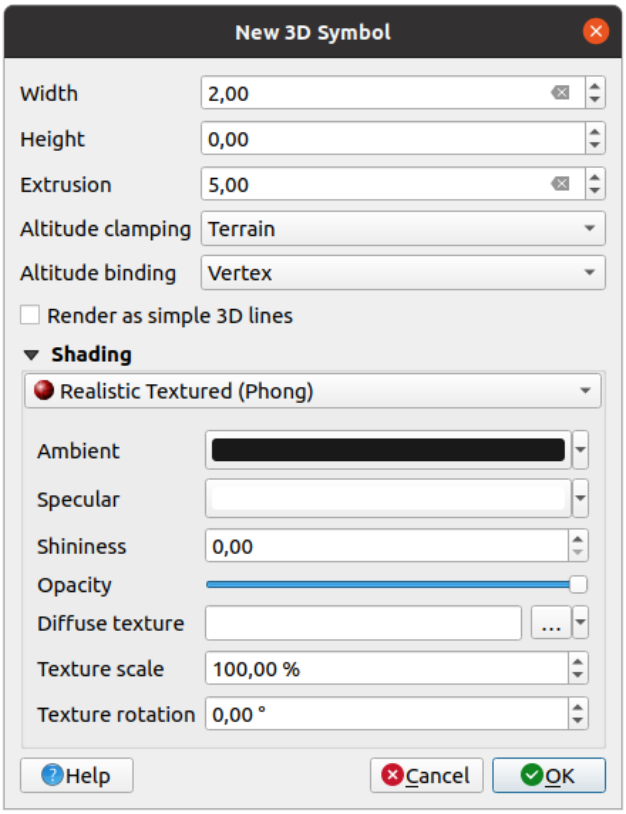


Figura 10.37: Propiedades de símbolo lineal 3D

- Debajo de las configuraciones *Anchura* y *Altura* puede definir *Extrusión* de las líneas vectoriales. Si las líneas no tienen valores z, puede definir los volúmenes 3d con esta configuración.
- Con *Fijación de altitud*, define la posición de las líneas 3D en relación con la superficie del terreno subyacente, si ha incluido datos de elevación ráster u otros vectores 3D.
- La *Altitud de enlazado* define cómo se fija el objeto al terreno. O cada *Vértice* de la entidad se sujetará al terreno o esto se hará por *Centroide*.
- Es posible *Renderizar como líneas simpes 3D*.
- Las propiedades *sombreado* se pueden definir.

10.4.3 Capas Poligonales

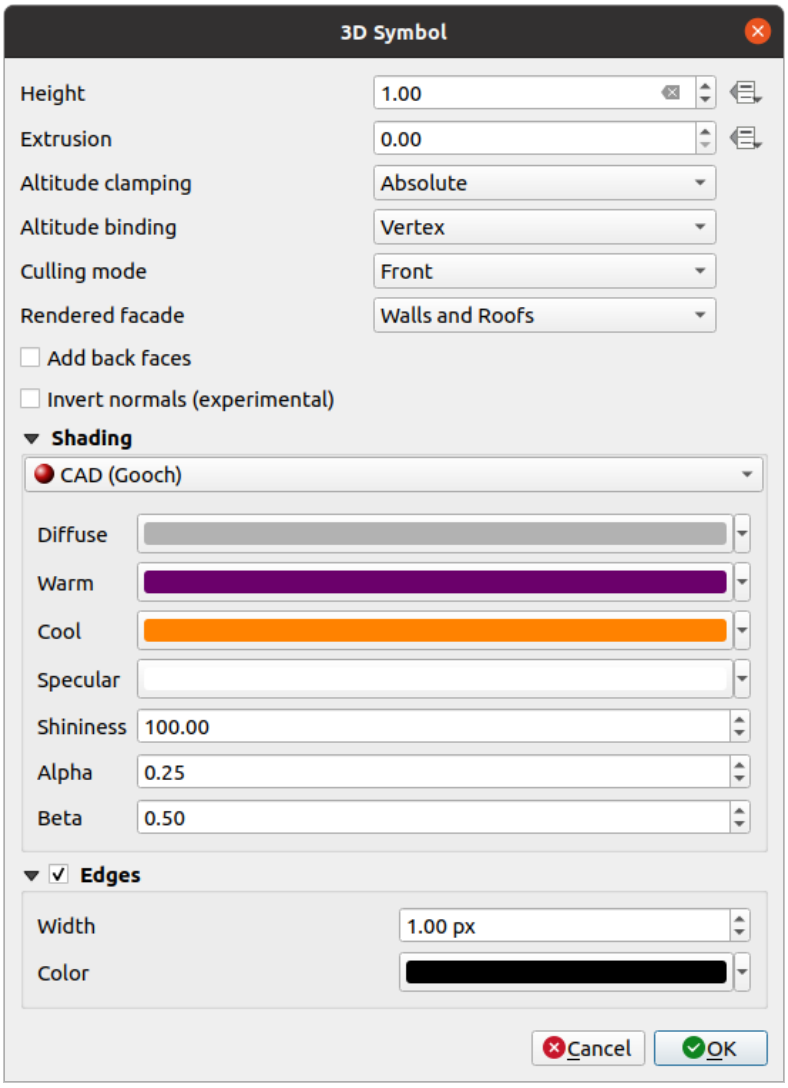


Figura 10.38: Propiedades de un símbolo poligonal 3D

- Como los demás, la *Altura* se puede definir en unidades SRC. También puede utilizar el botón para sobrescribir el valor con una expresión personalizada, una variable o una entrada de la tabla de atributos
- Nuevamente, la *Extrusion* es posible para valores z faltantes. También para la extrusión puede utilizar para usar los valores de la capa vectorial y tener diferentes resultados para cada polígono:

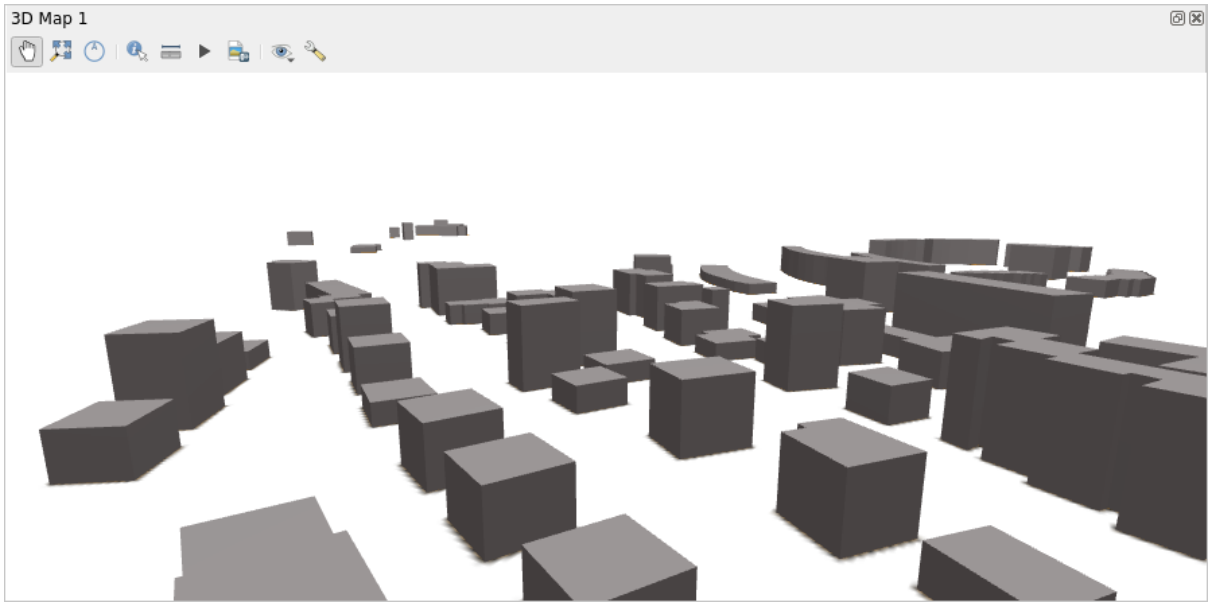


Figura 10.39: Extrusión Definida por Datos

- La *Altitud de anclaje*, *Altitud de enlazado* puede ser definida como se explica arriba.
- El :guilabel:``Modo de selección` a aplicar al símbolo; puede ser:
  - *No Culling*: esto puede ayudar a evitar superficies aparentemente faltantes cuando los datos de polígonoZ/multiparche no tienen un orden consistente de vértices (por ejemplo, todo en el sentido de las agujas del reloj o en sentido contrario)
  - *parte delantera*
  - or *parte trasera*
- La *Fachada renderizada* determina las caras a mostrar. Los valores posibles son *Sin fachadas*, *Muros*, *Cubiertas*, o *Muros y cubiertas*
- :guilabel:`Agregar caras posteriores" para cada triángulo, crea caras frontal y posterior con las normales correctas, a expensas de un mayor número de datos de vértice. Esta opción se puede usar para corregir problemas de sombreado (por ejemplo, debido a datos con un orden de vértices inconsistente).
- *Invertir normales (experimental)*: puede ser útil para fijar órdenes de vértice de cara en sentido horario/antihorario
- Las propiedades *sombreado* se pueden definir.
- visualización de los símbolos *Bordes* se puede habilitar y asignar un *Ancho* y *Color*.

Consejo

Combinación para la mejor representación de datos 3D

Culling mode, *Agregar cara posterior* y *Invert normales* tienen como objetivo corregir el aspecto de los datos 3D si no se ve bien. Por lo general, cuando se cargan algunos datos, lo mejor es probar primero `culling mode=back` y `add back faces=disabled`; es lo más eficiente. Si el renderizado no se ve correcto, intente `agregar caras traseras=habilitado` y mantenga el `modo de selección=sin selección`. Otras combinaciones son más avanzadas y útiles solo en algunos escenarios según cuán mezclado esté el conjunto de datos de entrada.

10.4.4 Sombreando la textura

El sombreado te ayuda a revelar detalles en 3D de objetos que, de lo contrario, podrían estar ocultos debido a la iluminación de la escena. En última instancia, es un material más fácil para trabajar, ya que no necesita preocuparse por configurar la iluminación de escena adecuada para visualizar las características.

En QGIS se utilizan varias técnicas de sombreado y su disponibilidad depende del tipo de geometría del símbolo:

- *Realista (Phong)*: describe la forma en que una superficie refleja la luz como una combinación de la reflexión *Difusa* de las superficies rugosas con la reflexión *Especular* de las superficies brillantes (*Brillo*). También incluye una opción *Anbiente* para tener en cuenta la pequeña cantidad de luz que se dispersa por toda la escena. Utilice el deslizador *Opacidad* para renderizar objetos semitransparentes en 3D. Más información en Descripción de Phong reflection.
- *Textura realista (Phong)*: igual que *Realista (Phong)* excepto en que se utiliza una imagen como *Textura difusa*. La imagen puede ser un archivo en disco, una URL remota o *incorporada en el proyecto.*. Se requieren los parámetros *Escala de la textura* y *Giro de la textura*. Utilice el deslizador *Opacidad* para renderizar objetos semitransparentes en 3D.
- *CAD (Gooch)*: esta técnica permite que el sombreado se produzca sólo en los tonos medios, de modo que las líneas de los bordes y los resaltes sigan siendo visualmente prominentes. Junto con las opciones *Difuso*, *Especular*, *Sombreado*, debe proporcionar un color *Cálido* (para las superficies orientadas hacia la luz) y un color *Frío* (para las orientadas hacia el exterior). Además, las contribuciones relativas a los colores frío y cálido del color difuso se controlan mediante las propiedades *Alpha* y *Beta* respectivamente. Véase también Gooch shading.
- *Rugosidad del metal*: un material de renderizado basado en la física que proporciona una representación precisa de cómo interactúa la luz con las superficies. Existen opciones para configurar el material *Color base*, *Rugosidad metálica* y *Rugosidad*.
- *Texturas Embebidas* con forma de modelos 3D

10.4.5 Ejemplo de Aplicación

Para ir a través de los ajustes explicados arriba puede echar un vistazo a https://app.merginmaps.com/projects/saber/luxembourg/tree.

CAPÍTULO 11

Administrar el origen de datos

11.1 Abriendo Datos

Como parte de un ecosistema de software de código abierto, QGIS se basa en diferentes bibliotecas que, combinadas con sus propios proveedores, ofrecen capacidades para leer y, a menudo, escribir muchos formatos:

- Los formatos de datos vectoriales incluyen GeoPackage, GML, GeoJSON, GPX, KML, Comma Separated Values, formatos ESRI (Shapefile, Geodatabase…), formatos de archivo MapInfo y MicroStation, AutoCAD DWG/DXF, GRASS y muchos más… Lea la lista completa de formatos vectoriales admitidos.
- Los formatos de datos ráster incluyen GeoTIFF, JPEG, ASCII Gridded XYZ, MBTiles, R o Idrisi rasters, GDAL Virtual, SRTM, Sentinel Data, ERDAS IMAGINE, ArcInfo Binary Grid, ArcInfo ASCII Grid, y muchos más… Lea la lista completa de formatos ráster soportados.
- Database formats include PostgreSQL, SQLite/SpatiaLite, Oracle, MS SQL Server, SAP HANA, MySQL…
- Los servicios de datos y mapas web (WM (T) S, WFS, WCS, CSW, mosaicos XYZ, servicios ArcGIS, …) también son manejados por proveedores de QGIS. Ver *Trabajando con protocolos OGC / ISO* para obtener más información sobre algunos de estos.
- Puede leer archivos compatibles de carpetas archivadas y utilizar formatos nativos de QGIS como archivos QML (*QML: el formato de archivo de estilo QGIS*) y capas virtuales y de memoria.

Más de 80 formatos vectoriales y 140 ráster son compatibles con GDAL y los proveedores nativos de QGIS.

Nota

No todos los formatos enumerados pueden funcionar en QGIS por varias razones. Por ejemplo, algunos requieren bibliotecas propietarias externas, o es posible que la instalación GDAL/OGR de su sistema operativo no se haya creado para admitir el formato que desea utilizar. Para ver la lista de formatos disponibles, ejecute la línea de comando `ogrinfo --formats` (para vectorial) y `gdalinfo --formats` (para ráster), o marque la *Configuración -> Opciones -> Menú GDAL* en QGIS.

En QGIS, dependiendo del formato de los datos, existen diferentes herramientas para abrir un conjunto de datos, principalmente disponibles en el menú *Capa* ► *Añadir Capa* ► o desde la barra de herramientas *Gestionar Capas* (activada a través del menú *Ver* ► *Barras de Herramientas*). Sin embargo, todas estas herramientas apuntan a un

único diálogo, el *Gestor de fuentes de datos*, que puede abrir con el botón Gestor de fuentes de datos abiertos, disponible en la barra de herramientas *Gestor de fuentes de datos*, o pulsando `Ctrl+L`. El cuadro de diálogo *Gestor de fuentes de datos* (Figura 11.1) ofrece una interfaz unificada para abrir datos basados en archivos, así como bases de datos o servicios web soportados por QGIS.

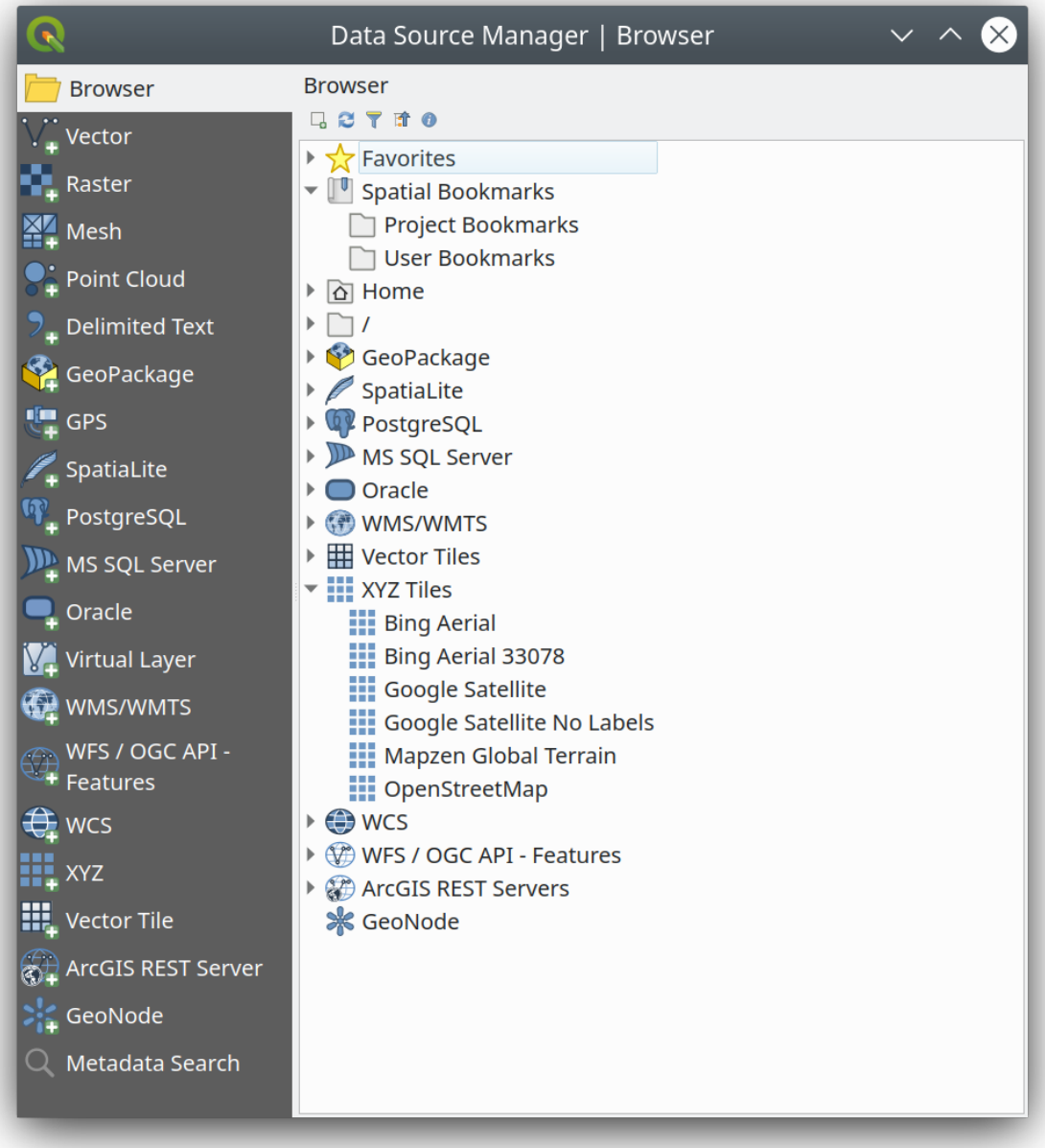


Figura 11.1: Cuadro de diálogo Administrador de fuente de datos de QGIS

Además de este punto de entrada principal, también tiene el complemento *Administrador de BBDD* que ofrece capacidades avanzadas para analizar y manipular bases de datos conectadas. Puede encontrar más información sobre las capacidades de Administrador de BBDD en *Complemento de Administración de BBDD*.

Hay muchas otras herramientas, complementos nativos o de terceros, que lo ayudan a abrir varios formatos de datos.

Este capítulo describirá solo las herramientas proporcionadas por defecto en QGIS para cargar datos. Se centrará principalmente en el cuadro de diálogo *Administrador de fuentes de datos* pero más que describir cada pestaña, también explorará las herramientas basadas en el proveedor de datos o las especificidades del formato.

11.1.1 El panel Navegador

El *Navegador* es una de las principales formas de agregar datos a los proyectos de forma rápida y sencilla. Está disponible como:

- una pestaña *Administrador de fuentes de datos*, habilitada presionando el botón  Abrir Administrador de fuentes de datos (`Ctrl+L`);
- como panel QGIS puede abrir desde el menú:menuselection:*Ver* -> *Paneles* (o  :menuselection:` Configuración -> Paneles`) o presionando `Ctrl+2`.

En ambos casos, el *Navegador* le ayuda a navegar en su sistema de archivos y administrar geodatos, independientemente del tipo de capa (raster, vector, tabla) o el formato de la fuente de datos (archivos sin formato o comprimidos, bases de datos, servicios web) .

Explorando la interfaz

En la parte superior del panel navegador, encontrará algunos botones que pueden ayudarle a:

-  Añadir capas seleccionadas: también puede agregar datos al lienzo del mapa seleccionando **Agregar capa(s) seleccionadas** en el menú contextual de la capa;
-  Actualizar el árbol de navegador;
-  Explorador de Filtros para buscar datos específicos. Ingrese una palabra de búsqueda o un comodín y el navegador filtrará el árbol para mostrar solo las rutas a las tablas, nombres de archivo o carpetas de la base de datos que coincidan; no se mostrarán otros datos o carpetas. Vea el ejemplo del Panel del navegador (2) en Figura 11.2. La comparación puede distinguir entre mayúsculas y minúsculas o no. También se puede configurar para:
  - *Normal*: mostrar elementos que contienen el texto buscado
  - *Wildcard(s)*: afine la búsqueda usando los caracteres `?` y/o `*` para especificar la posición del texto de búsqueda
  - *Expresión regular*
-  Colapsar Todo el árbol entero;
-  Habilitar/deshabilitar el widget de propiedades: cuando se activa, se agrega un nuevo widget en la parte inferior del panel que muestra, si corresponde, los metadatos del elemento seleccionado.

Las entradas en el panel *Navegador* están organizados jerárquicamente, y hay varias entradas de nivel superior:

1. *Favoritos* donde puede colocar accesos directos a ubicaciones de uso frecuente
2. *Marcadores Espaciales* donde puede almacenar extensiones de mapas de uso frecuente (consulte *Marcar extensiones en el mapa*)
3. *Project Home*: para un acceso rápido a la carpeta en la que se almacenan (la mayoría de) los datos relacionados con su proyecto. El valor predeterminado es el directorio donde reside el archivo de su proyecto.
4. *Home* directorio en el sistema de archivos y el directorio raíz del sistema de archivos.
5. Unidades de red o locales conectadas
6. Luego viene una serie de tipos de contenedores / bases de datos y protocolos de servicio, según su plataforma y bibliotecas subyacentes:
   -  *GeoPackage*
   -  *SpatiaLite*
   -  *PostgreSQL*

-  *HANA SAP*
-  *MS SQL Server*
-  *Oracle*
-  *WMS/WMTS*
-  *Teselas Vectoriales*
-  *Teselas XYZ*
-  *WCS*
-  *WFS/OGC API-Objetos*
-  *ArcGIS REST Server*

Interactuar con los elementos del navegador

El navegador admite arrastrar y soltar dentro del navegador, desde el navegador al lienzo y el panel *Capas*, y desde el panel *Capas* a contenedores de capas (por ejemplo, GeoPackage) en el navegador.

Los elementos del archivo de proyecto dentro del navegador se pueden expandir, mostrando el árbol de capas completo (incluidos los grupos) contenido dentro de ese proyecto. Los elementos del proyecto se tratan de la misma manera que cualquier otro elemento en el navegador, por lo que se pueden arrastrar y soltar dentro del navegador (por ejemplo, para copiar un elemento de capa a un archivo de geopaquete) o agregar al proyecto actual a través de arrastrar y soltar o doble hacer clic.

El menú contextual de un elemento en el panel *Navegador* se abre haciendo clic derecho sobre él.

Para las entradas del directorio del sistema de archivos, el menú contextual ofrece lo siguiente:

- *Nuevo* ► para crear en la entrada seleccionada una:
  - *Directorio…*
  - *GeoPackage…*
  - *ArchivoShape…*
- *Agregar como un favorito*: las carpetas favoritas pueden ser renombradas (*Renombrar favoritos…*) o borradas (*Borrar Favorito*) en cualquier momento.
- *Ocultar del Navegador*: las carpetas ocultas pueden ser conmutadas a visibles en la configuración *Configuración* ► *Opciones* ► *Fuenntes de Datos* ► *Ocultar rutas de explorador*
- *Escanear rápido este directorio*
- *Abrir directorio*
- *Abrir en Terminal*
- *Propiedades…*
- *Propiedades de directorio…*

Para las entradas de hoja que pueden actuar como capas en el proyecto, el menú contextual tendrá entradas de apoyo. Por ejemplo, para fuentes de datos vectoriales, ráster y de malla que no son de base de datos, no basadas en servicios:

- *Exportar capa* ► *A archivo…*
- *Añadir capa a proyecto*
- *Propiedades de capa*
- *Abrir con Administrador de Fuentes de Datos…*

- *Gestión ► Renombrar «<name of file>»…* o *Borrar «<name of file>»…*
- *Mostrar en archivos*
- *Propiedades de archivo*

En la entrada *propiedades de capa*, encontrará (similar a lo que encontrará en las propiedades de capa *vectorial* y *ráster* una vez agregadas las capas al proyecto):

- *Metadatos* para la capa. Grupos de metadatos: :guilabel:'Información del proveedor` (si es posible, *Ruta* será un hipervínculo a la fuente), *Identificación*, *Extensión*, *Acceso* , *Campos* (para capas vectoriales), *Bandas* (para capas ráster), *Contactos*, *Enlaces* (para capas vectoriales), *Referencias* (para capas ráster), *Historial*.
- Un panel *Vista previa*
- La tabla de atributos para fuentes vectoriales (en el panel *Atributos*).

Utilice *Abrir con el Gestor de Fuentes de Datos…* para abrir y configurar directamente la fuente de datos en el *Administador de Fuentes de Datos* utilizando el URI de su fuente de datos. Esto simplifica el proceso de añadir una capa desde el *Navegador* permitiéndole establecer opciones de apertura específicas para la fuente de datos. Actualmente está disponible para fuentes de datos vectoriales (incluyendo la entrada dedicada GeoPackage), ráster y SpatiaLite.

Para agregar una capa al proyecto usando *Navegador*:

1. Habilite *Navegador* como se describe arriba. Se muestra un árbol del navegador con su sistema de archivos, bases de datos y servicios web. Es posible que deba conectar bases de datos y servicios web antes de que aparezcan (consulte las secciones dedicadas).
2. Encuentre la capa en la lista.
3. Utilice el menú contextual, haga doble clic en su nombre o arrástrelo y suéltelo en el *lienzo del mapa*. Su capa será añadida ahora al *panel de Capas* y se puede ver en el lienzo del mapa.

> **Truco**
>
> **Abra un proyecto QGIS directamente desde el navegador**
>
> También puede abrir un proyecto QGIS directamente desde el panel del navegador haciendo doble clic en su nombre o arrastrando y soltando en el lienzo del mapa.

Una vez que se carga un archivo, puede acercarlo usando las herramientas de navegación del mapa. Para cambiar el estilo de una capa, abra el cuadro de diálogo *Propiedades de la capa* haciendo doble clic en el nombre de la capa o haciendo clic derecho en el nombre en la leyenda y eligiendo *Propiedades* en el menú contextual. Consulte la sección *Propiedades de simbología* para obtener más información sobre cómo configurar la simbología para las capas vectoriales.

Hacer clic con el botón derecho en un elemento del árbol del navegador le ayuda a:

- para un archivo o una tabla, muestre sus metadatos o ábralo en su proyecto. Las tablas incluso se pueden renombrar, eliminar o truncar.
- para una carpeta, márquela en sus favoritos u ocúltela del árbol del navegador. Las carpetas ocultas se pueden administrar desde la pestaña *Configuración -> Opciones -> Origen de datos*.
- administre sus *marcadores espaciales*: Los marcadores se pueden crear, exportar e importar como archivos `XML`.
- crear una conexión a una base de datos o un servicio web.
- actualizar, renombrar o eliminar un esquema.

También puede importar archivos a bases de datos o copiar tablas de un esquema / base de datos a otro con solo arrastrar y soltar. Hay un segundo panel de navegador disponible para evitar desplazamientos largos mientras se arrastra. Simplemente seleccione el archivo y arrastre y suelte de un panel al otro.

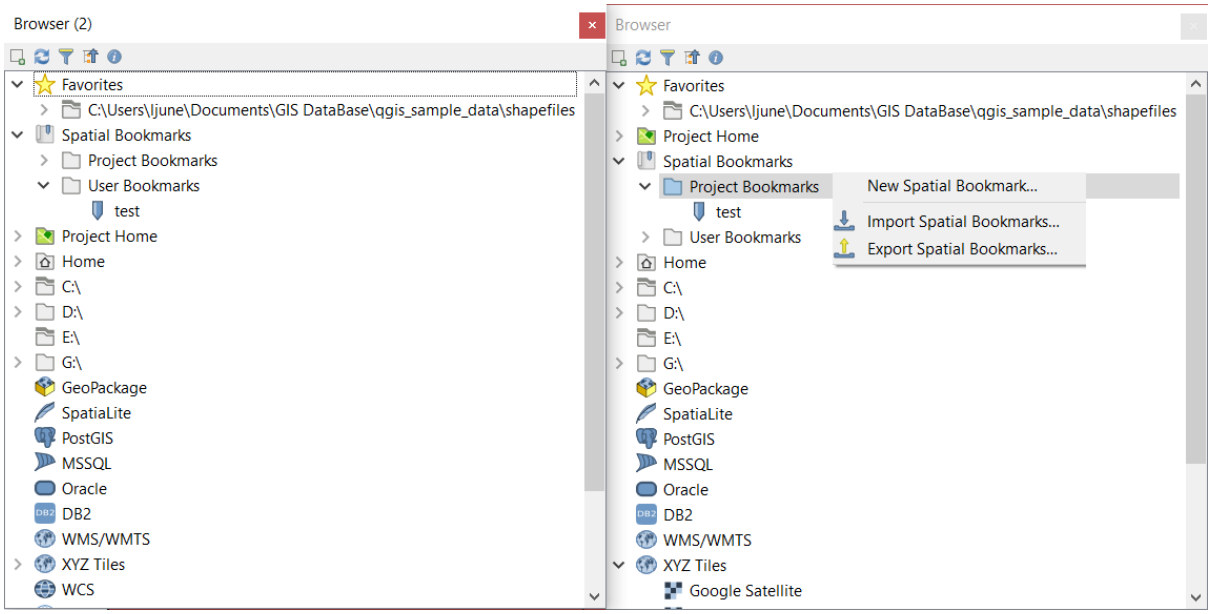


Figura 11.2: Paneles del navegador QGIS uno al lado del otro

Truco

Agregar capas a QGIS simplemente arrastrando y soltando desde el navegador de archivos de su sistema operativo

También puede agregar archivos al proyecto arrastrándolos y soltándolos desde el explorador de archivos de su sistema operativo en:guilabel:*Layers Panel* o el lienzo del mapa.

11.1.2 El gestor de Base de Datos

The *DB Manager* Plugin is another tool for integrating and managing spatial database formats supported by QGIS (PostgreSQL, SpatiaLite, GeoPackage, Oracle Spatial, MS SQL Server, Virtual layers). It can be activated from the *Plugins* ► *Manage and Install Plugins…* menu.

El Complemento Administrador de BBDD proporciona varias características:

- conectarse a bases de datos y mostrar su estructura y contenido
- previsualizar tablas de bases de datos
- agregue capas al lienzo del mapa, ya sea haciendo doble clic o arrastrando y soltando.
- agregar capas a una base de datos desde el navegador QGIS o desde otra base de datos
- crear consultas SQL y agregar su salida al lienzo del mapa
- crear *capas virtuales*

Encontrará más información sobre las capacidades de DB Manager en *Complemento de Administración de BBDD*.

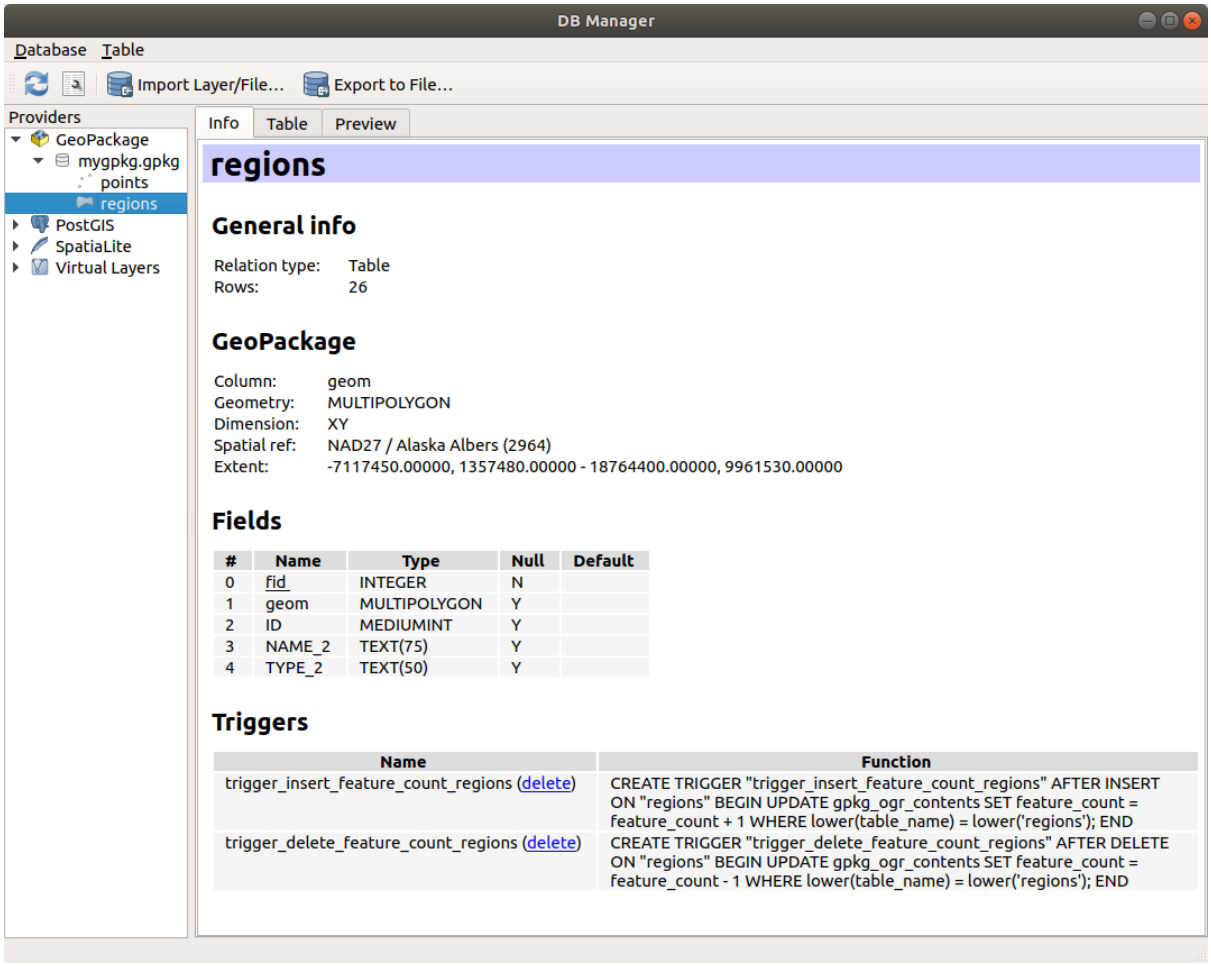


Figura 11.3: Diálogo del complemento administrador de BBDD

11.1.3 Herramientas de carga basadas en proveedores

Además del Panel del navegador y el Administrador de base de datos, las principales herramientas proporcionadas por QGIS para agregar capas, también encontrará herramientas que son específicas para los proveedores de datos.

> **Nota**
>
> Algunos *complementos externos* también proporcionan herramientas para abrir archivos de formato específico en QGIS.

Cargando una capa desde archivo

Para cargar una capa desde un archivo:

1. Abra la pestaña de tipo de capa en el:guilabel:cuadro de diálogo *Administrador de fuentes de datos*, es decir, haga clic en el botón Abrir Administrador de fuente de datos (o presione `Ctrl + L`) y habilita la pestaña de destino o:

   - para datos vectoriales (como capas GML, ESRI Shapefile, Mapinfo y DXF): presione `Ctrl + Shift + V`, seleccione la Opción de menú :menuselection:` Capa -> Agregar capa -> ` *Agregar capa vectorial» o haga clic en el Botón de la barra de herramientas \addOgrLayer| :sup:`Agregar capa vectorial*.

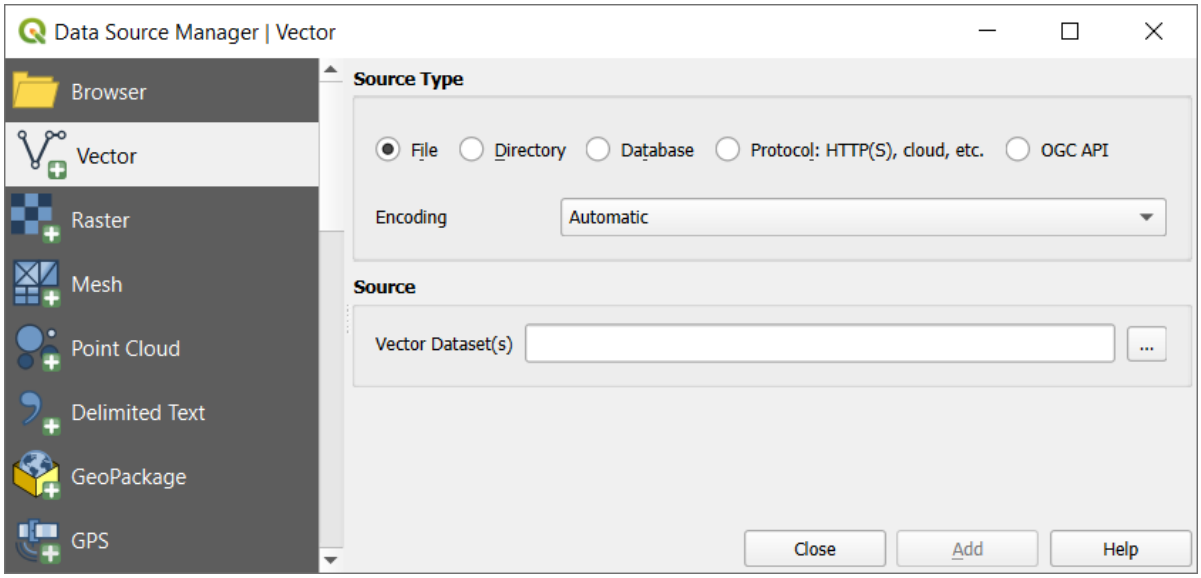


Figura 11.4: Diálogo de Añadir capa vectorial

- para datos ráster (como capas GeoTiff, MBTiles, GRIdded Binary y DWG): presione `Ctrl+Shift+R`, seleccione la opción de menú :menuselection:` Capa –> agregar capa –> ` *Agregar capa ráster* o haga clic en el botón de barra de herramientas Agregar capa ráster.

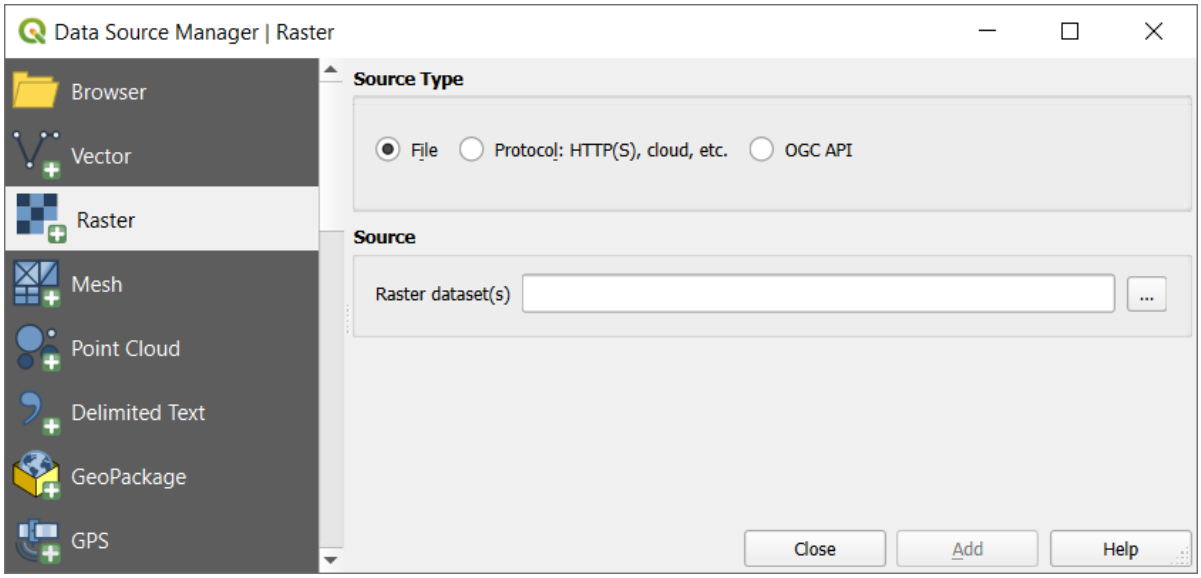


Figura 11.5: Diálogo para Añadir capa ráster

2. Marque tipo de fuente *Archivo*
3. Click en el botón ... Navegador
4. Navegue por el sistema de archivos y cargue una fuente de datos compatible. Se puede cargar más de una capa al mismo tiempo manteniendo presionada la tecla `Ctrl` y haciendo clic en varios elementos en el cuadro de diálogo o manteniendo presionada la tecla `Shift` para seleccionar un rango de elementos haciendo clic en los primeros y últimos elementos de la gama. Solo los formatos que han sido bien probados aparecen en el filtro de formatos. Se pueden cargar otros formatos seleccionando «Todos los archivos» (el elemento superior en el menú desplegable).
5. Pulse *Abrir* para cargar el archivo seleccionado en el cuadro de diálogo *Administrador de fuentes de datos*.

Dependiendo del tipo de capa seleccionada, existen opciones adicionales *Opciones* (codificación, tipo de geometría, filtrado de tablas, bloqueo de archivos, formateo de datos…) que se pueden configurar. Estas opciones se describen en detalle en la documentación específica del controlador GDAL vector o raster. En la parte superior de las opciones, un texto con hipervínculo llevará directamente a la documentación del controlador apropiado para el formato de archivo seleccionado.

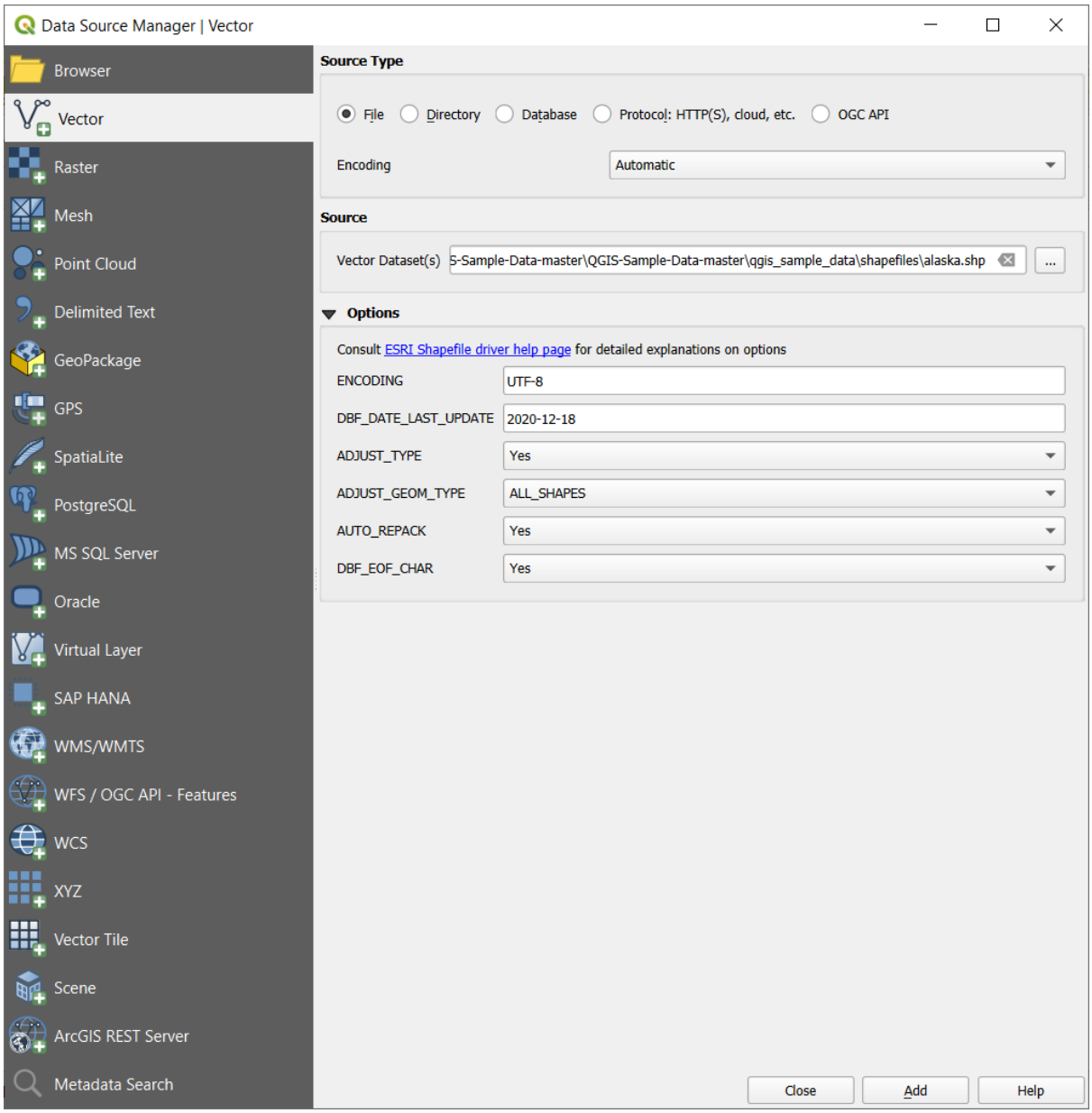


Figura 11.6: Cargar un archivo Shape con opciones de apertura

6. Pulse *Añadir* para cargar el archivo en QGIS y mostrarlo en la vista de mapa. Cuando añada conjuntos de datos vectoriales que contengan varias capas, aparecerá el cuadro de diálogo *Seleccionar elementos para añadir*. En este cuadro de diálogo, puede elegir las capas específicas del conjunto de datos que desea añadir. Además, en *Opciones* puede elegir entre:

   - *Añadir capas a un grupo.*
   - *Mostrar sistema y tablas internas.*
   - *Mostrar capas vectoriales vacías.*

Figura 11.7 muestra QGIS después de cargar el archivo `alaska.shp`.

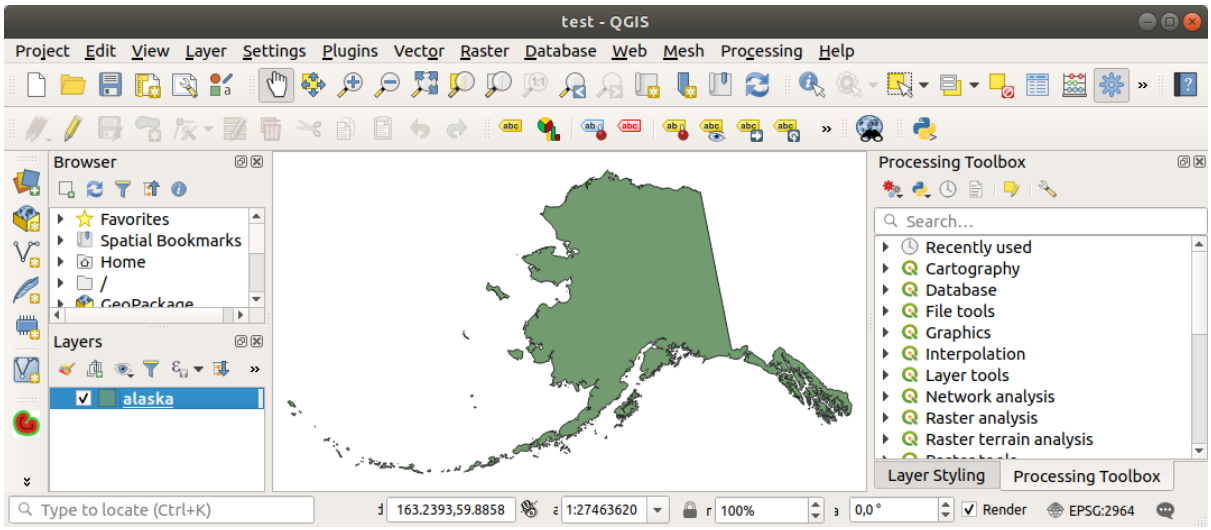


Figura 11.7: QGIS con archivo shape de Alaska cargado

Nota

Debido a que algunos formatos como MapInfo (p. Ej. `.tab`) o Autocad (`.dxf`) permiten mezclar diferentes tipos de geometría en un solo archivo, al cargar dichos conjuntos de datos se abre un diálogo para seleccionar geometrías para usar en para tener una geometría por capa.

Las pestañas Añadir capa vectorial y Añadir capa ráster permitir la carga de capas de tipos de fuente distintos *Archivo*:

- Puede cargar formatos vectoriales específicos como `Cobertura binaria de ArcInfo`, `Reino Unido. Formato de transferencia nacional`, así como el formato TIGER sin procesar de la `Oficina del censo de EE. UU.` o `OpenfileGDB`. Para hacerlo, seleccione *Directorio* como :guilabel:` Tipo de fuente`. En este caso, se puede seleccionar un directorio en el diálogo después de presionar ... Examinar.

- Con el tipo de fuente *Base de datos* puede seleccionar una conexión de base de datos existente o crear una con el tipo de base de datos seleccionado. Algunos tipos de base de datos posibles son `ODBC`, `Esri Personal Geodatabase`, `MS SQL Server` así como `PostgreSQL` o `MySQL`.

  Pressing the *New* button opens the *Create a New OGR Database Connection* dialog whose parameters are among the ones you can find in *Crear una conexión almacenada*. Pressing *Open* lets you select from the available tables.

- El tipo de fuente *Protocolo: HTTP(S), cloud, etc.* abre datos almacenados localmente o en la red, ya sea de acceso público o en depósitos privados de servicios comerciales de almacenamiento en la nube. Los tipos de protocolos admitidos son:

  - `HTTP/HTTPS/FTP`, con *URI* y, si es necesario, una *autenticación*.
  - El almacenamiento en la nube como `AWS S3`, `Google Cloud Storage`, `Microsoft Azure Blob`, `Microsoft Azure Data Lake Storage`, `Alibaba OSS Cloud` y `Open Stack Swift Storage` admite el control directo sobre VSI *Opciones de credenciales* al añadir capas vectoriales OGR o raster GDAL. Primero debe rellenar la *Cubeta o contenedor* y la *Clave de objeto*. A continuación, puede añadir las *Opciones de credenciales* necesarias.

    Al añadir capas vectoriales OGR o raster GDAL desde los protocolos basados en la nube, también puede establecer opciones *Credencial* adicionales para ese controlador y bucket específicos. Cuando las opciones de credenciales se encuentran en el URI de una capa, también se establecerán automáticamente. Esto permite que diferentes capas utilicen diferentes credenciales.

  - servicio compatible con OGC `WFS 3` (todavía experimental), utilizando el formato `GeoJSON` o `GEOJSON - Newline Delimited` o basado en la base de datos `CouchDB`. Un *URI* es obligatorio, con *autenticación* opcional .
  - Para todos los tipos de fuentes vectoriales es posible definir la *Codificación* o usar la configuración *Automática* ►.

- El tipo de fuente *OGC API* le permite acceder a datos vectoriales y raster de servidores que implementan los estándares OGC API. Para utilizar esta opción:

  1. Seleccione *OGC API* en el cuadro de diálogo *Administrador de fuentes de datos*.
  2. Introduzca el punto final del servicio API OGC al que desea conectarse. Ten en cuenta que no es necesario anteponer «OGCAPI:» al punto final.
  3. Haga clic en *Conectar* para establecer una conexión con el servidor.

Cargando una capa de malla

Una malla es una cuadrícula no estructurada generalmente con componentes temporales y de otro tipo. El componente espacial contiene una colección de vértices, aristas y caras en el espacio 2D o 3D. Más información sobre capas de malla en *Trabajando con Malla de Datos*.

Para añadir una malla a QGIS:

1. Abra el diálogo *Administrador de Fuente de datos* , ya sea seleccionándolo desde el menú *Capa* -> o haciendo clic en el botón Abrir Administrador de fuentes de datos.
2. Habilite la pestaña *Malla* en el panel izquierdo
3. Presione el botón … Navegador para seleccionar el archivo. *Varios formatos* son soportados.
4. Selecciona el archivo y presiona *Add*. La capa se agregará utilizando la representación de malla nativa.
5. Si el archivo seleccionado contiene muchas capas de malla, aparecerá un cuadro de diálogo para elegir las subcapas que desea cargar. Haga su selección y presione *OK* y las capas se cargarán con la representación de malla nativa. También es posible cargarlos dentro de un grupo.

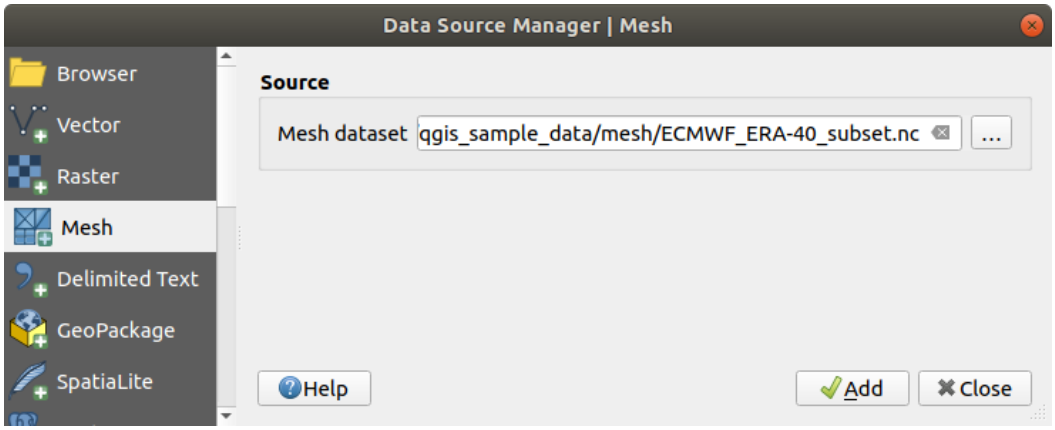


Figura 11.8: Pestaña malla en el Administrador de fuente de datos

Importando un archivo de texto delimitado

Archivos de texto delimitados (p.ej. `.txt`, `.csv`, `.dat`, `.wkt`) se puede cargar utilizando las herramientas descritas anteriormente. De esta forma, aparecerán como tablas simples. A veces, los archivos de texto delimitados pueden contener coordenadas/geometrías que podría querer visualizar. Esto es para lo que está diseñado *Añadir capa de texto delimitado*.

1. Click en el icono Abrir Administrador de Fuentes de Datos para abrir el diálogo *Administrador de fuente de datos*
2. Habilite la pestaña *Texto Delimitado*
3. Seleccione el archivo de texto delimitado a importar (p.ej., `qgis_sample_data/csv/elevp.csv`) clickando en el botón ... Navegador.
4. Configure los ajustes para conocer tu conjunto de datos y necesidades, como se eplica abajo.

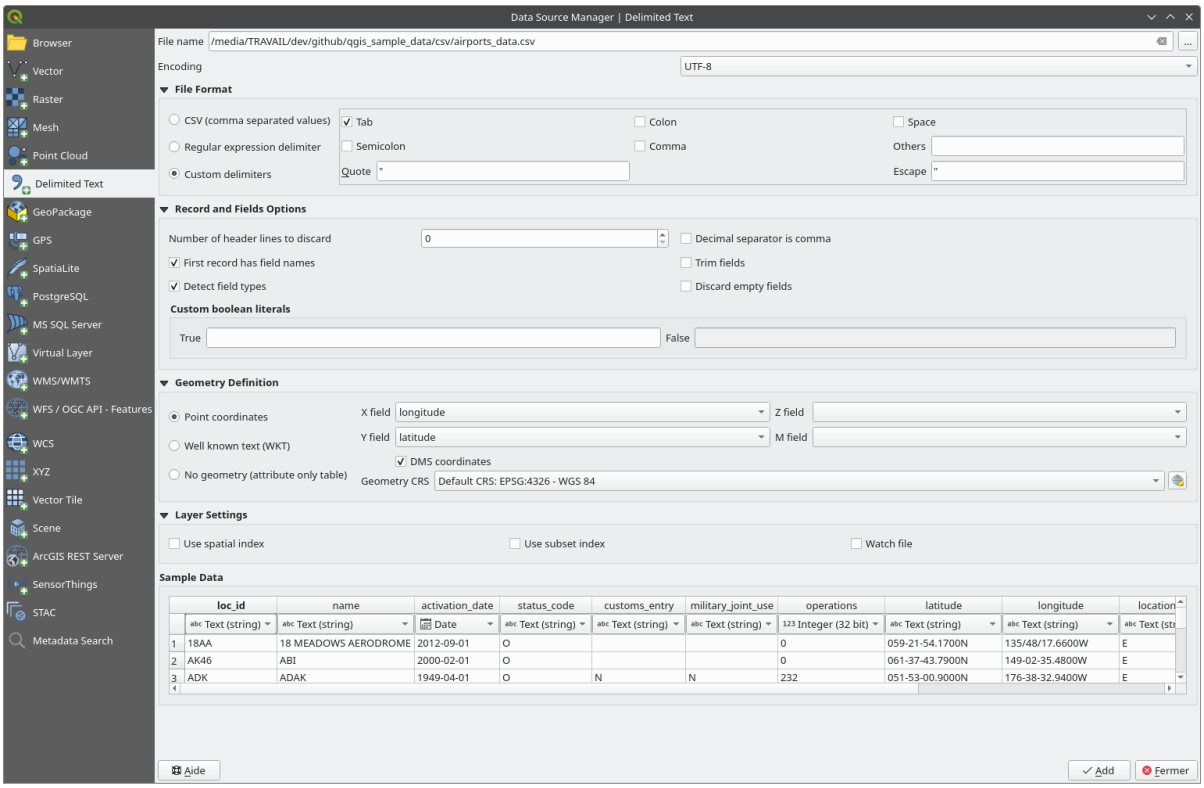


Figura 11.9: Diálogo de Texto Delimitado

Formato de Archivo

Una vez que se selecciona el archivo, QGIS intenta analizar el archivo con el delimitador utilizado más recientemente, identificando campos y filas. Para permitir que QGIS analice correctamente el archivo, es importante seleccionar el delimitador correcto. Puede especificar un delimitador eligiendo entre:

- *CSV (valores separados por comas)* para usar un carácter coma.
- *Delimitador de expresión regular* e introduce texto en el campo *Expresión*. Por ejemplo, para cambiar el delimitador a tabulación, use `\t` (esto se usa en expresiones regulares para el carácter de tabulación).
- *Personalizar delimitadores*, elegir entre algunos delimitadores predefinidos como `comma`, `space`, `tab`, `semicolon`, ...

Registros y campos

Algunas otras opciones convenientes pueden ser usadas para reconocimiento de datos:

- *Número de líneas de encabezado a descartar*: conveniente cuando desea evitar las primeras líneas en el archivo en la importación, ya sea porque son líneas en blanco o con otro formato.
- *El primer registro tiene nombres de campo*: los valores en la primera línea se usan como nombres de campo, de lo contrario QGIS usa los nombres de campo `field_1`, `field_2`...
- *Detectar tipos de campo*: reconoce automáticamente el tipo de campo. Si no se marca, todos los atributos se tratan como campos de texto.
- *El separador decimal es una coma*: puede forzar que el separador decimal sea una coma.
- *Recortar campos*: le permite recortar los espacios iniciales y finales de los campos.
- :guilabel:Descartar campos vacíos`.
- *Literales booleanos personalizados*: permite añadir un par de cadenas personalizadas que se detectarán como valores booleanos.

Detección del tipo de campo

QGIS intenta detectar los tipos de campo automáticamente (a menos que *Detectar tipos de campo* no esté marcada) examinando el contenido de un archivo CSVT sidecar opcional (ver Especificación GeoCSV) y escaneando todo el archivo para asegurarse de que todos los valores pueden ser realmente convertidos sin errores, el tipo de campo fall-back es texto.

El tipo de campo detectado aparece bajo el nombre del campo en la tabla de previsualización de datos de muestra y puede modificarse manualmente si es necesario.

Se admiten los siguientes tipos de campo:

- Las parejas literales que no distinguen mayúsculas de minúsculas y que se interpretan como valores booleanos son `1`/`0`, `true`/`false`, `t`/`f`, `yes`/`no`.
- `Número entero (integer)`
- `Número entero (entero - 64 bits)`
- `Número decimal`: número en coma flotante de doble precisión
- `Date`
- `Time`
- `Fecha y hora`
- `Texto`

Definición de geometría

Una vez que se analiza el archivo, configure *Definición de geometría*

- *coordenadas de punto* y proporcione el *campo X*, *campo Y*, *campo Z* (para datos de 3 dimensiones) y *campo M* (para la dimensión de medición) si la capa es del tipo de geometría puntual y contiene dichos campos. Si las coordenadas están definidas como grados / minutos / segundos, active la casilla de verificación *Coordenadas DMS*. Proporcione el apropiado *SRC de Geometría* usando el widget Seleccionar SRC.

- La opción *Well known text (WKT)* si la información espacial se representa como WKT: seleccione el *Campo de geometría* que contiene la geometría de WKT y elija el apropiado *Campo de geometría* o deje que QGIS lo detecte automáticamente. Proporcione el *SRC de Geometría* apropiado usando el widget Seleccionar SRC.
- Si el archivo contiene datos no espaciales, active : guilabel:*Sin geometría (solo tabla de atributos)* y se cargará como una tabla ordinaria.

Configuración de capa

Adicionálmente, puede habilitar:

- *Usar índice espacial* para mejorar el rendimiento de visualización y selección espacial de entidades.
- :guilabel:»Utilizar índice de subconjunto» para mejorar el rendimiento de *filtros de subconjunto* (cuando se define en las propiedades de la capa).
- *Ver archivo* para observar cambios en el archivo por otras aplicaciones mientras QGIS se está ejecutando.

Al final, haga clic en:guilabel:‘Add` para agregar la capa al mapa. En nuestro ejemplo, una capa de puntos llamada `Elevación` se agrega al proyecto y se comporta como cualquier otra capa de mapa en QGIS. Esta capa es el resultado de una consulta en el archivo fuente `.csv` (por lo tanto, vinculado a él) y requeriría *ser guardado* para obtener una capa espacial en el disco.

Datos de muestra

A medida que ajusta las propiedades del analizador sintáctico, la vista previa de los datos de muestra se actualiza en función de los ajustes aplicados.

También en la Tabla de Datos de Muestra es posible anular los tipos de columna determinados automáticamente.

Importando un archivo DXF o DWG

Los archivos `DXF` y `DWG` se pueden agregar a QGIS simplemente arrastrando y soltando desde el Panel del navegador. Se le pedirá que seleccione las subcapas que le gustaría agregar al proyecto. Las capas se agregan con propiedades de estilo aleatorias.

> **Nota**
>
> Para archivos DXF que contienen varios tipos de geometría (punto, línea y / o polígono), el nombre de las capas se generará como *<filename.dxf> entidades <geometry type>*.

Para mantener la estructura del archivo dxf/dwg y su simbología en QGIS, es posible que desee utilizar la dedicada *Proyecto -> Importar / Exportar -> Importar capas desde DWG / DXF …* :

1. Importa elementos desde el archivo de dibujo en la ase de datos de GeoPackage.
2. añadir elementos importados al proyecto..

En el diálogo *Importar DWG/DXF*, para importar los contenidos del archivo de dibujo:

1. Introduzca la ubicación del *Dibujo de origen*, es decir, el archivo de dibujo DWG/DXF a importar.
2. Especifique el sistema de referencia de coordenadas de los datos en el archivo de dibujo.
3. Introduzca la ubicación del *Paquete de destino*, es decir, el archivo GeoPackage que almacenará los datos. Si se proporciona un archivo existente, se sobrescribirá.
4. Elija cómo importar `bloques` con el combobox dedicado:

- *Expandir geometrías de bloques*: importa los bloques del archivo de dibujo como elementos normales.
- *Expandir geometrías de bloques y añadir puntos de inserción*: importa los bloques del archivo de dibujo como elementos normales y añade el punto de inserción como una capa de puntos.
- *Añadir sólo puntos de inserción*: añade el punto de inserción de los bloques como una capa de puntos.

5. Marque *Usar curvas* para promover las capas importadas a un tipo de geoemtría `curvada`.
6. Utilice el botón *Importar* para importar el dibujo en el archivo GeoPackage de destino. La base de datos de GeoPackage se llenará automáticamente con el contenido del archivo de dibujo. Dependiendo del tamaño del archivo, esto puede llevar algún tiempo.

Despues de que los datos `.dwg` or `.dxf` se ha importado a la base de datos de GeoPackage, el marco de la mitad inferior del cuadro de diálogo se rellena con la lista de capas del archivo importado. Allí puede seleccionar qué capas agregar al proyecto QGIS:

1. En la parte superior, establezca una etiqueta *Nombre del grupo* para agrupar los archivos de dibujo en el proyecto. Predeterminado: el nombre del archivo de dibujo de origen.
2. Marque las capas para mostrar: cada capa seleccionada se agrega a un grupo ad hoc que contiene capas vectoriales para las características de punto, línea, etiqueta y área de la capa de dibujo. El estilo de las capas se parecerá al aspecto que tenían originalmente en *CAD.
3. Elige si la capa debe ser visible al abrirla.
4. Marcando la opción *Mezclar capas* ubica todas las capas en un único grupo.
5. Presione *Aceptar* para abrir las capas en QGIS.

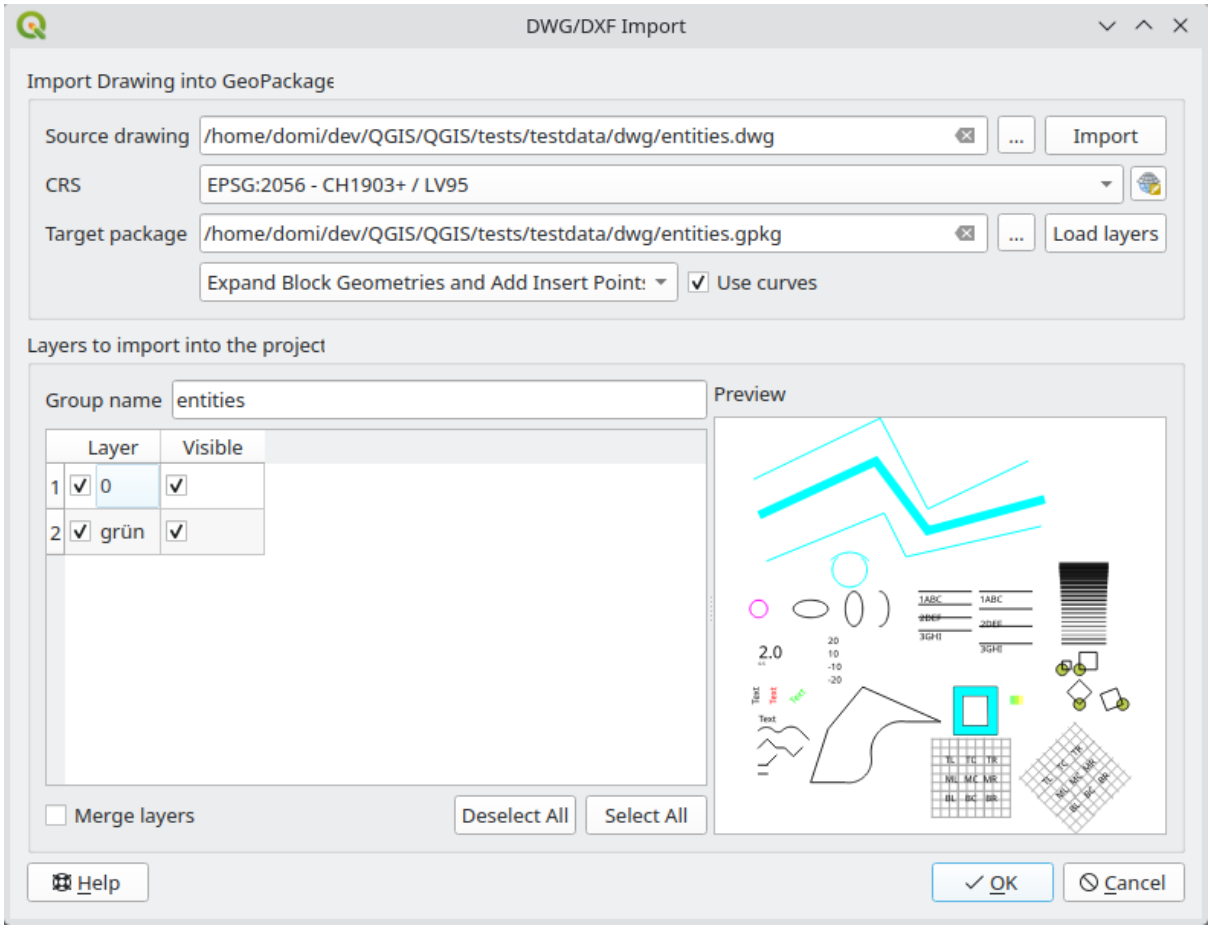


Figura 11.10: Diálogo de importación para archivos DWG/DXF

Importar Vectoriales de OpenStreetMap

El proyecto OpenStreetMap es popular porque en muchos países no se encuentran disponibles geodatos gratuitos como mapas de carreteras digitales. El objetivo del proyecto OSM es crear un mapa editable gratuito del mundo a partir de datos GPS, fotografías aéreas y conocimiento local. Para respaldar este objetivo, QGIS brinda soporte para datos OSM.

Usando el *Panel de Navegador*, puede cargar un archivo `.osm` en el lienzo del mapa, en cuyo caso aparecerá un diálogo para seleccionar subcapas según el tipo de geometría. Las capas cargadas contendrán todos los datos de ese tipo de geometría en el archivo `.osm`, y mantendrán la estructura de datos del archivo `osm`.

Capas SpatiaLite

 La primera vez que carga datos de una base de datos SpatiaLite, comience por:

- clickando en el botón de la barra de herramientas  Añadir capa SpatiaLite
- seleccionando la opción  *Añadir capa SpatiaLite.*. del menú *Capa* ► *Añadir capa*
- o pulsando `Ctrl+Shift+L`

Esto abrirá una ventana que le permitirá conectarse a una base de datos SpatiaLite ya conocida por QGIS (que elija en el menú desplegable) o definir una nueva conexión a una nueva base de datos. Para definir una nueva conexión, haga clic en *Nuevo* y use el navegador de archivos para apuntar a su base de datos SpatiaLite, que es un archivo con una extensión `.sqlite`.

QGIS también soporta vistas editables en SpatiaLite.

GPS

Hay docenas de formatos de archivo diferentes para almacenar datos GPS. El formato que utiliza QGIS se llama GPX (formato de intercambio de GPS), que es un formato de intercambio estándar que puede contener cualquier número de waypoints, rutas y tracks en el mismo archivo.

Utilice el botón … Explorar para seleccionar el archivo GPX y, a continuación, utilice las casillas de verificación para seleccionar los tipos de características que desea cargar desde ese archivo GPX. Cada tipo de característica se cargará en una capa separada.

Más información sobre la manipulación de datos GPS en *Trabajar con datos GPS*.

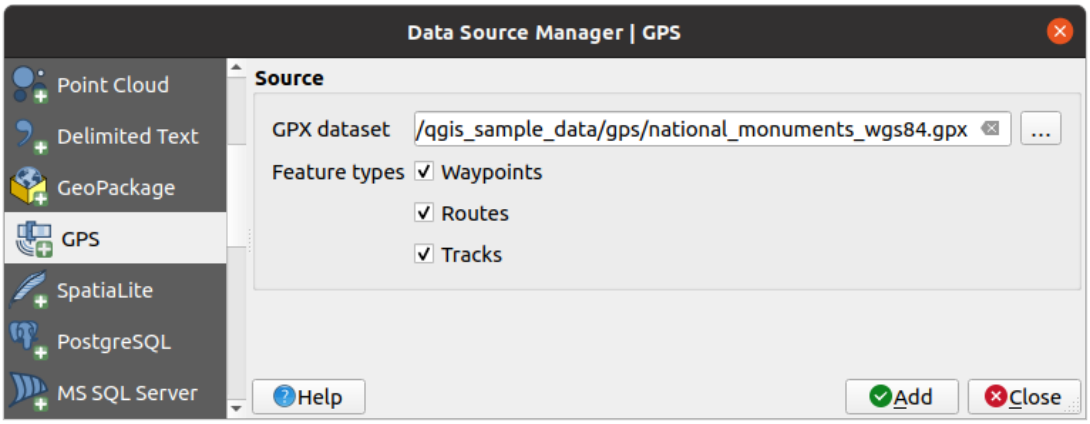


Figura 11.11: Diálogo de carga de datos GPS

GRASS

El trabajo con datos vectoriales de GRASS se describe en la sección *Integracion GRASS SIG*.

Herramientas relacionadas con la base de datos

Crear una conexión almacenada

Para leer y escribir tablas desde un formato de base de datos compatible con QGIS, debe crear una conexión a esa base de datos. Mientras *Panel del navegador QGIS* es la forma más sencilla y recomendada de conectarse y utilizar bases de datos, QGIS proporciona otras herramientas para conectarse a cada una de ellas y cargar sus tablas:

- *Add PostgreSQL Layer…* or by typing `Ctrl+Shift+D`
- *Añadir la capa espacial de Oracle…* o tecleando `Ctrl+Shift+O`
- *Añadir capa de MS SQL Server*.
- *Añadir capa espacial de SAP HANA…* o tecleando `Ctrl+Shift+G`

Estas herramientas son accesibles desde *Barra de herramientas de administración de capas* y desde el menú :menuselection:` Capa -> Agregar capa -> . *La conexión a la base de datos SpatiaLite se describe en :ref:`label_spatialite*.

> **Truco**
>
> **Cree una conexión a la base de datos desde el panel del navegador de QGIS**
>
> Al seleccionar el formato de base de datos correspondiente en el árbol del navegador, hacer clic con el botón derecho y elegir conectar, aparecerá el cuadro de diálogo de conexión de la base de datos.

La mayoría de los diálogos de conexión siguen una estructura común:

- una sección con información sobre las credenciales para conectarse a la base de datos
- una sección con opciones para ajustar qué datos pueden solicitarse en la base de datos

Conexión a PostgreSQL

The first time you use a PostgreSQL data source, you must create a connection to a database that contains the data. Press the appropriate button as exposed above, opening the *PostgreSQL* tab of the *Data Source Manager* dialog. To access the connection manager, click on the *New* button to display the *Create a New PostgreSQL Connection* dialog.

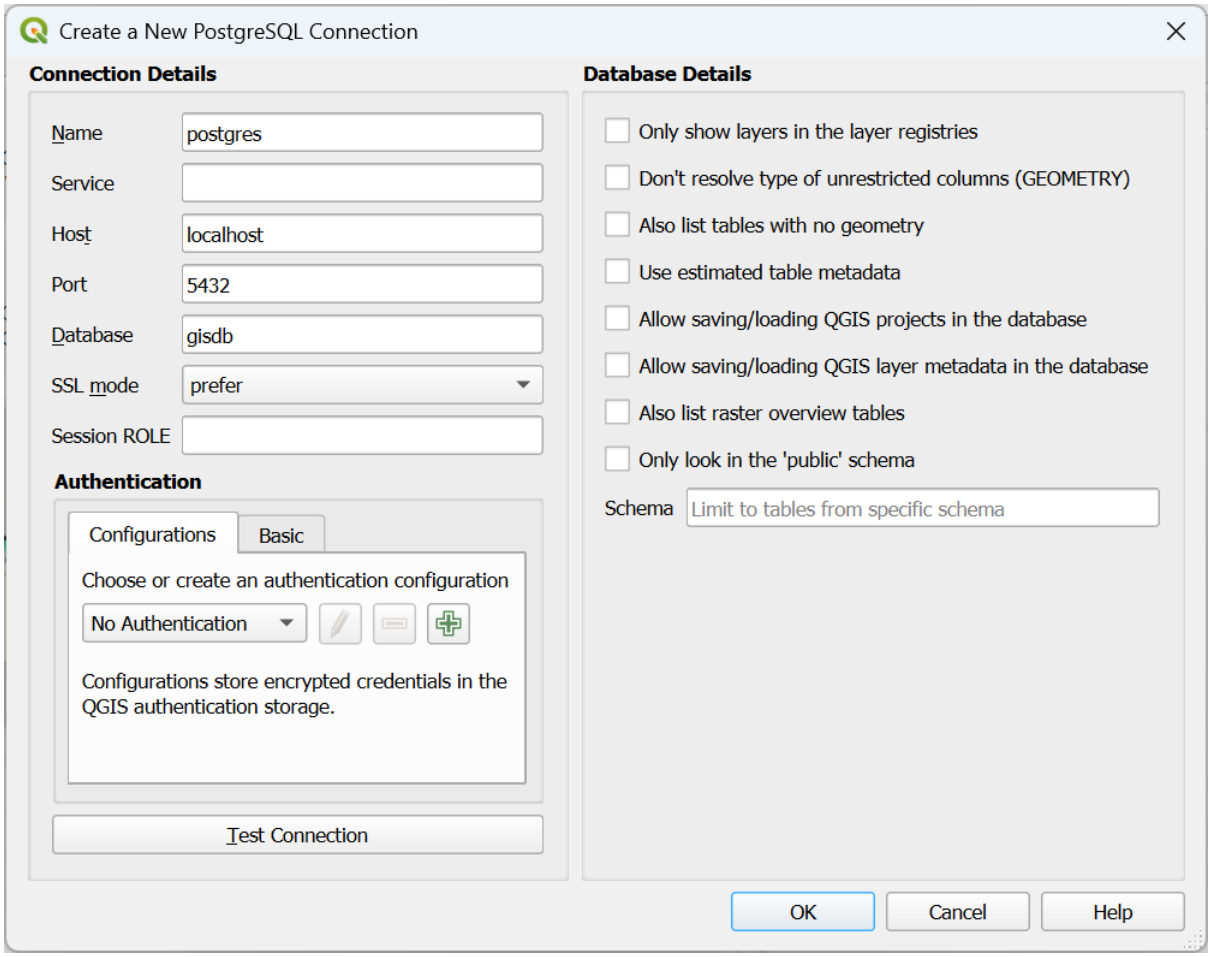


Figura 11.12: Create a New PostgreSQL Connection Dialog

- *Nombre*: Un nombre para esta conexión. Puede ser lo mismo que *Database*.
- *Service*: Service parameter to be used alternatively to hostname/port (and potentially database). This can be defined in `pg_service.conf`. Check the *PostgreSQL Service connection file* section for more details.
- *Host*: Nombre del host de la base de datos. Debe ser un nombre de host que se pueda resolver, como el que se utilizaría para abrir una conexión TCP / IP o hacer ping al host. Si la base de datos está en la misma computadora que QGIS, simplemente ingrese *localhost* aquí.
- *Port*: Port number the PostgreSQL database server listens on. The default port for PostgreSQL is `5432`.
- *Base de Datos*: Nombre de la base de datos.
- *Modo SSL*: Configuración del cifrado SSL. Están disponibles las siguientes opciones:
  - *Prefer* (valor predeterminado): no me importa el cifrado, pero deseo pagar los gastos generales del cifrado si el servidor lo admite.
  - *Requerimiento*: Quiero que mis datos estén encriptados y acepto los gastos generales. Confío en que la red se asegurará de que siempre me conecte al servidor que quiero.
  - *Verificar CA*: Quiero que mis datos estén encriptados y acepto los gastos generales. Quiero estar seguro de que me conecto a un servidor en el que confío.

  - *Verificar completo*: Quiero que mis datos estén encriptados y acepto los gastos generales. Quiero estar seguro de que me conecto a un servidor en el que confío y que es el que especifico.
  - *Permitir*: No me importa la seguridad, pero pagaré los gastos generales del cifrado si el servidor insiste en ello.
  - *Deshabilitar*: No me importa la seguridad y no quiero pagar los gastos generales del cifrado.
- *Rol de sesión*: se utiliza para establecer el identificador de usuario de la sesión actual. Esto es útil para dar automáticamente la propiedad de un nuevo objeto (tabla, vista, función) al grupo session_role y así compartir la propiedad y los derechos asociados con todos los miembros del grupo session_role. Más información sobre session role.
- *Authentication*: For general details about the authentication dialog behavior, see *Autenticación*.

Opcionalmente, dependiendo del tipo de base de datos, puede activar las siguientes casillas de verificación:

- ☑ *Mostrar solo capas en los registros de capas*
- ☑ *No resuelva el tipo de columnas no restringidas (GEOMETRÍA)*
- ☑ *Enumerar también tablas sin geometría*: indica que las tablas sin geometría también deben aparecer en la lista por defecto.
- ☑ *Utilizar metadatos estimados de la tabla*: Al inicializar las capas, pueden ser necesarias varias consultas para establecer las características de las geometrías almacenadas en la tabla de la base de datos. Cuando esta opción está marcada, estas consultas examinan sólo una muestra de las filas y utilizan las estadísticas de la tabla, en lugar de la tabla completa. Esto puede acelerar drásticamente las operaciones en grandes conjuntos de datos, pero puede dar lugar a una caracterización incorrecta de las capas (por ejemplo, el recuento de características de las capas filtradas no se determinará con precisión) e incluso puede provocar un comportamiento extraño si columnas que se supone que son únicas en realidad no lo son.
- ☑ *Permitir guardar/cargar proyectos QGIS en la base de datos* - más detalles *aquí*
- ☑ *Permitir guardar/cargar metadatos de capas QGIS en la base de datos* - más detalles *aquí*
- ☑ *Also list raster overview tables*
- ☑ *Busque solo en el esquema "público"*
- *Esquema*: Permite especificar un único esquema al que limitar una conexión. Cuando se establece, sólo las tablas del esquema coincidente se incluirán en el panel del navegador y en la selección del origen de datos para la conexión. Esto puede utilizarse para limitar el trabajo de base de datos necesario para rellenar las tablas de una conexión que apunte a un almacén de base de datos grande.

Una vez configurados todos los parámetros y opciones, puede probar la conexión pulsando el botón *Probar conexión* o aplicarla pulsando el botón *Aceptar*.

PostgreSQL Service connection file

El archivo de conexión de servicio permite que los parámetros de conexión de PostgreSQL se asocien con un solo nombre de servicio. Luego, un cliente puede especificar ese nombre de servicio y se utilizarán las configuraciones asociadas.

Es llamada `.pg_service.conf` en sistemas *nix (GNU/Linux, macOS etc.) y `pg_service.conf` en Windows.

El archivo de servicio puede verse como:

```
[water_service]
host=192.168.0.45
port=5433
dbname=gisdb
user=paul
password=paulspass
```

(continúe en la próxima página)

(proviene de la página anterior)

```
[wastewater_service]
host=dbserver.com
dbname=water
user=waterpass
```

Nota

Hay dos servicios en el ejemplo anterior: `water_service` y `wastewater_service`. Puedes usarlos para conectarte desde QGIS, pgAdmin, etc. especificando sólo el nombre del servicio al que quieres conectarte (sin los corchetes). Si quieres usar el servicio con `psql`, puedes hacer `psql service=water_service`.

Puede encontrar todos los parámetros de PostgreSQL aquí

Nota

Si no desea guardar las contraseñas en el archivo de servicio, puede usar el .pg_pass option.

Nota

**Servidor y servicio QGIS

Cuando se utiliza un archivo de servicio y QGIS Server, debe configurar el servicio en el lado del servidor también. Puede seguir la documentación QGIS Server.

En los sistemas operativos *nix (GNU/Linux, macOS, etc.) puede guardar el archivo: archivo: *.pg_service.conf* en el directorio de inicio del usuario y los clientes de PostgreSQL lo reconocerán automáticamente. Por ejemplo, si el usuario registrado es `web`, `.pg_service.conf` debe guardarse en el directorio :file:`/home/web/`directorio para trabajar directamente (sin especificar ningún otro entorno variables).

Puede especificar la ubicación del archivo de servicio creando una variable de entorno `PGSERVICEFILE` (por ejemplo, ejecute el comando `export PGSERVICEFILE=/home/web/.pg_service.conf` en su sistema operativo *nix para establecer temporalmente la variable `PGSERVICEFILE`).

También puede hacer que el archivo de servicio esté disponible en todo el sistema (todos los usuarios) colocando el archivo `.pg_service.conf` en `pg_config --sysconfdir` o agregando la variable de entorno especifique el directorio que contiene el archivo de servicio. Si existen definiciones de servicio con el mismo nombre en el usuario y en el archivo del sistema, el archivo del usuario tiene prioridad.

Advertencia

Hay algunas consideraciones bajo Windows:

- El archivo de servicio debe guardarse como: file: *pg_service.conf* y no como: file:` .pg_service.conf`.
- El archivo de servicio debe guardarse en formato Unix para que funcione. Una forma de hacerlo es abrirlo con Notepad ++ y *Editar -> Conversión EOL -> Formato UNIX -> Guardar archivo*.
- Puede agregar variables ambientales de varias formas; uno probado, que se sabe que funciona de manera confiable, es *Panel de control -> Sistema y seguridad -> Sistema -> Configuración avanzada del sistema -> Variables de entorno* agregando `PGSERVICEFILE` con la ruta - e.g. `C:\Users\John\pg_service.conf`
- Después de agregar una variable de entorno, es posible que también deba reiniciar la computadora.

Conectando a Oracle Spatial

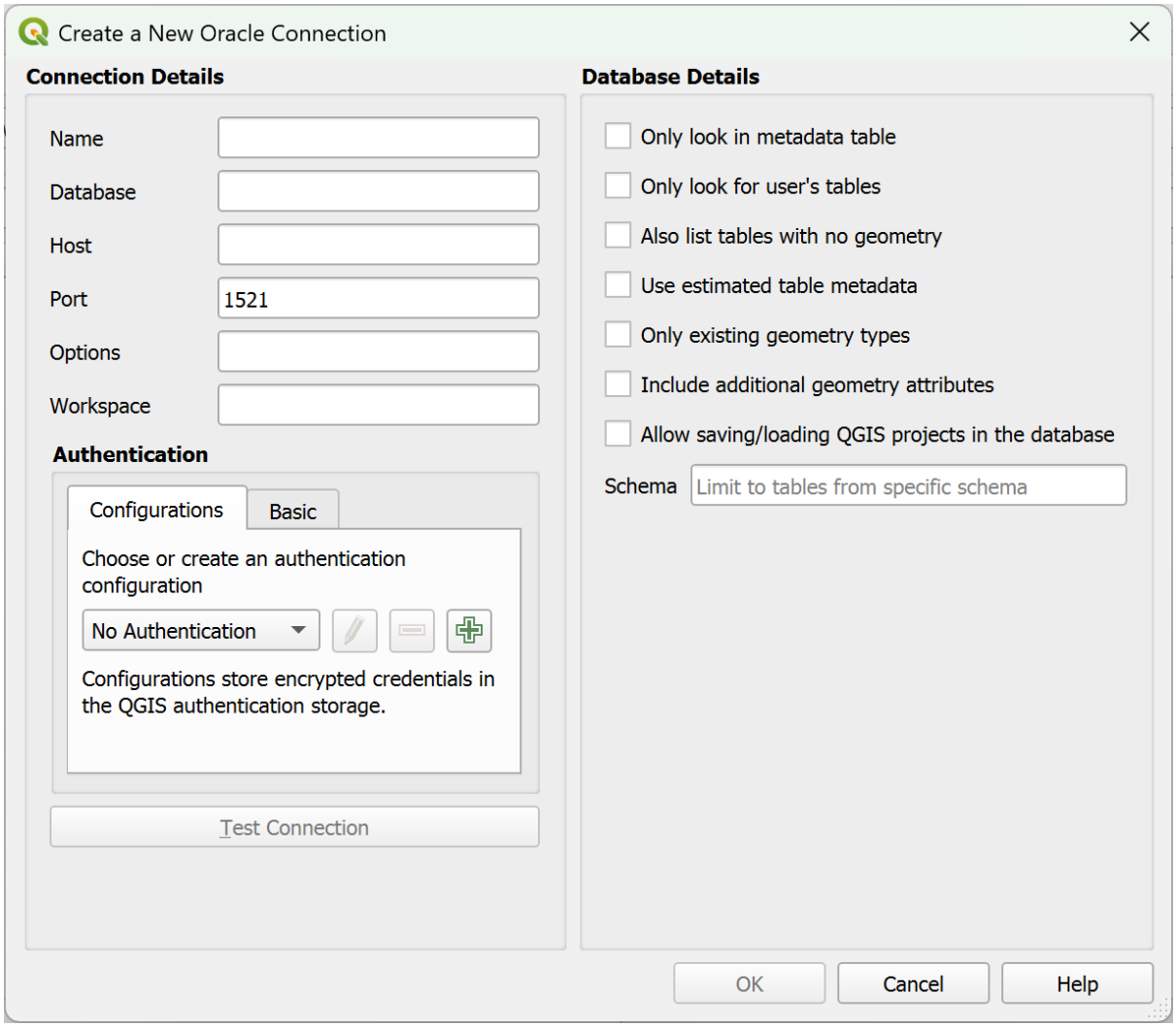


Figura 11.13: Create a New Oracle Connection Dialog

Las características espaciales de Oracle Spatial ayudan a los usuarios a gestionar datos geográficos y de localización de tipo nativo dentro de una base de datos Oracle. El diálogo de conexión propone:

- *Name*: A name for this connection. It can be the same as *Database*;
- *Database*: SID or SERVICE_NAME of the Oracle instance;
- *Host*: The name of the database host;
- *Port*: Port number the Oracle database server listens on. The default port is `1521`;
- *Options*: Oracle connection specific options (e.g. OCI_ATTR_PREFETCH_ROWS, OCI_ATTR_PREFETCH_MEMORY). The format of the options string is a semicolon separated list of option names or option=value pairs;
- *Workspace*: Workspace to switch to;
- *Authentication*: For general details about the authentication dialog behavior, see *Autenticación*.

Opcionalmente, puede activar las siguientes casillas de verificación:

- :guilabel:‘Sólo buscar en la tabla de metadatos`: restringe las tablas mostradas a las que están en la vista `todos_sdo_geom_metadatos`. Esto puede acelerar la visualización inicial de tablas espaciales.

- *Sólo buscar tablas del usuario*: al buscar tablas espaciales, restringe la búsqueda a tablas que son propiedad del usuario.
- *Enumerar también tablas sin geometría*: indica que las tablas sin geometría también deben aparecer en la lista por defecto.
- *Utilice estadísticas de tabla estimadas para los metadatos de la capa*: cuando se configura la capa, se requieren varios metadatos para la tabla de Oracle. Esto incluye información como el recuento de filas de la tabla, el tipo de geometría y las extensiones espaciales de los datos en la columna de geometría. Si la tabla contiene una gran cantidad de filas, determinar estos metadatos puede llevar mucho tiempo. Al activar esta opción, se realizan las siguientes operaciones rápidas de metadatos de tabla: El recuento de filas se determina a partir de *'all_tables.num_rows`*. Las extensiones de la tabla siempre se determinan con la función SDO_TUNE.EXTENTS_OF, incluso si se aplica un filtro de capa. La geometría de la tabla se determina a partir de las primeras 100 filas de geometría no nula de la tabla.
- *Sólo tipos de geometría existentes*: solo enumera los tipos de geometría existentes y no ofrece agregar otros.
- *Incluir atributos de geometría adicionales*.
- *Permitir guardar/cargar proyectos QGIS en la base de datos* - más detalles *aquí*
- *Esquema*: Permite especificar un único esquema al que limitar una conexión. Cuando se establece, sólo las tablas del esquema coincidente se incluirán en el panel del navegador y en la selección del origen de datos para la conexión. Esto puede utilizarse para limitar el trabajo de base de datos necesario para rellenar las tablas de una conexión que apunte a un almacén de base de datos grande.

Truco

Capas Oracle Spatial

Normalmente, una capa espacial de Oracle se define mediante una entrada en la tabla **USER_SDO_METADATA**.

Para asegurarse de que las herramientas de selección funcionen correctamente, se recomienda que sus tablas tengan una **clave principal**.

Conexión a MS SQL Server

Como se menciona en *Crear una conexión almacenada* QGIS permite crear conexión MS SQL Server a través de *Gestor de fuentes de datos*.

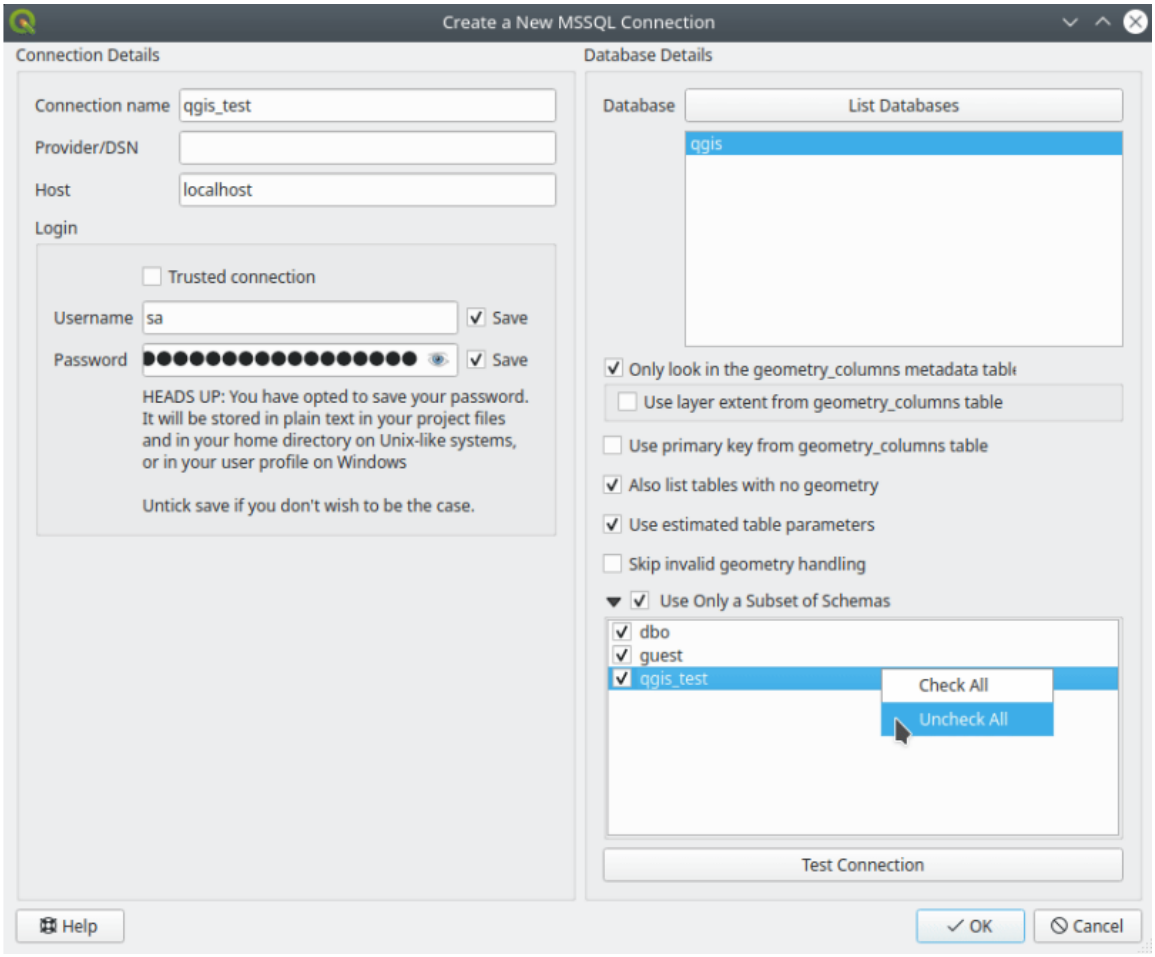


Figura 11.14: Conexión a MS SQL Server

Para crear una nueva conexión a MS SQL Server, debe proporcionar algunos de los siguientes datos en el cuadro de diálogo *Detalles de conexión*:

- *Nombre de la conexión*
- *Provider/DSN*
- *Anfitrión*
- *Información de inicio de sesión*. Puede elegir ☑ *Guardar* sus credenciales.

Navegue hasta la sección *Detalles de la base de datos* y pulse el botón *Lista de bases de datos* para ver los conjuntos de datos disponibles. Seleccione los conjuntos de datos que desee y pulse *Aceptar*. Opcionalmente, también puede realizar una *Prueba de conexión*. Una vez que pulse *Aceptar* se cerrará el cuadro de diálogo *Crear una nueva conexión a MS SQL Server* y en el *Administrador de fuentes de datos* pulse *Conectar*, seleccione una capa y pulse *Añadir*.

Opcionalmente, puede activar las siguientes opciones:

- ☑ *Buscar solo en la tabla de metadatos geometry_columns*: restringe las tablas disponibles a las que se encuentran en la tabla de metadatos `geometry_columns` al buscar tablas. Esto puede acelerar la búsqueda de tablas.
- ☑ *Usar extensión de capa de la tabla geometry_columns*: esta opción, dependiente de la anterior, permite a QGIS omitir el cálculo de la extensión al cargar capas y, por lo tanto, reducir el tiempo necesario para cargarlas. Se basa en la extensión especificada manualmente utilizando columnas adicionales específicas de QGIS (`qgis_xmin`, `qgis_xmax`, `qgis_ymin`, `qgis_ymax`) en la tabla `geometry_columns`.
- ☑ *Usar clave primaria de la tabla geometry_columns*: permite a QGIS omitir el cálculo de la clave primaria

para las vistas al cargarlas, lo que reduce el tiempo necesario para cargarlas. Se basa en los nombres introducidos manualmente en una columna específica de QGIS denominada `qgis_pkey`, establecida en la tabla `geometry_columns`. Si se utiliza más de una columna para la clave primaria, deben introducirse valores separados por comas.

- ☑ *También listar tabla sin geometría*: las tablas sin una columna de geometría adjunta también se mostrarán en la lista de tablas disponibles.
- ☑ *Utilizar parámetros de tabla estimados*: sólo se utilizarán metadatos de tabla estimados. Esto evita un escaneo lento de la tabla, pero puede dar lugar a propiedades de capa incorrectas, como la extensión de la capa.
- ☑ *Omitir el manejo de geometría inválida*: se desactivará todo el manejo de registros con geometría inválida. Esto acelera el proceso del proveedor, sin embargo, si hay geometrías no válidas en una tabla, el resultado es impredecible y puede incluir registros perdidos. Marque esta opción sólo si está seguro de que todas las geometrías presentes en la base de datos son válidas, y de que cualquier geometría o tabla añadida recientemente también será válida.
- ☑ *Utilizar sólo un subconjunto de esquemas* le permitirá filtrar los esquemas para la conexión MS SQL. Si se activa, sólo se mostrarán los esquemas marcados. Puede hacer clic con el botón derecho del ratón para *Marcar* o *Desmarcar* cualquier esquema de la lista.

Conexión a SAP HANA

Nota

Necesita el cliente de SAP HANA para conectarse a una base de datos de SAP HANA. Puede descargar el cliente SAP HANA para su plataforma en el sitio web de herramientas de desarrollo de SAP <https://tools.hana.ondemand.com/#hanatools>`_.

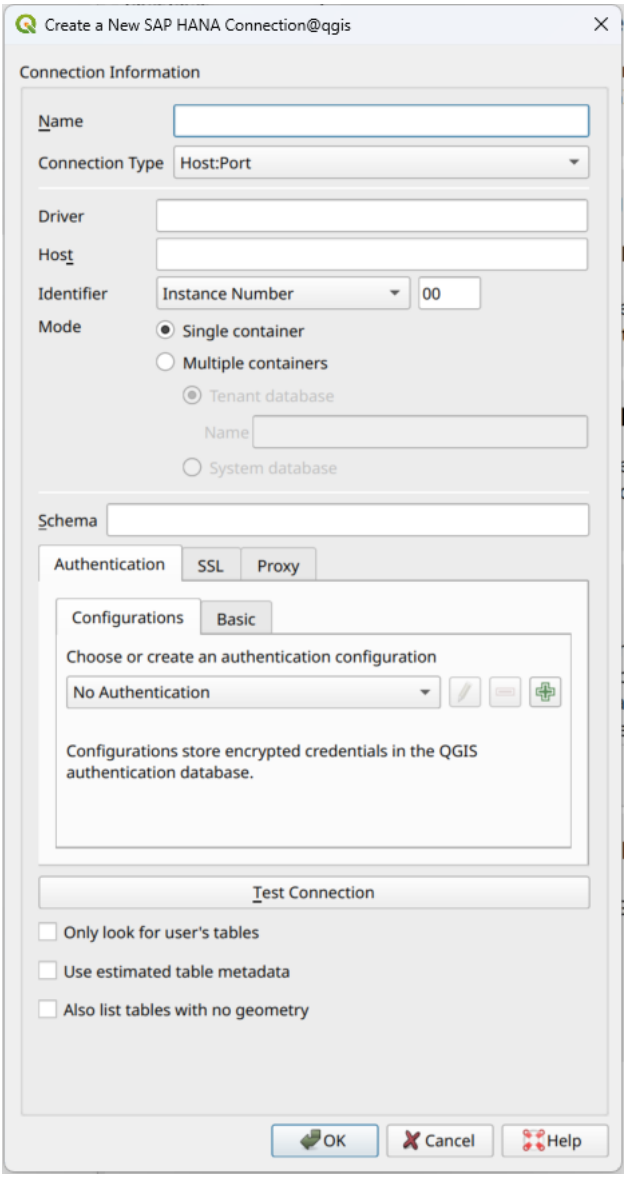


Figura 11.15: Crear un nuevo diálogo de conexión SAP HANA

Se pueden introducir los siguientes parámetros:

- *Name*: Un nombre para esta conexión.
- *Driver* : El nombre del controlador ODBC de HANA. Es `HDBODBC` si usa QGIS de 64 bits, `HDBODBC32` si usa QGIS de 32 bits. El nombre del controlador apropiado se ingresa automáticamente.
- *Driver* : El nombre con el que se ha registrado el controlador ODBC de SAP HANA en `/etc/odbcinst.ini` o la ruta completa al controlador ODBC de SAP HANA. El instalador del cliente SAP HANA instalará el controlador ODBC en `/usr/sap/hdbclient/libodbcHDB.so` de manera predeterminada.
- *Host*: Nombre de la base de datos del anfitrión
- *Identifier*: Identifica la instancia a la que conectarse en el host. Puede ser *Instance Number* o *Port Number*. Los números de instancia constan de dos dígitos, los números de puerto están en el rango de 1 a 65,535.
- *Mode*: Especifica el modo en el que se ejecuta la instancia de SAP HANA. Esta configuración solo se tiene en cuenta si *Identifier* está configurado en *Instance Number*. Si la base de datos alberga varios contenedores, puede conectarse a un inquilino con el nombre indicado en *Tenant database* o puede conectarse a la base de datos del sistema.

- *Schema*: Este parámetro es opcional. Si se da un nombre de esquema, QGIS solo buscará datos en ese esquema. Si este campo se deja en blanco, QGIS buscará datos en todos los esquemas.
- *Authentication*: For general details about the authentication dialog behavior, see *Autenticación*.
- *Configuración SSL*
  - *Enable TLS/SSL encryption*: Habilita el cifrado TLS 1.1 - TLS1.2. El servidor elegirá el más alto disponible.
  - *Provider*: Especifica el proveedor de la biblioteca criptográfica utilizada para la comunicación SSL. *sapcrypto* debería funcionar en todas las plataformas, *openssl* debería funcionar en , *mscrypto* debería funcionar en y *commoncrypto* requiere la instalación de CommonCryptoLib.
  - *Validate SSL certificate*: Si está marcado, el certificado SSL se validará utilizando el almacén de confianza proporcionado en *Trust store file with public key*.
  - *Override hostname in certificate*: Especifica el nombre de host utilizado para verificar la identidad del servidor. El nombre de host especificado aquí verifica la identidad del servidor en lugar del nombre de host con el que se estableció la conexión. Si especifica `*` como nombre de host, el nombre de host del servidor no se valida. No se permiten otros comodines.
  - *Archivo de almacenamiento de claves con clave privada*: Actualmente se ignora. Este parámetro podría permitir la autenticación a través de un certificado en lugar de un usuario y una contraseña en el futuro.
  - *Trust store file with public key*: Especifica la ruta a un archivo de almacén de confianza que contiene los certificados públicos del servidor si se usa OpenSSL. Normalmente, el almacén de confianza contiene el certificado raíz o el certificado de la autoridad de certificación que firmó los certificados públicos del servidor. Si está utilizando la biblioteca criptográfica CommonCryptoLib o msCrypto, deje esta propiedad vacía.
- *Only look for user's tables*: Si está marcado, QGIS busca solo tablas y vistas que son propiedad del usuario que se conecta a la base de datos.
- *Usar metadatos de tabla estimados*: Si se marca, se utilizarán los metadatos estimados de la tabla si están disponibles. Para tablas grandes, esto evita cargas lentas de la tabla y cálculos potencialmente costosos, pero puede dar lugar a propiedades de capa incorrectas, como la extensión de la capa. La estimación rápida de la extensión está disponible a partir de QRC1/2024 y SP8 en HANA Cloud y HANA On-Premise respectivamente.
- *Also list tables with no geometries*: Si está marcado, QGIS busca también tablas y vistas que no contengan una columna espacial.

Truco

Conectando a SAP HANA Cloud

Si desea conectarse a una instancia de SAP HANA Cloud, generalmente debe establecer *Port Number* en `443` y marcar *Enable TLS/SSL encryption*.

Cargando una Capa Base de Datos

Once you have one or more connections defined to a database (see section *Crear una conexión almacenada*), you can load layers from it. Of course, this requires that data are available. See section *Importando Datos en PostgreSQL* for a discussion on importing data into a PostgreSQL database.

Para cargar una capa desde una base de datos, puede realizar los siguientes pasos:

1. Abra la pestaña correspondiente de la base de datos en el cuadro de diálogo *Gestor de fuentes de datos*.
2. Elija el nombre de la conexión en la lista desplegable y pulse *Conectar*.
3. The table below will be filled with your data grouped by schema, with a number of metadata information helpful for loading.
4. Seleccione o deseleccione *También enumerar tablas sin geometría*.
5. Opcionalmente, use algunas *Opciones de Búsqueda* para reducir la lista de tablas a aquellas que coincidan con su búsqueda. También puede configurar esta opción antes de presionar el botón *Conectar*, acelerando la búsqueda de la base de datos.
6. Busque la capa(s) que desea agregar en la lista de capas disponibles.
7. Selecciónelo haciendo clic en él. Puede seleccionar varias capas manteniendo presionada la tecla `Shift` o `Ctrl` mientras hace clic.

   Layers can be selected only if they have no warning icon at the left-hand side of their first column. This may indicate an issue to detect:

   - features geometry type: in which case you can select the appropriate one in the drop-dowmn list of the *Spatial Type* column
   - layer CRS: you can enter the correct code in the *SRID* column
   - layer's primary key, in order to unequivocally identify each feature: this can be fixed by selecting one or more attributes in the drop-down list at the corresponding *Feature id* column.

   > **Truco**
   >
   > **Use the first column to store «primary keys» for views**
   >
   > Since PostgreSQL views don't support primary keys, a unique attribute or combination of attributes should always be selected. To help users to speed-up workflows, QGIS automatically selects the first attribute in the view. Therefore, users can define their views in a way that a unique column is in the first position of the view's definition. In this way, the view will be loaded with no extra interaction and the warning icon will never appear.

8. Si corresponde, use el botón *Establecer filtro* (o haga doble clic en la capa) para iniciar el diálogo: guilabel:` Generador de consultas` (consulte la sección:ref:*vector_query_builder*) y defina qué características cargar desde el capa seleccionada. La expresión de filtro aparece en la columna `sql`. Esta restricción se puede eliminar o editar en la *Propiedades de capa -> General -> Filtro de características del proveedor*.
9. La casilla de verificación en la columna `Seleccionar en id` que está activada por defecto obtiene los identificadores de funciones sin los atributos y generalmente acelera la carga de datos.
10. Haga clic en el botón *Añadir* para agregar la capa al mapa.

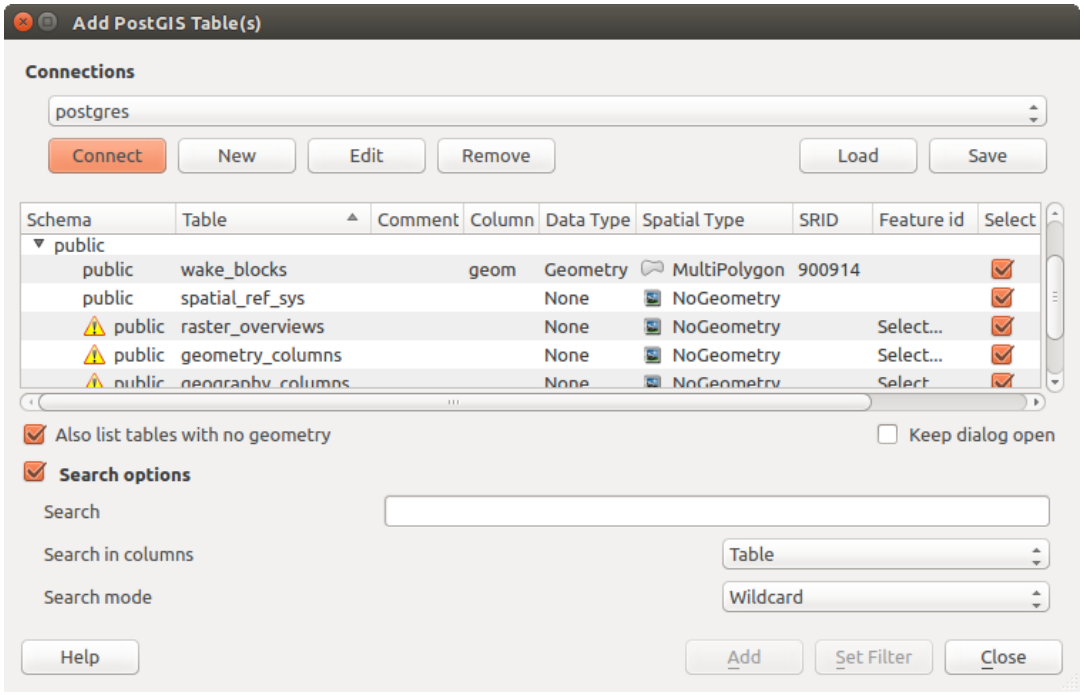


Figura 11.16: Add PostgreSQL Table(s) Dialog

Truco

** Utilice el panel del navegador para acelerar la carga de la tabla(s) de la base de datos **

Agregar tablas de base de datos del *Administrador de fuentes de datos* a veces puede llevar mucho tiempo ya que QGIS obtiene estadísticas y propiedades (por ejemplo, tipo y campo de geometría, CRS, número de características) para cada tabla de antemano. Para evitar esto, una vez *la conexión está establecida*, es mejor usar *Panel Navegador* or the *Administrador de BBDD* para arrastrar y soltar las tablas de la base de datos en el lienzo del mapa.

11.1.4 Panel de búsqueda de metadatos de capas

De forma predeterminada, QGIS puede recuperar metadatos de capas de las conexiones o proveedores de datos que permiten el almacenamiento de metadatos (más detalles sobre *guardar metadatos en la base de datos*). El panel *Búsqueda de metadatos* permite examinar las capas por sus metadatos y añadirlas al proyecto (ya sea con un doble clic o con el botón *Añadir*). La lista puede filtrarse:

- por texto, observando un conjunto de propiedades de metadatos (identificador, título, resumen)
- por extensión espacial, utilizando la actual *extensión del proyecto* o la extensión del lienzo del mapa
- por el tipo de capa (geometría)

Nota

Las fuentes de metadatos se implementan a través de un sistema de proveedores de metadatos por capas que puede ampliarse mediante complementos.

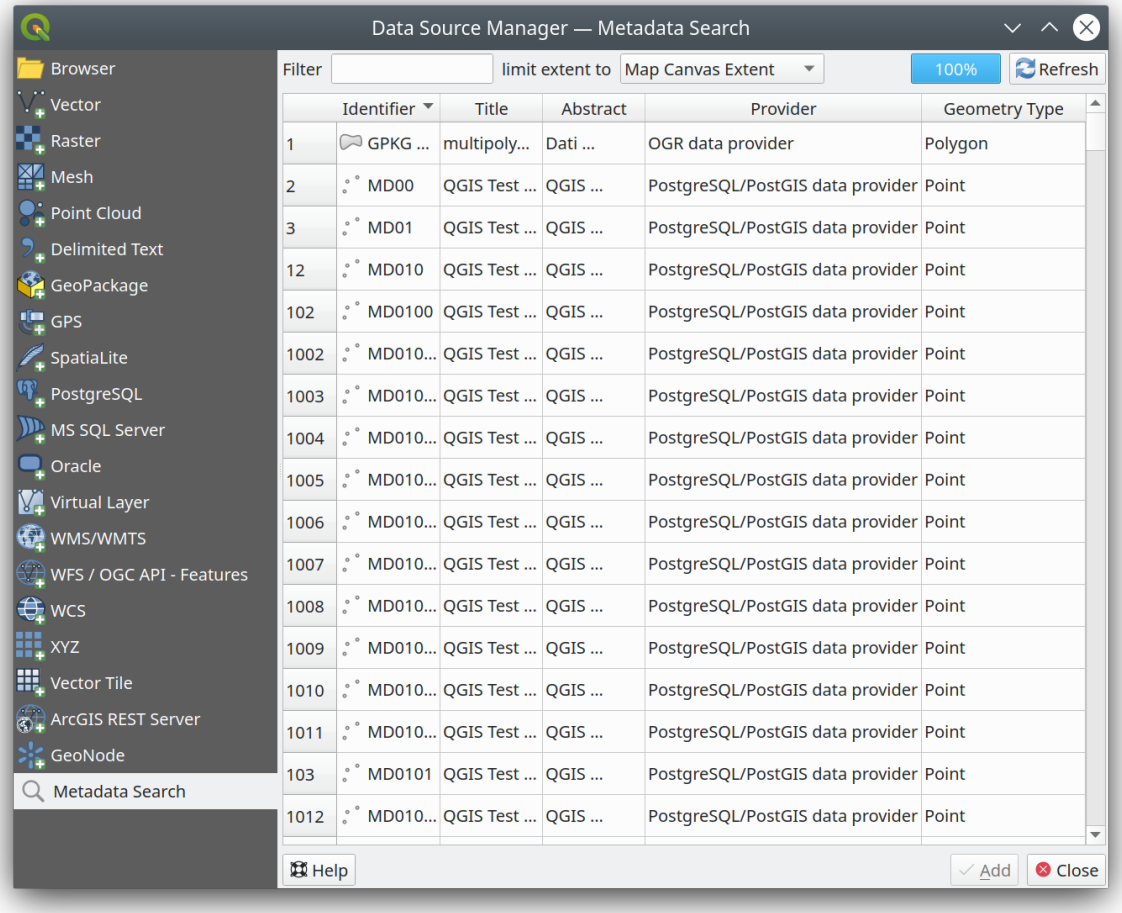


Figura 11.17: Panel de búsqueda de metadatos de capas

11.1.5 Formato personalizados QGIS

QGIS propone dos formatos personalizados:

- Capa temporal temporal: una capa de memoria que está vinculada al proyecto (consulte *Crear una nueva capa borrador temporal* para obtener más información)
- Capas virtuales: una capa resultante de una consulta en otra capa (s) (ver *Creando capas virtuales* para más información)

11.1.6 QLR - Archivo de Definición de Capa QGIS

Las definiciones de capa se pueden guardar como *Layer Definition File* (QLR - `.qlr`) usando :menuselection:`Exportar -> Guardar como archivo de definición de capa…`en el menú contextual de la capa.

El formato QLR hace posible compartir capas QGIS «completas» con otros usuarios de QGIS. Los archivos QLR contienen enlaces a las fuentes de datos y toda la información de estilo QGIS necesaria para diseñar la capa.

Los archivos QLR se muestran en el Panel del navegador y se pueden usar para agregar capas (con sus estilos guardados) al Panel de capas. También puede arrastrar y soltar archivos QLR desde el administrador de archivos del sistema al lienzo del mapa.

Las acciones disponibles para los archivos QLR en el panel del navegador son:

- *Exportar capa* ► *A archivo*

- *Añadir capa a proyecto*
- *Propiedades de Capa*

11.1.7 Conectar con el servicios web

Con QGIS puede acceder a diferentes tipos de servicios web OGC (WM(T)S, WFS(-T), WCS, CSW,…). Gracias a QGIS Server, también puede publicar dichos servicios. QGIS-Server-manual contiene descripciones de estas capacidades.

Usando de Servicios de Teselas Vectoriales

Los servicios de Vector Tile se pueden agregar a través de la pestaña  *Vector Tiles* del diálogo *Data Source Manager* o el menú contextual de la entrada *Vector Tiles* en el panel *Browser*. Los servicios pueden ser una *New Generic Connection…* o una *New ArcGIS Vector Tile Service Connection…*.

Configura un servicio agregando:

- un *Nombre*
- a *Style URL*: una URL a una configuración de estilo MapBox GL JSON. Si se proporciona, ese estilo se aplicará siempre que las capas de la conexión se agreguen a QGIS. En el caso de las conexiones del servicio de teselas vectoriales de Arcgis, la URL anula la configuración de estilo predeterminada especificada en la configuración del servidor.

  Puede cargar teselas vectoriales directamente desde un URL *Estilo*. La fuente de datos se analiza automáticamente a partir del estilo, y se admiten URL con múltiples fuentes. Esto hace que *Source URL* sea opcional.
- la *URL de origen*: del tipo `http://example.com/{z}/{x}/{y}.pbf` para servicios genéricos y `http://example.com/arcgis/rest/services/Layer/VectorTileServer` para servicios basados en ArcGIS. El servicio debe proporcionar mosaicos en formato `.pbf`.
- la  *Min. Zoom Level* y la  *Max. Zoom Level*. Los mosaicos vectoriales tienen una estructura piramidal. Al utilizar estas opciones, tiene la oportunidad de generar capas individualmente a partir de la pirámide de mosaicos. Estas capas luego se utilizarán para renderizar el Vector Tile en QGIS.

  Para la proyección de Mercator (utilizada por OpenStreetMap Vector Tiles), el nivel de zoom 0 representa el mundo entero a una escala de 1:500.000.000. El nivel de zoom 14 representa la escala 1:35.000.
- the *authentication* configuration if necessary
- un *Referente*

Figura 11.18 muestra el diálogo con la configuración del servicio Teselas Vectoriales.

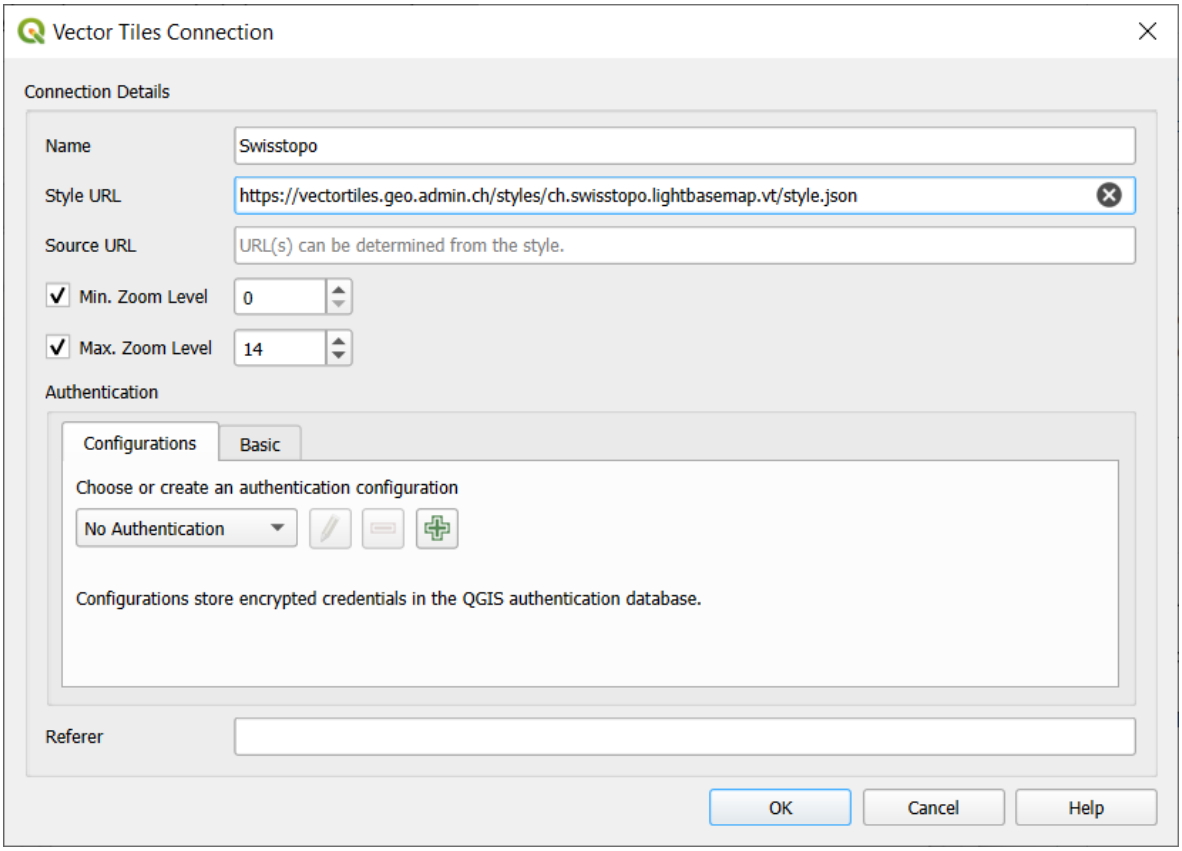


Figura 11.18: Teselas vectoriales - Configuración del servicio

Las configuraciones se pueden guardar en el archivo `.XML` (*Save Connections*) a través de la entrada *Vector Tiles* en el cuadro de diálogo *Data Source Manager* o su menú contextual en el panel *Browser*. Asimismo, se pueden agregar desde un archivo (*Load Connections*).

Una vez que se establece una conexión a un servicio de teselas vectoriales, es posible:

- *Edit* la configuración de conexión de teselas vectoriales
- *Remove* la conexión
- Desde el panel *Browser* haga clic derecho sobre la entrada y también podrá:
  - *Añadir capa al proyecto*: un doble clic también añade la capa
  - Vea *Propiedades de la capa* … y obtenga acceso a los metadatos y una vista previa de los datos proporcionados por el servicio. Hay más configuraciones disponibles cuando la capa se ha cargado en el proyecto.

Usando servicios de Teselas XYZ

Los servicios de teselas XYZ pueden añadirse a través de la pestaña *XYZ* del cuadro de diálogo *Administrador de fuentes de datos* o del menú contextual de la entrada *Mosaicos XYZ* del panel *Navegador*. Por omisión, QGIS proporciona algunos servicios teselas XYZ listos para usar:

- *Mapzen Global Terrain*, permitiendo un acceso inmediato a la fuente global DEM para los proyectos. Más detalles y recursos en https://registry.opendata.aws/terrain-tiles/
- *Mapzen Global Terrain*, que permite el acceso inmediato a la fuente global DEM para proyectos. Más detalles y recursos en https://registry.opendata.aws/terrain-tiles/

Para añadir un nuevo servicio, pulse *Nuevo* (respectivamente *Nueva conexión* del panel Navegador) y proporcione:

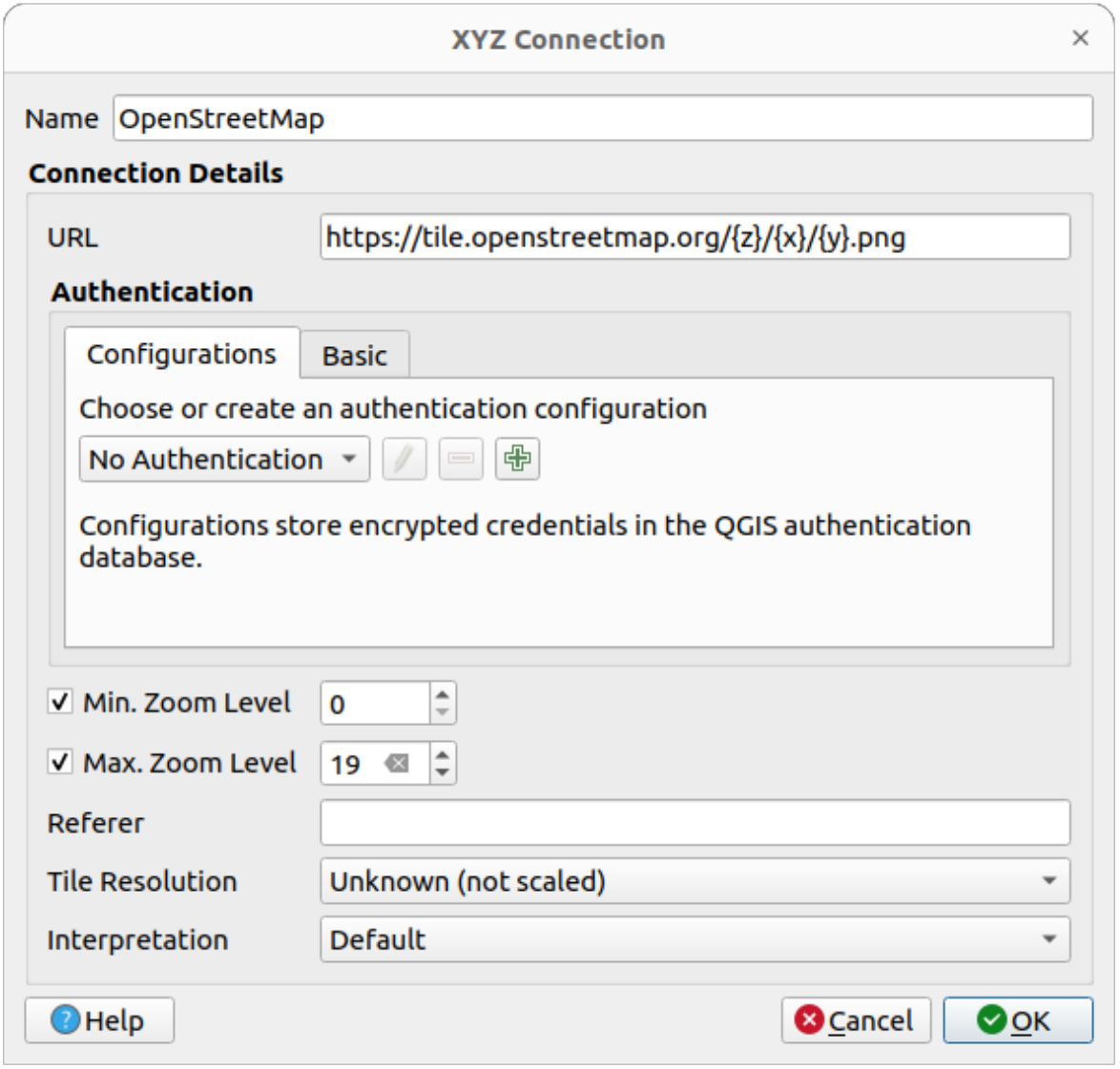


Figura 11.19: XYZ Teselas - Configuración de OpenStreetMap

- un *Nombre*
- la *URL*, puede añadir `http://ejemplo.com/{z}/{x}/{y}.png` o `file:///local_path/{z}/{x}/{y}.png`.
- the *authentication* configuration if necessary
- el *Min. Zoom level* y *Max. Zoom level*
- un *Referente*
- la *Tile Resolution*: los valores posibles son *Unknown (not scaled)*, *Standard (256x256 / 96DPI)* y *High (512x512 / 192DPI)*
- *Interpretación*: convierte conjuntos de datos ráster WMTS/XYZ en una capa ráster de tipo flotante de banda única siguiendo un esquema de codificación predefinido. Los esquemas admitidos son *Por defecto* (no se realiza conversión), *MapTiler Terrain RGB* y *Terrarium Terrain RGB*. El conversor seleccionado traducirá los valores fuente RGB a valores flotantes para cada píxel. Una vez cargada, la capa se presentará como una capa ráster de punto flotante de banda única, lista para ser estilizada utilizando los renderizadores *raster habituales de QGIS.*.

Pulse *Aceptar* para establecer la conexión. A continuación, será posible:

- *Añadir* la nueva capa al proyecto; se carga con el nombre dado en la configuración.
- *Edit* la configuración de conexión XYZ
- *Remove* la conexión
- Desde el panel *Browser* haga clic derecho sobre la entrada y también podrá:
  - *Exportar capa…* ► *a Archivo*, *guardándolo como un ráster*
  - *Añadir capa al proyecto*: un doble clic también añade la capa
  - Vea *Propiedades de la capa …* y obtenga acceso a los metadatos y una vista previa de los datos proporcionados por el servicio. Hay más configuraciones disponibles cuando la capa se ha cargado en el proyecto.

Las configuraciones pueden guardarse en un archivo `.XML` (*Guardar Conexiones*) a través de la entrada *XYZ* del diálogo *Administrador de Fuentes de Datos* o de su menú contextual en el panel *Navegador*. Del mismo modo, pueden añadirse desde un archivo (*Cargar conexiones*).

El archivo XML para OpenStreetMap se ve así:

```
<!DOCTYPE connections>
<qgsXYZTilesConnections version="1.0">
  <xyztiles url="https://tile.openstreetmap.org/{z}/{x}/{y}.png"
   zmin="0" zmax="19" tilePixelRatio="0" password="" name="OpenStreetMap"
   username="" authcfg="" referer=""/>
</qgsXYZTilesConnections>
```

Truco

Carga de teselas XYZ sin crear una conexión

También es posible añadir teselas XYZ a un proyecto sin almacenar necesariamente su configuración de conexión en su perfil de usuario (por ejemplo, para un conjunto de datos que pueda necesitar una sola vez). En la pestaña *Administrador de fuentes de datos* ► *XYZ*, edite las propiedades del grupo *Detalles de conexión*. El campo *Name* anterior debería convertirse en `Custom`. Pulse *Añadir* para cargar la capa en el proyecto. Se llamará por Predeterminado `Capa XYZ`.

Ejemplos de servicios de XYZ Teselas:

- OpenStreetMap Monocromo: *URL*: `http://tiles.wmflabs.org/bw-mapnik/{z}/{x}/{y}.png`, *Min. Zoom Level*: 0, *Max. Zoom Level*: 19.
- Google Maps: *URL*: `https://mt1.google.com/vt/lyrs=m&x={x}&y={y}&z={z}`, *Min. Zoom Level*: 0, *Max. Zoom Level*: 19.
- Open Weather Mapa Temperatura: *URL*: `http://tile.openweathermap.org/map/temp_new/{z}/{x}/{y}.png?appid={api_key}` *Min. Zoom Level*: 0, *Max. Zoom Level*: 19.

Uso de servidores REST de ArcGIS

An ArcGIS REST Server can host many different types of web services (feature service, map service, image service, …). ArcGIS REST Servers can be added via the  *ArcGIS REST Server* tab of the *Data Source Manager* dialog or the contextual menu of the *ArcGIS REST Servers* entry in the *Browser* panel:

1. Press *New* (respectively *New Connection*) and provide:
   - a *Name*: A name for the connection.
   - the *URL*: Main address of the ArcGIS REST Server.
   - a *Prefix*: Used to specify the proxy prefix in the URL, which is necessary for some ArcGIS servers that use web proxy prefixes.

- a *Community endpoint URL*: Endpoint URL for ArcGIS Online or Portal for ArcGIS, used to access content groups.
- a *Content endpoint URL*: Endpoint URL for the content service. This is used to access items in ArcGIS Online or Portal for ArcGIS.
- the *authentication* credentials if necessary.
- a *Referer*: The referer URL to be sent in the HTTP headers when making requests to the server. This may be required by some servers for authentication purposes.

> **Nota**
>
> Las conexiones del servicio de entidades de ArcGIS que tienen establecidas las URL de punto final del portal correspondiente pueden explorarse mediante grupos de contenido en el panel del navegador.
>
> Si una conexión tiene establecidos los puntos finales del Portal, al expandir la conexión en el navegador se mostrará una carpeta de “Groups” y “Services”, en lugar de la lista completa de servicios que se muestra normalmente. Al expandir la carpeta de grupos, se mostrará una lista de todos los grupos de contenido de los que el usuario es miembro, cada uno de los cuales se puede expandir para mostrar los elementos de servicio que pertenecen a ese grupo.

2. Then press *OK* to validate the configuration settings. These configurations can be saved to `.XML` file (*Save*). Likewise, they can be added from a file (*Load*).
3. Once a connection to an ArcGIS REST Server is set, it is possible to:
   - *Edit* la configuración de conexión de ArcGIS REST Server
   - *Remove* la conexión
   - *Refresh* la conexión
4. Press *Connect* to request the server and display its contents. They are organized in a tree structure whose nodes depend on the connection’s endpoint.

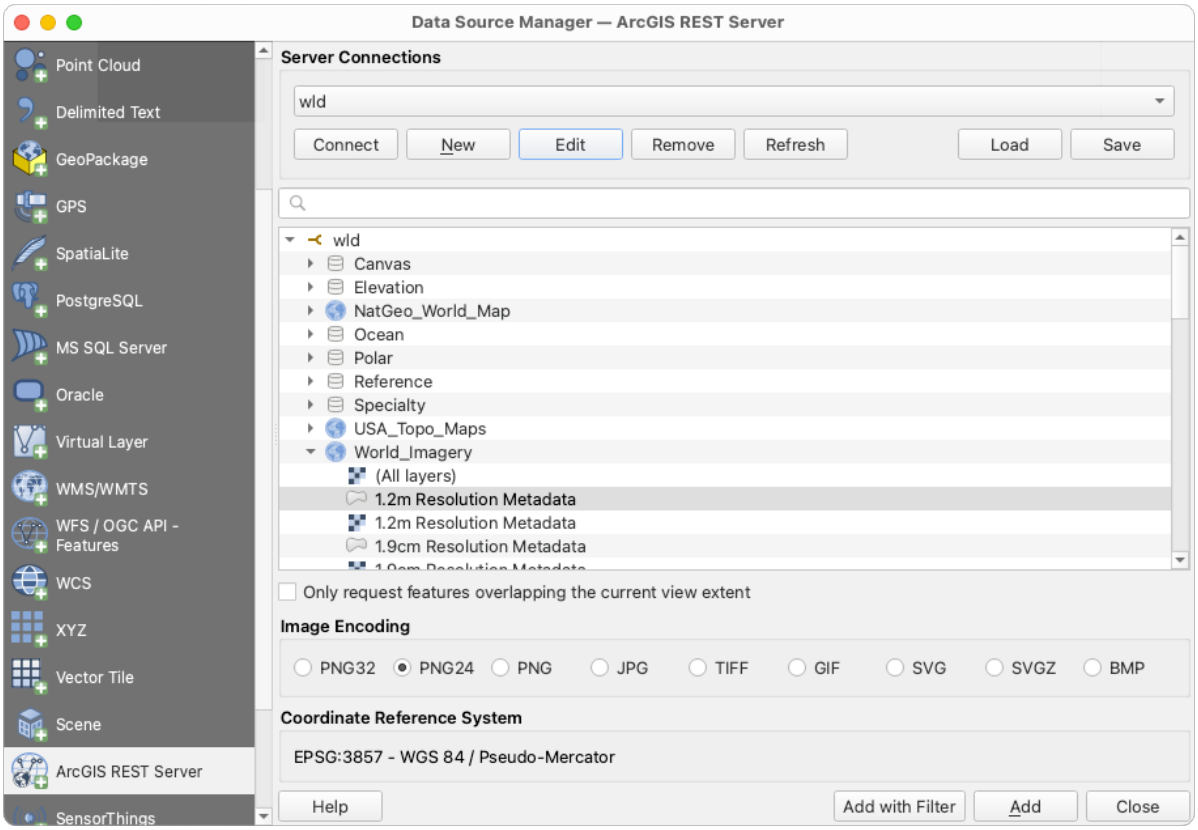


Figura 11.20: Data Source Manager - ArcGIS REST Server

5. Expand the tree to find and select the layers of interest. Their *Coordinate Reference System* is displayed at the bottom of the dialog.
6. For raster-based layers, you can select the *Image encoding* to use among a number of image formats advertized by the target service (e.g., `PNG`, `JPG`, `GIF`, `SVG`, `SVGZ`, ...).
7. To add the selected layers to the map canvas, press *Add* button.

   Because layers can sometimes load and render slowly on the client side, applying a filter to restrict the features retrieved from the service can significantly improve performance, since only the filtered features are requested from the server. This can be done by:

   - checking *Only request features overlapping the current view extent*
   - for vector layers (feature service), pressing *Add with filter* to apply attribute-based filters to the layer with *El constructor de cadena de Expresión* functions.

     In the *Browser* panel, right-click the ArcGIS REST Server layer and select *Add Filtered Layer to Project* will also open the builder dialog.

Most of the above tools to connect and access layers on an ArcGIS REST Server are also available from the *Browser* panel, within the contextual menu of the target node item. Depending on the node, you can also:

- *View Service Info* which will open the default web browser and display information on the requested service.
- View the *Layer Properties...* and get access to the metadata and a preview of the data provided by the service. More settings are available when the layer has been loaded into the project.

Utilización de servicios de escena 3D en mosaico

QGIS admite varios formatos de conjuntos de datos 3D en mosaico, agrupados como «escenas en mosaico». Entre ellos se incluyen teselas Cesium 3D y teselas Quantized Mesh.

Para cargar un conjunto de datos de escena en mosaico en QGIS, utilice la pestaña *Escena* del cuadro de diálogo *Administrador de fuentes de datos*.

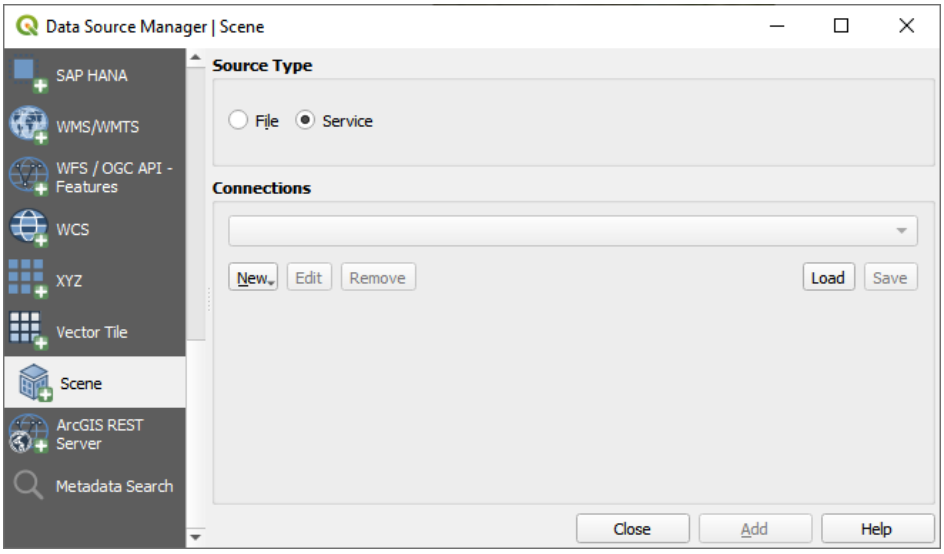


Figura 11.21: Administrador de fuentes de datos - Escena

Cree una conexión haciendo clic en *Nueva*. Puede añadir una *Nueva conexión de Cesium 3D Tiles* o una *Nueva conexión Quantized Mesh*.

Elija un *Nombre* y establezca el *URL* en la URL de un archivo JSON de descripción de capa.

La URL puede ser remota (por ejemplo, `http://example.com/tileset.json`) o local (por ejemplo, `file:///path/to/tiles/tileset.json`).

The *authentication* configuration if necessary.

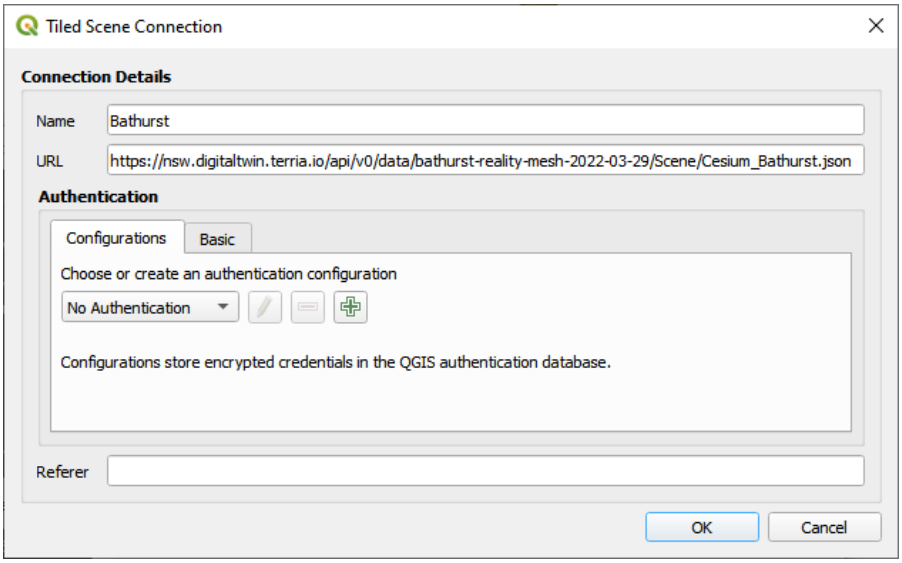


Figura 11.22: Conexión de escenas en mosaico

También puede añadir el servicio desde *Panel de Navegador*.

Una vez creada la nueva conexión, podrá *Añadir* la nueva capa a su mapa.

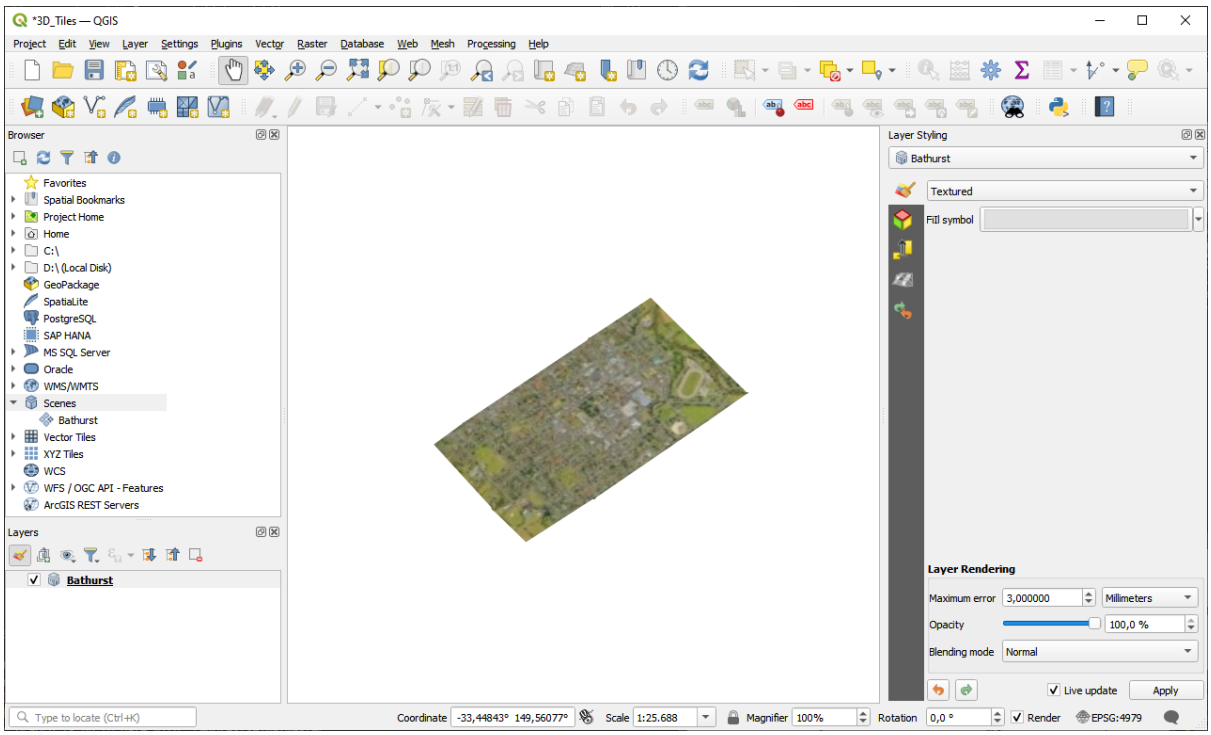


Figura 11.23: Capa de teselas 3D - Texturizado

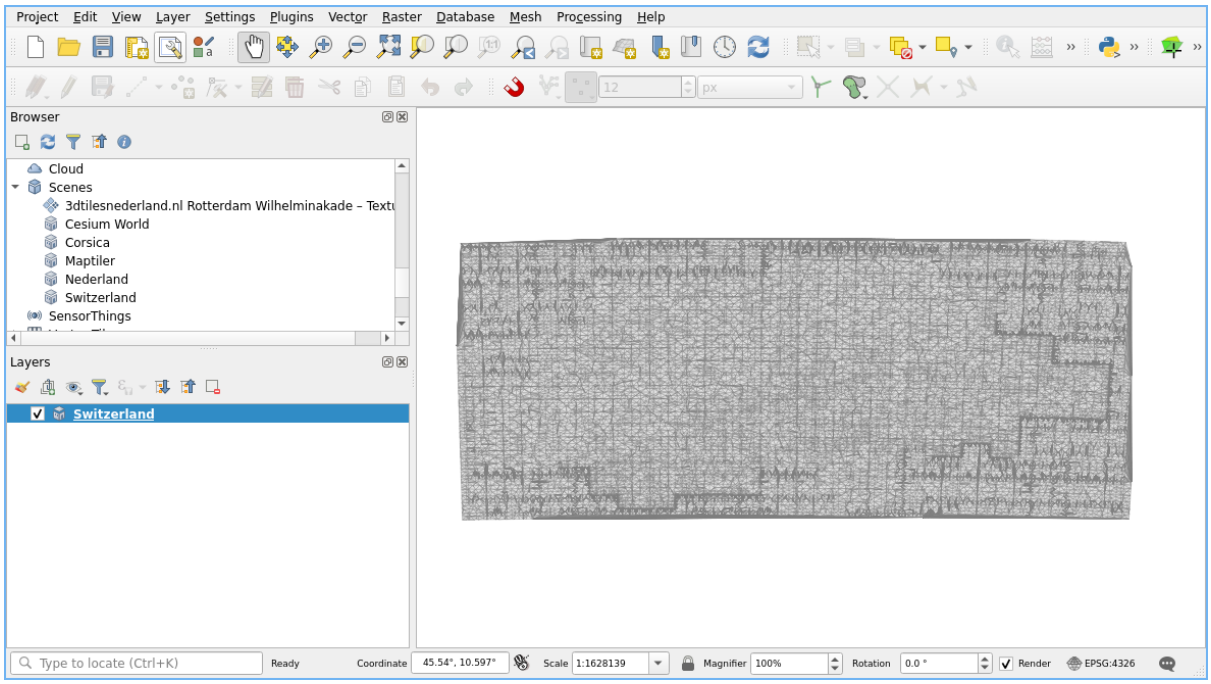


Figura 11.24: Capa Quantized Mesh

Uso de conexiones a la nube

QGIS admite conexiones a servicios en la nube como Alibaba Cloud OSS, Amazon S3, Google Cloud Storage, Microsoft Azure Blob Storage, Microsoft Azure Data Lake Storage y OpenStack Swift Object Storage. Puede cargar datos vectoriales y ráster de estos servicios en QGIS. Configure una nueva conexión *Nube* en el panel *Navegador* haciendo clic con el botón derecho en la entrada *Nube* y seleccionando *Nueva conexión*. Verá una lista desplegable de los servicios en la nube disponibles. Seleccione el servicio al que desea conectarse y rellene los campos requeridos:

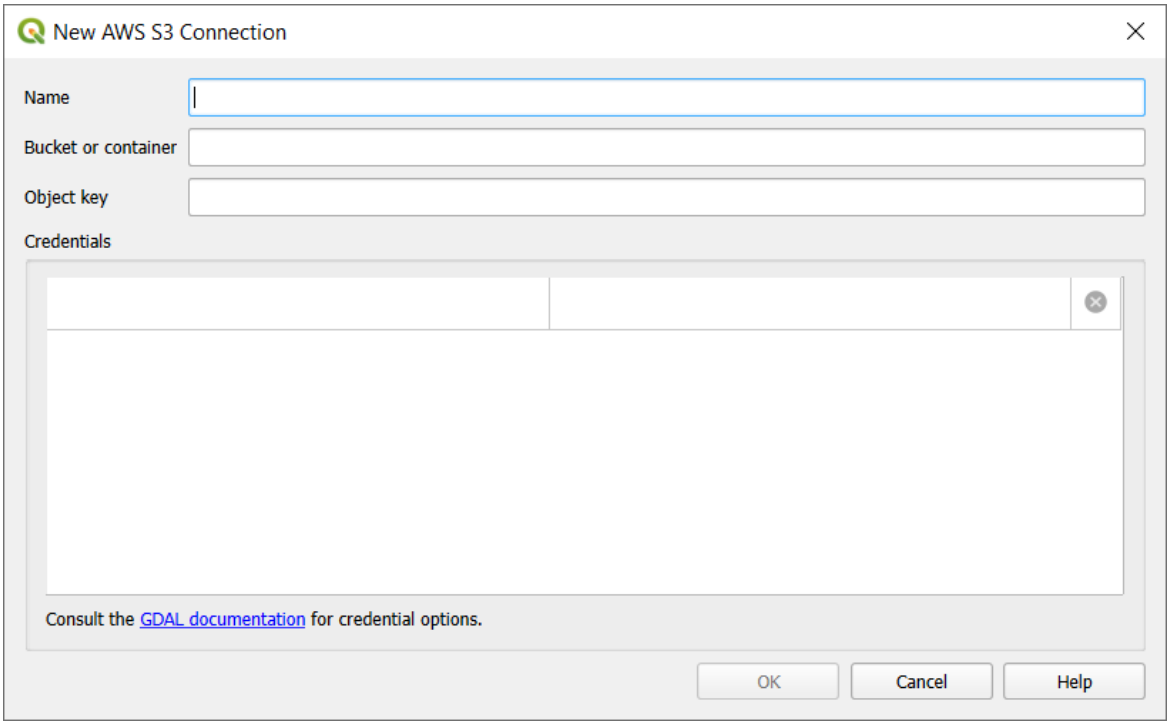


Figura 11.25: Diálogo de conexión a la nube

- *Nombre*: Un nombre para la conexión.
- *Cubo o contenedor*: El nombre del cubo o contenedor en el servicio en la nube.
- *Clave del objeto* (opcional): La clave del objeto en el cubo o contenedor.
- *Credenciales*: Las credenciales para acceder al servicio en la nube.

También puede elegir entre *Guardar conexión* en un archivo XML o *Cargar conexión* desde un archivo XML.

11.2 Creando capas

Las capas pueden ser creadas de muchas formas, incluyendo:

- capas vacías desde borrador
- capas desde capas existentes
- capas desde el portapapeles
- capas como resultado de una consultas SQL basada en una o mas capas (*virtual layers*)

QGIS también proporciona herramientas para importar/exportar desde/a diferentes formatos.

11.2.1 Creando nuevas capas vectoriales

QGIS le permite crear nuevas capas en diferentes formatos. Proporciona herramientas para crear GeoPackage, Shapefile, SpatiaLite, formato GPX y capas temporales borrador (también conocidas como capas de memoria). La creación de una *nueva capa GRASS* es compatible con el complemento GRASS.

Creando una nueva capa de GeoPackage

Para crear una nueva capa de GeoPackage, pulse el botón *Nueva Capa de GeoPackage…* en el menú *Capa* ► *Crear Capa* ► o desde la barra de herramientas *Administrador de Fuentes de Datos*. También puede crear una nueva capa de GeoPackage a través del *Panel del Navegador* seleccionando el botón *Crear Base de Datos y Capa…*. Aparecerá el cuadro de diálogo *Nueva capa de GeoPackage* como se muestra en Figura 11.26.

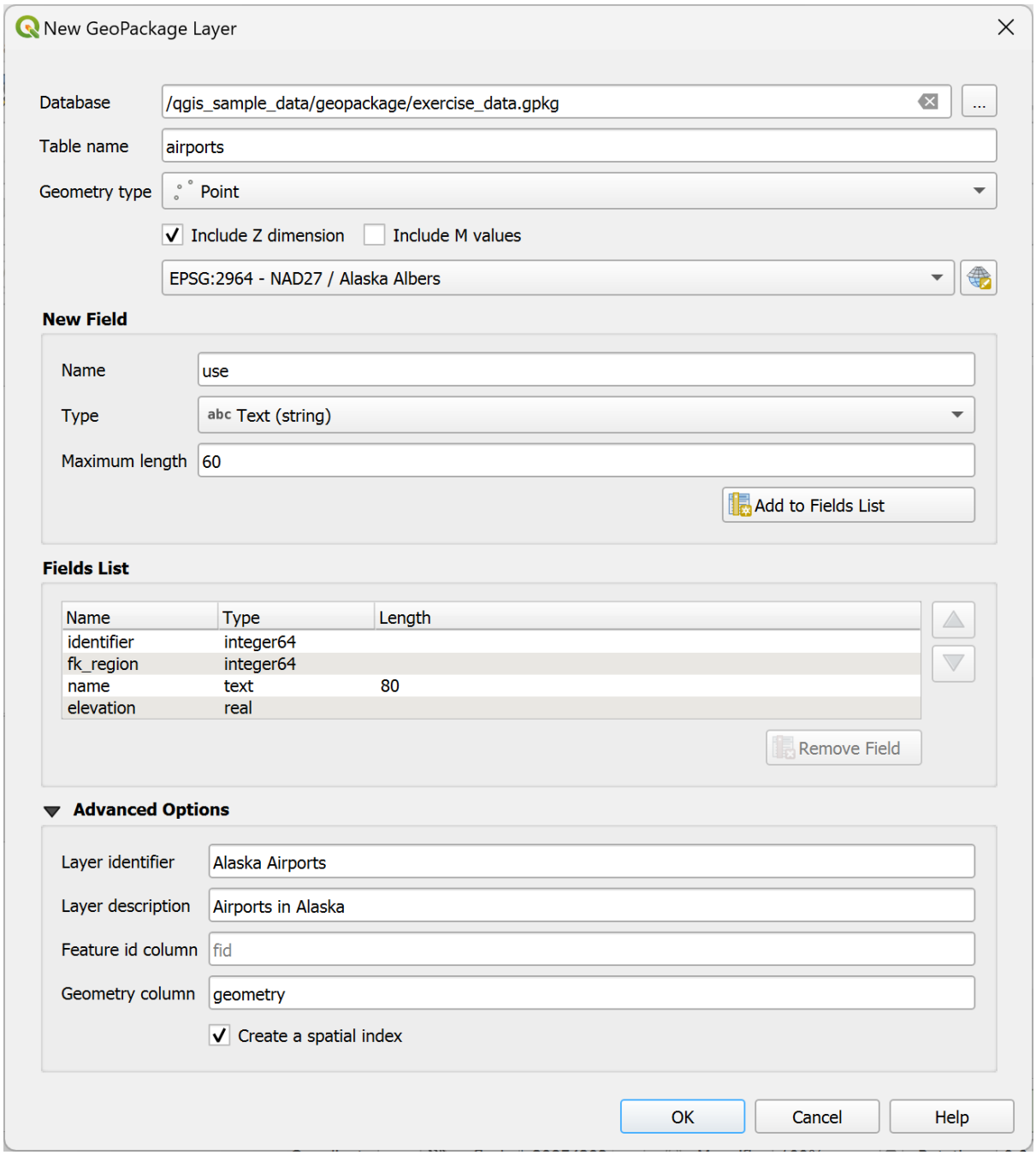


Figura 11.26: Crear un nuevo diálogo de capa GeoPackage

1. El primer paso es indicar la ubicación del archivo de la base de datos. Esto se puede hacer presionando el botón … a la derecha del campo *Base de datos* y seleccione un archivo GeoPackage existente o cree uno nuevo. QGIS agregará automáticamente la extensión correcta al nombre que proporcione.
2. Dá un nombre a la nueva capa/tabla (*Nombre Tabla*)
3. Defina el *Tipo de geometría*. Si no es una capa sin geometría, puede especificar si debe *Incluir dimensión Z* y/o *Incluir valores M*.
4. Especificar el sistema de coordenadas de referencia usando el botón 

Para añadir campos a una capa que estás creando:

1. Introduce el *Nombre* del campo
2. Seleccione los datos *Tipo*. Los tipos admitidos son *Datos de texto*, :guilabel:` Número entero` (tanto entero como entero64), *Número decimal*, :guilabel:` Fecha` y *Fecha y hora*, *Binario (BLOB)* y *Booleano*.
3. Dependiendo del formato de datos seleccionado *longitud Máxima* de valores.
4. Click en el botón  *Añadir a Lista de campos*
5. Repite los pasos anteriores para cada campo que necesites añadir
6. Luego puede cambiar el orden de los camos usando los botones  Mover arriba y  Mover abajo
7. Una vez contento con los atributos, click *Aceptar*. QGIS añadirá la nueva capa a la leyenda, y puede editarla como se describe en la sección *Digitalizando una capa existente*.

Por defecto, al crear una capa de GeoPackage, QGIS genera una *columna de identificación de entidad* llamada `fid` que actúa como la clave principal de la capa. El nombre se puede cambiar. El campo de geometría, si está disponible, se denomina `geometría` y puede elegir *Crear un índice espacial* en él. Estas opciones se pueden encontrar en:guilabel:*Opciones avanzadas* junto con *Identificador de capa* (nombre corto legible por humanos de la capa) y *Descripción de capa*.

Se puede administrar más capas de GeoPackage con el *DB Manager*.

Crear una nueva capa de archivo shape

Para crear una nueva capa de formato ESRI Shapefile, presione el botón  *Nueva Capa Shapefile…* en *Capa* -> *Crear Capa* -> o desde la barra de herramientas *Administrador de Fuentes de Datos*. Se mostrará el diálogo *Nueva Capa Shapefile* como se muestra en Figura 11.27.

1. Proporcione una ruta y un nombre de archivo usando el botón … junto a *Nombre de archivo*. QGIS agregará automáticamente la extensión correcta al nombre que proporcione.
2. A continuación, indica el *codificación de archivo* de los datos
3. Elija el *Tipo de geometría* de la capa: Sin geometría(resultando en un archivo de formato `.DBF`), punto, multipunto, línea o polígono
4. Especifique si la geometría debe tener dimensiones adicionales: *Ninguno*, *Z (+ M valores) `o :guilabel:`M valores*
5. Especificar el sistema de coordenadas de referencia usando el botón 

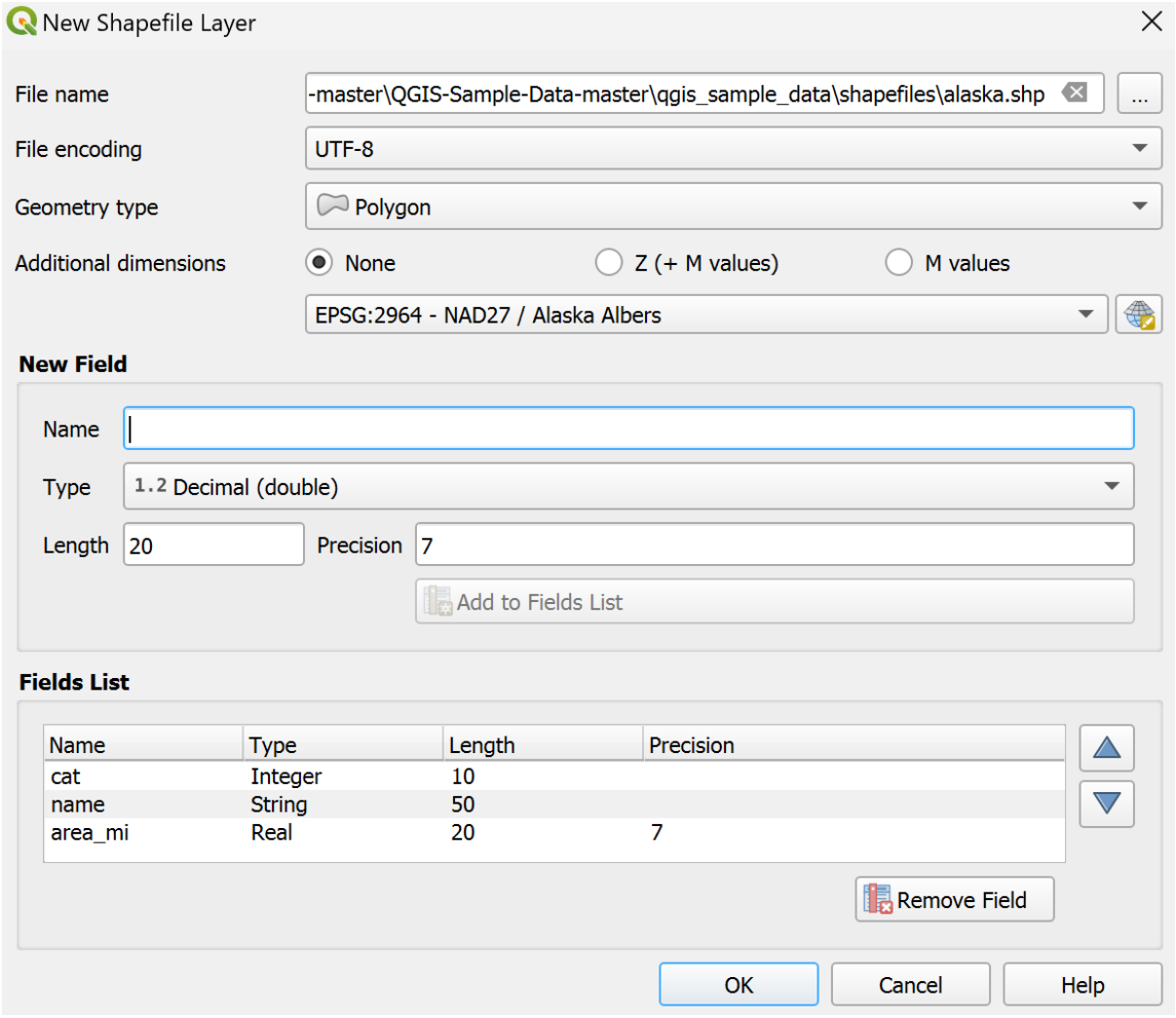


Figura 11.27: Diálogo Creando una nueva capa de archivo shape

Para añadir campos a una capa que estás creando:

1. Introduce el *Nombre* del campo
2. Seleccione los datos *Tipo*. Solo se admiten los atributos de los tipos de datos *Número decimal*, *Número entero*, *Datos de texto*, *Fecha* y *Booleano*.
3. En función del formato de datos, introduce la *Longitud* y *Precisión*.
4. Click en el botón *Añadir a Lista de campos*
5. Repite los pasos anteriores para cada campo que necesites añadir
6. Luego puede cambiar el orden de los camos usando los botones Mover arriba y Mover abajo
7. Una vez contento con los atributos, click *Aceptar*. QGIS añadirá la nueva capa a la leyenda, y puede editarla como se describe en la sección *Digitalizando una capa existente*.

Por defecto, una primera columna de enteros `id` se añade pero puede ser borrada.

Crear una nueva capa SpatiaLite

Para crear una nueva capa SpatiaLite, presione el botón  *Nueva Capa SpatiaLite…* en el *Capa* -> *Crear Capa* -> o desde la barra de herramientas *Administrador de Fuentes de Datos*. *Nueva Capa SpatiaLite* se mostrará como se muestra en Figura 11.28.

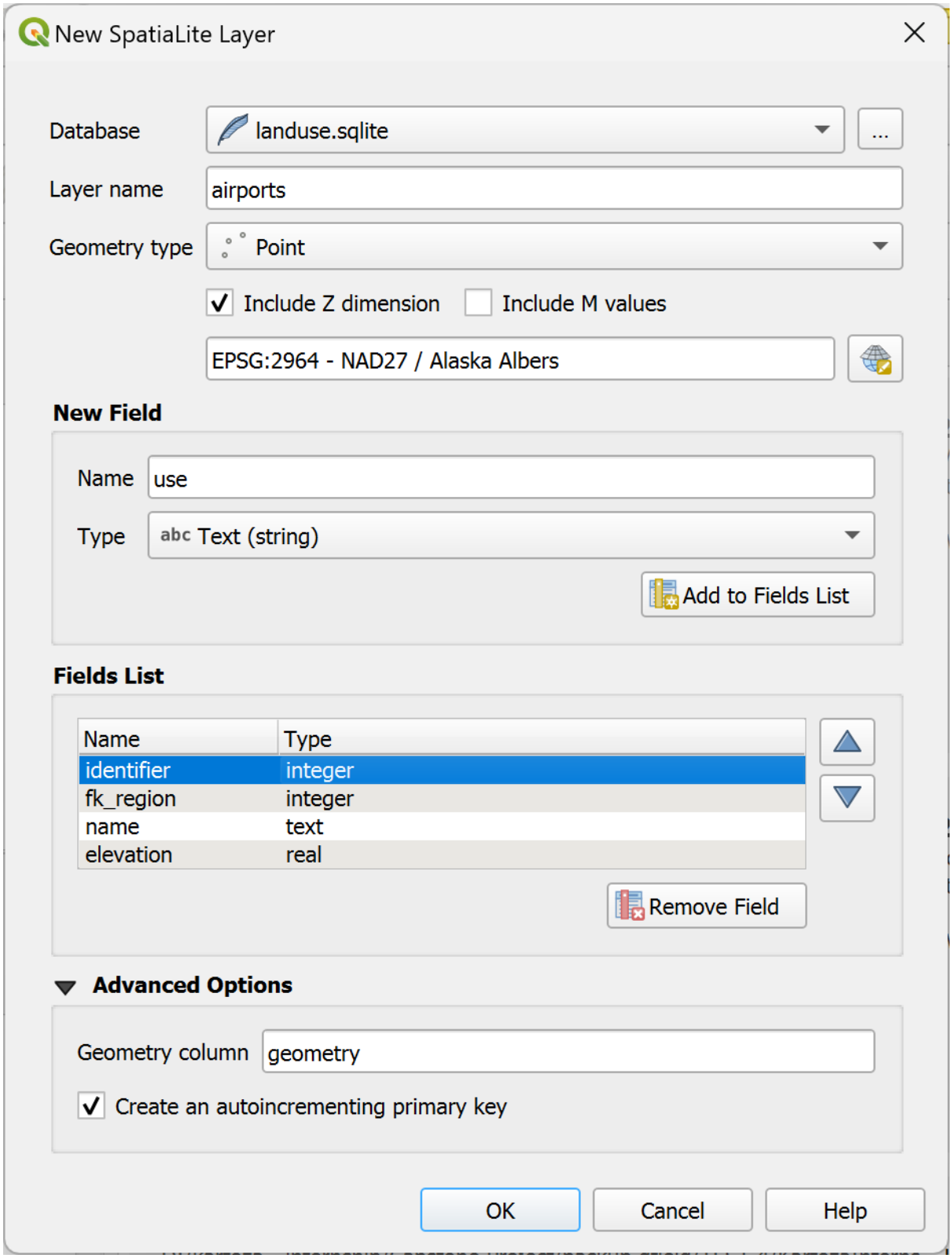


Figura 11.28: Diálogo de Crear una nueva capa SpatiaLite

1. El primer paso es indicar la ubicación del archivo de la base de datos. Esto se puede hacer presionando el botón … a la derecha del campo *Base de datos* y seleccione un archivo SpatiaLite existente o cree uno nuevo. QGIS agregará automáticamente la extensión correcta al nombre que proporcione.
2. Proporciona un nombre (*Nombre de capa*) para la nueva capa

3. Defina el *Tipo de geometría*. Si no es una capa sin geometría, puede especificar si debe *Incluir dimensión Z* y/o *Incluir valores M*.
4. Especifica el sistema de coordenadas de referencia usando el botón .

Para añadir campos a una capa que estás creando:

1. Introduce el *Nombre* del campo
2. Selecciona el dato *Type*. Los tipos soportados son *Text data*, *Whole number*, *Decimal number*, *Date* y *Date time*.
3. Click en el botón *Añadir a Lista de campos*
4. Repite los pasos anteriores para cada campo que necesites añadir
5. Luego puede cambiar el orden de los camos usando los botones Mover arriba y Mover abajo
6. Una vez contento con los atributos, click *Aceptar*. QGIS añadirá la nueva capa a la leyenda, y puede editarla como se describe en la sección *Digitalizando una capa existente*.

Si lo desea, puede seleccionar **|casilla de verificación|** *Cree una clave primaria de incremento automático* en la sección *Opciones avanzadas*. También puedes cambiar el nombre de *Columna Geometría* (`geometría` predeterminada).

Administración adicional de capas SpatiaLite se puede hacer con :ref:Àdministrador de DB<dbmanager>`.

Crear una nueva capa de malla

Para crear una nueva capa de malla, presione *New Mesh Layer…* en el menú *Layer* ► *Create Layer* ► o desde la barra de herramientas *Data Source Manager*. El cuadro de diálogo *New Mesh Layer* se mostrará como se muestra en Figura 11.29.

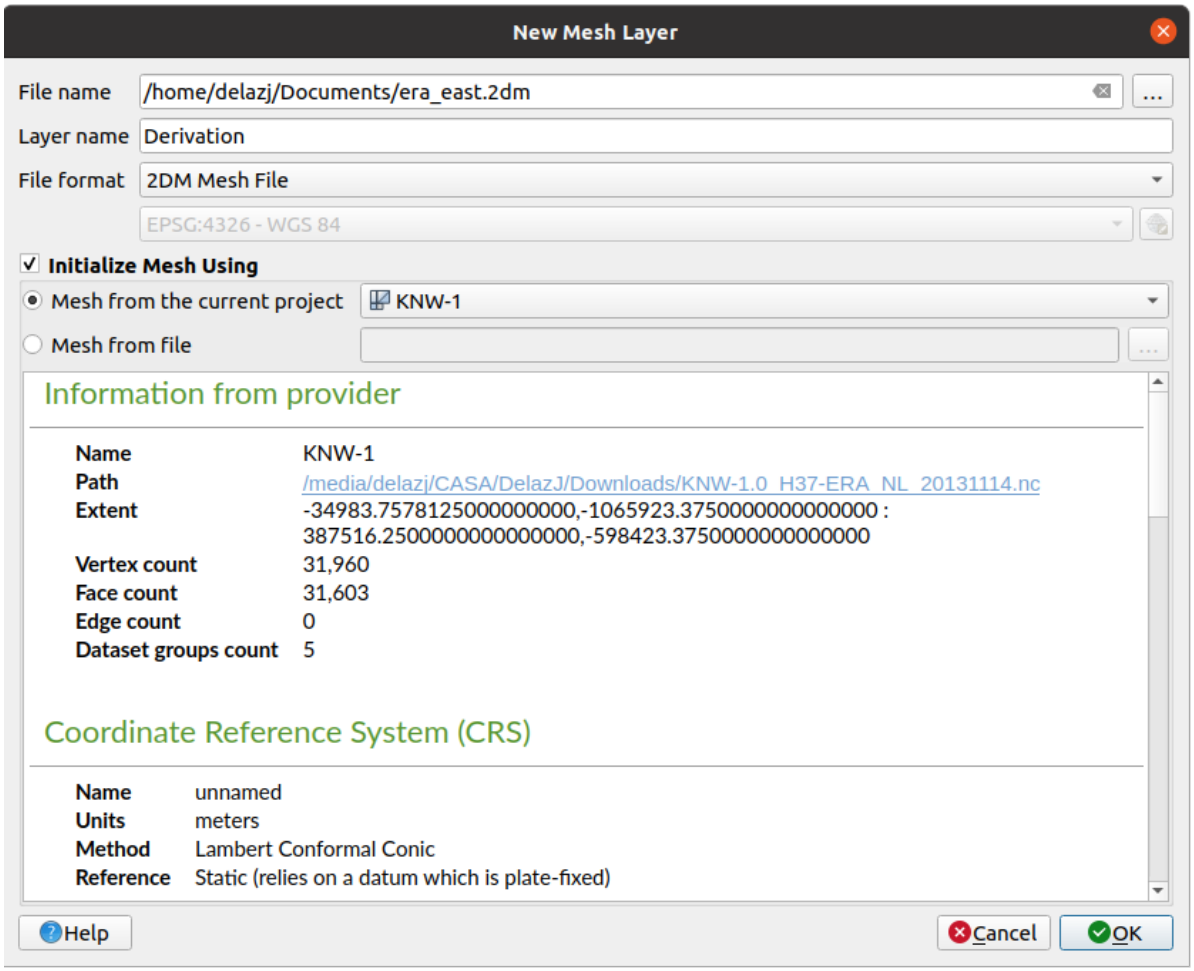


Figura 11.29: Creación de un cuadro de diálogo Nueva capa de malla

1. El primer paso es indicar la ubicación del archivo de malla. Esto se puede hacer presionando el botón … a la derecha del campo *File name* y seleccionar un archivo de malla existente o crear uno nuevo.
2. Proporcione un nombre (*Layer name*), es decir, el nombre con el que se muestra la capa en el panel *Layers*
3. Seleccione el *File format*: los formatos de archivo de malla admitidos actualmente son `2DM Mesh File (*.2dm)`, `Selafin File (*.slf)` y ``UGRID (*.nc)``.
4. Indique el *Coordinate Reference System* para asignar al conjunto de datos
5. Los pasos anteriores generarán una capa vacía a la que luego podrá digitalizar vértices y agregar grupos de conjuntos de datos. Sin embargo, también es posible inicializar la capa con una capa de malla existente, es decir, poblar la nueva capa con vértices o caras de la otra. Para hacerlo:

   1. Marque *Initialize Mesh using*
   2. y seleccione una *Mesh from the current project* o *Mesh from a file*. La información sobre el archivo de malla seleccionado se muestra para su verificación.

   Tenga en cuenta que solo el marco de la capa de malla se transfiere a la nueva capa; sus conjuntos de datos no se copian.

Crear una nueva capa GPX

Para crear un nuevo archivo GPX:

1. Seleccione **:menuseleccion:`Crear capa -->`**  *Nueva capa GPX…* del menú *Capa*.
2. En el diálogo, elija dónde guardar el nuevo archivo, asígnele un nombre y presione *Save*.
3. Se añaden tres nuevas capas a la *Layers Panel*:
   - una capa de puntos para digitalizar ubicaciones (`waypoints`) con campos que almacenan el nombre, la elevación, el comentario, la descripción, la fuente, la URL y el nombre de la URL
   - una capa de línea para digitalizar secuencias de ubicaciones que conforman una ruta planificada (`routes`) con campos que almacenan el nombre, símbolo, número, comentario, descripción, fuente, URL, nombre de URL
   - y una capa de línea para rastrear el movimiento del receptor a lo largo del tiempo (`tracks`) con campos que almacenan el nombre, símbolo, número, comentario, descripción, fuente, URL, nombre de URL.
4. Ahora puede editar cualquiera de ellas como se describe en la sección *Digitalizando una capa existente*.

Crear una nueva capa borrador temporal

Las capas borrador temporales son capas en memoria, lo que significa que no se guardan en el disco y se descartarán cuando se cierre QGIS. Pueden ser útiles para almacenar entidades que necesita temporalmente o como capas intermedias durante las operaciones de geoprocesamiento.

Para crear una nueva capa temporal de Scratch, elija  *Nueva capa temporal temporal…* en la entrada *Capa* -> *Crear capa* -> o en la barra de herramientas *Administrador de Fuentes de Datos*. Se mostrará el cuadro de diálogo *Nueva capa temporal temporal* como se muestra en: numref:*figure_create_temporary*. Luego:

1. Proporciona el **:guilable:`nombre de capa`**
2. Selecciona el *tipo de Geometría*. Aquí puede crear una:
   - tipo de capa `sin geometría`, dotada de una simple tabla,
   - capa `Punto` o `Multipunto`,
   - Capa `Línea/CurvaCompuesta` o `Multilínea/MultiCurva`,
   - Capa `Polígono/PolígonoCurvo` o `MultiPolígono/Multisuperficie`.
3. Para tipos geométricos, especifique las dimensiones del conjunto de datos: marque si debe *Incluir dimensión Z* y/o *Incluir valores M*
4. Especifica el sistema de coordenadas de referencia usando el botón  .
5. Agrega campos a la capa. Tenga en cuenta que, a diferencia de muchos formatos, las capas temporales se pueden crear sin campos. Por tanto, este paso es opcional.
   1. Introduce el *Nombre* del campo
   2. Seleccione el *Tipo* de datos : Son soportados *Texto*, *Número entero*, *Número decimal*, *Booleano*, *Fecha*, *Hora*, *Fecha & Hora* y *Binario (BLOB)*.
   3. Dependiendo del formato de datos seleccionado, ingrese *Longitud* y *Precisión*
   4. Click en el botón  *Añadir a Lista de campos*
   5. Repetir los pasos de arriba para cada campo que necesite agregar
   6. Luego puede cambiar el orden de los camos usando los botones  Mover arriba y  Mover abajo
6. Una vez que esté satisfecho con la configuración, haga click en *Aceptar*. QGIS agregará la nueva capa al panel *Capas*, y puede editarlo como se describe en la sección *Digitalizando una capa existente*.

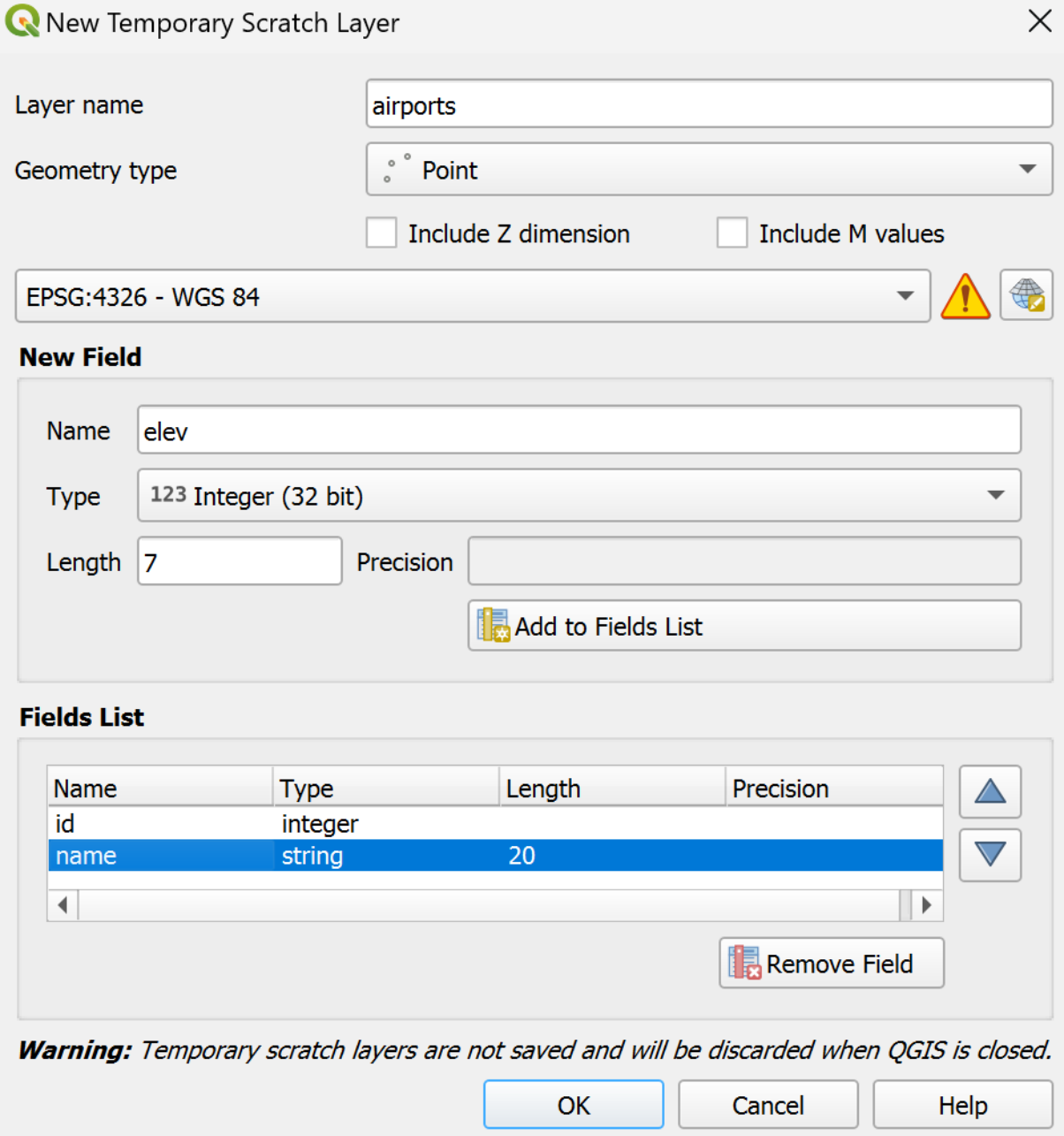


Figura 11.30: Diálogo Crear una nueva capa Borrador Temporal

También puede crear capas temporales precargadas utilizando, por ejemplo, el portapapeles (ver *Creando nuevas capas desde el portapapeles*) o como resultado de un *Algoritmo de procesamiento*.

Truco

Almacenar permanentemente una capa de memoria en disco

Para evitar la pérdida de datos al cerrar un proyecto con capas temporales temporales, puede guardar estas capas en cualquier formato vectorial compatible con QGIS:

- clickando el icono indicador al lado de la capa;
- seleccionando la èntrada *Hacer permanente* en el menú contextual de capa;
- usando la entrada *Exportar* ► desde el menú contextual o desde el menú *Capa* ► *Guardar como…*.

Cada uno de estos comandos abre el diálogo *Guardar capa vectorial como* descrito en la sección *Creando nuevas capas desde una capa existente* y el archivo guardado reemplaza al temporal en el panel *Capas*.

11.2.2 Creando nuevas capas desde una capa existente

Las capas (ráster, vector y nube de puntos) pueden guardarse en un formato diferente y/o reproyectarse a un sistema de referencia de coordenadas (SRC) diferente utilizando el menú **:menuselección:`Capa --> Guardar como...`** o haciendo clic con el botón derecho del ratón sobre la capa en el panel *Capas* y seleccionando:

- *Exportar* ► *Guardar como…* para capas raster y de nubes de puntos
- *Exportar* ► *Guardar entidades como…* o *Exportar* ► *Guardar Entidades seleccionadas como…* para capas vectoriales.
- Drag and drop the layer from the layer tree to the PostgreSQL entry in the *Browser Panel*. Note that you must have a PostgreSQL connection in the *Browser Panel*.

Parámentros comunes

El diálogo *Guardar capa como…* muestra varios parámetros para cambiar el comportamiento al guardar la capa. Entre los parámetros comunes para ráster y vector se encuentran:

- *nombre de archivo*: la ubicación del archivo en el disco. Puede referirse a la capa de salida o a un contenedor que almacena la capa (por ejemplo, formatos similares a bases de datos como GeoPackage, SpatiaLite o Open Document Spreadsheets).
- *CRS*: puede ser cambiado para reproyectar los datos
- *Extent*: restringe la extensión de la entrada que se va a exportar utilizando el widget :ref:`extent_selector `
- *Añadir archivo guardado al mapa*: para añadir la nueva capa al lienzo

Sin embargo, algunos parámetros son específicos de determinados formatos:

Parámetros específicos ráster

Dependiendo del formato a exportar, algunas de estas opciones pueden no estar disponibles:

- *modo de salida* (puede ser **datos crudos** o **imagen renderizada**)
- *Formato*: exporta a cualquier formato ráster en el que GDAL pueda escribir, como GeoTiff, GeoPackage, MBTiles, PDF geoespacial, SAGA GIS Binary Grid, Intergraph Raster, ESRI .hdr etiquetado…
- *Resolución*
- : guilabel:*Opciones de creación*: utilice opciones avanzadas (compresión de archivos, tamaños de bloque, colorimetría …) al generar archivos, ya sea desde perfiles de creación predefinidos relacionados con el formato de salida o configurando cada parámetro.
- Creación de *Piramides*
- *Teselas VRT* en caso de que optes por *Crear VRT*
- *valores Sin Datos*

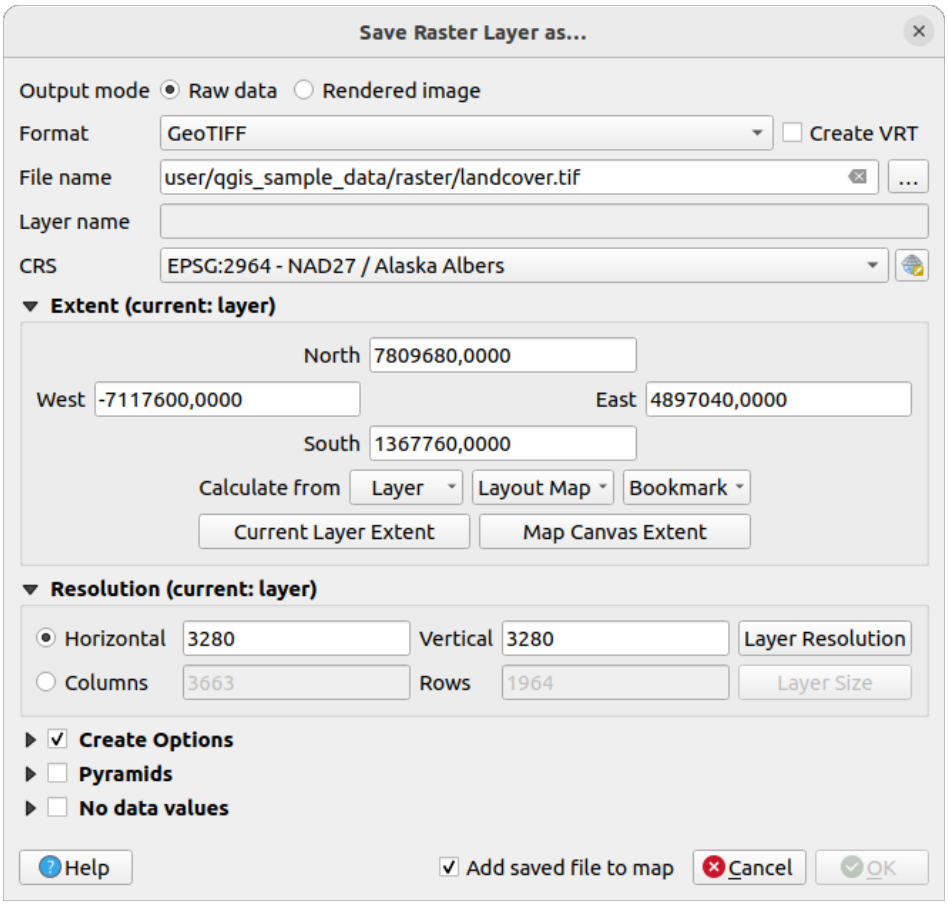


Figura 11.31: Guardando como una nueva capa ráster

Parámetros específicos de Vectorial

Dependiendo del formato a exportar, algunas de estas opciones pueden estar disponibles:

- *Formato*: exporta a cualquier formato vectorial en el que GDAL pueda escribir, como GeoPackage, GML, ESRI Shapefile, AutoCAD DXF, ESRI FileGDB, Mapinfo TAB o MIF, SpatiaLite, CSV, KML, ODS, …
- *nombre de capa*: disponible cuando el *nombre de Archivo* se refiere a un formato de contenedor, esta entrada representa la capa saliente.
- *Codificación*
- *Guardar solo entidades seleccionadas*
- *Seleccionar campos para exportar y sus opciones de exportación*: proporciona medios para exportar campos con nombres personalizados y configuraciones *form widget*:
  - Compruebe las filas bajo la columna *Nombre* para elegir los campos a mantener en la capa de salida o presione los botones *Seleccionar Todo* o *Deseleccionar Todo*
  - Apalanca la caja de confirmación *Usar alias para el nombre exportado* para poblar la columna *Exportar nombre* con el correspondiente campo alias o lo reinicia al nombre original del campo. Doble pulsación en una celda editará el nombre también.
  - Dependiendo de si los widgets personalizados de formulario de atributo están en uso usted puede *Reemplazar todos los valores de campo sin procesar por los valores mostrados*. P.ej. si se aplica un widget `valor de mapa` a un campo, la capa de salida contendrá los valores de la descripción en lugar de los valores originales. El reemplazo se puede hacer también sobre una base de campo por campo en la columna :guilabel:`Reemplazar con los valores mostrados.

- *Persist layer metadata*: asegura que cualquier capa :ref:`metadata ` presente en la capa de origen se copiará y almacenará:
  - en la capa recién creada, si la salida tiene formato GeoPackage
  - como archivo `.qmd` junto con la capa de salida, para otros formatos. Tenga en cuenta que los formatos basados en archivos que admiten más de un conjunto de datos (por ejemplo, SpatiaLite, DXF,…) pueden tener un comportamiento no deseado.
- *exportar Simbología*: se puede utilizar principalmente para la exportación DXF y para todos los formatos de archivo que administran estilos de funciones OGR (ver nota a continuación) como formatos de archivo de pestañas DXF, KML:
  - **Sin simbología**: estilo por defecto de la aplicación que lee los datos
  - **Simbología de entidad**: guardar estilo con Estilos de funciones OGR (ver nota a continuación)
  - **Simbología de la capa de símbolo**: guarde con Estilos de entidades OGR (vea la nota a continuación) pero exporta la misma geometría varias veces si se usan varias capas de símbolo de simbología
  - Se puede aplicar un valor de **Escala** a las últimas opciones

Nota

Los estilos de características de OGR son una forma de almacenar el estilo directamente en los datos como un atributo oculto. Solo algunos formatos pueden manejar este tipo de información. Los formatos de archivo KML, DXF y TAB son tales formatos. Para obtener detalles avanzados, puede leer el documento Especificación de estilos de entidades de OGR.

- *Geometría*: puede configurar las capacidades de geometría de la capa de salida
  - *tipo de geometría*: mantiene la geometría original de las entidades cuando se establece en **Automático**; de lo contrario, la elimina o la reemplaza con cualquier tipo. Puede agregar una columna de geometría vacía a una tabla de atributos y eliminar la columna de geometría de una capa espacial.
  - *Forzar multi tipo*: fuerza la creación de entidades multi-geometría en la capa.
  - *Incluir dimensión z* en geometrías.

Truco

Anular el tipo de geometría de capa permite hacer cosas como guardar una tabla sin geometría (por ejemplo archivos `.csv`) en un archivo de forma CON cualquier tipo de geometría (punto, línea, polígono), de modo que las geometrías se puedan agregar manualmente a filas con la herramienta  Agregar pieza.

- *Opciones de origen de datos*, *Opciones de capa* o *Opciones personalizadas* que le permiten configurar parámetros avanzados según el formato de salida. Algunos se describen en *Explorando campos y formatos de datos* pero para obtener detalles completos, consulte la documentación del controlador GDAL. Cada formato de archivo tiene sus propios parámetros personalizados, p. Ej. para el formato `GeoJSON` eche un vistazo a la documentación GDAL GeoJSON.

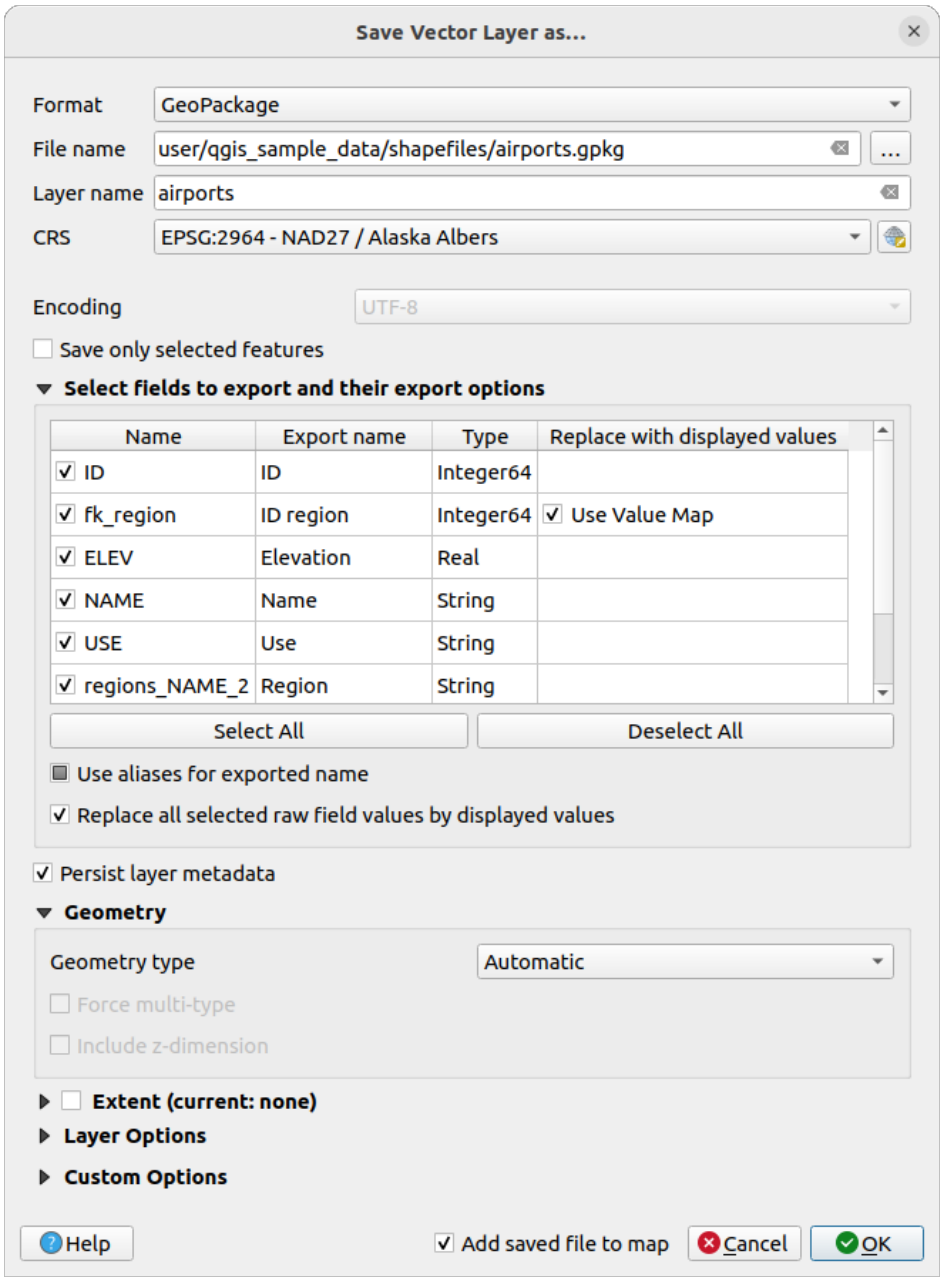


Figura 11.32: Guardando como una nueva capa vectorial

Al guardar una capa vectorial en un archivo existente, dependiendo de las capacidades del formato de salida (Geopackage, SpatiaLite, FileGDB …), el usuario puede decidir si:

- sobreescribir el archivo entero
- sobreescribir solo la capa objetivo (el nombre de capa es configurable)
- agregar entidades a la capa de destino existente
- agregar entidades, agregar nuevos campos si los hay.

Para formatos como ESRI Shapefile, MapInfo .tab, también está disponible la función de agregar.

Parámetros específicos de la nube de puntos

Al igual que las capas ráster y vectoriales, las capas de nubes de puntos pueden guardarse en un formato diferente y/o reproyectarse a un sistema de referencia de coordenadas (SRC) distinto. Esto permite exportar una capa de nube de puntos a formatos vectoriales o de nube de puntos. Los formatos admitidos actualmente son: Scratch temporal (capa de memoria), GeoPackage, ESRI Shapefile, DXF y nube de puntos LAS/LAZ. Además de los parámetros comunes enumerados anteriormente, la exportación de capas de nubes de puntos incluye las siguientes opciones:

- *Filtrar por capa poligonal*: Permite filtrar los datos de la nube de puntos en función de una capa poligonal.
- *Rango de elevación*: Permite filtrar los datos de la nube de puntos en función de un rango Z especificado.
- *Límitar número de puntos*: Proporciona una opción para limitar el número de puntos exportados desde la capa de nube de puntos.

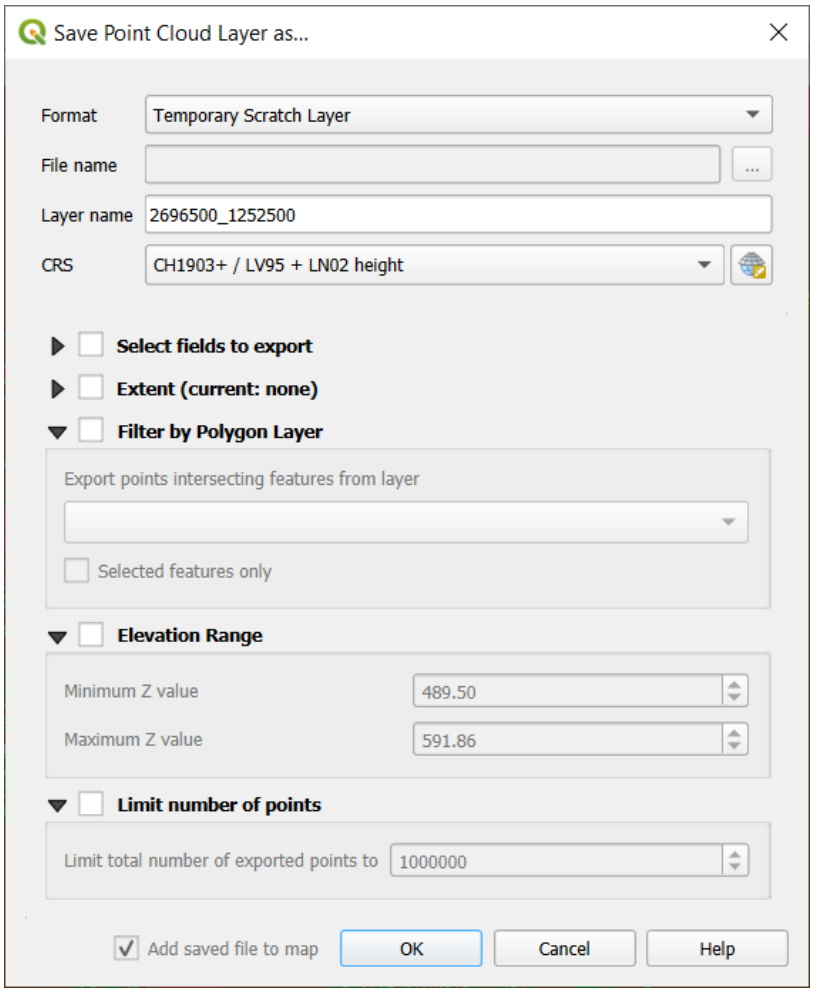


Figura 11.33: Guardar una capa de nube de puntos como una nueva capa

11.2.3 Creando nuevos archivos DXF

Además del diálogo *Guardar como…* que proporciona opciones para exportar una sola capa a otro formato, que incluye `*.DXF`, QGIS proporciona otra herramienta para exportar múltiples capas como una sola capa DXF. Es accesible en el menú *Proyecto -> Importar / Exportar -> Exportar proyecto a DXF …*.

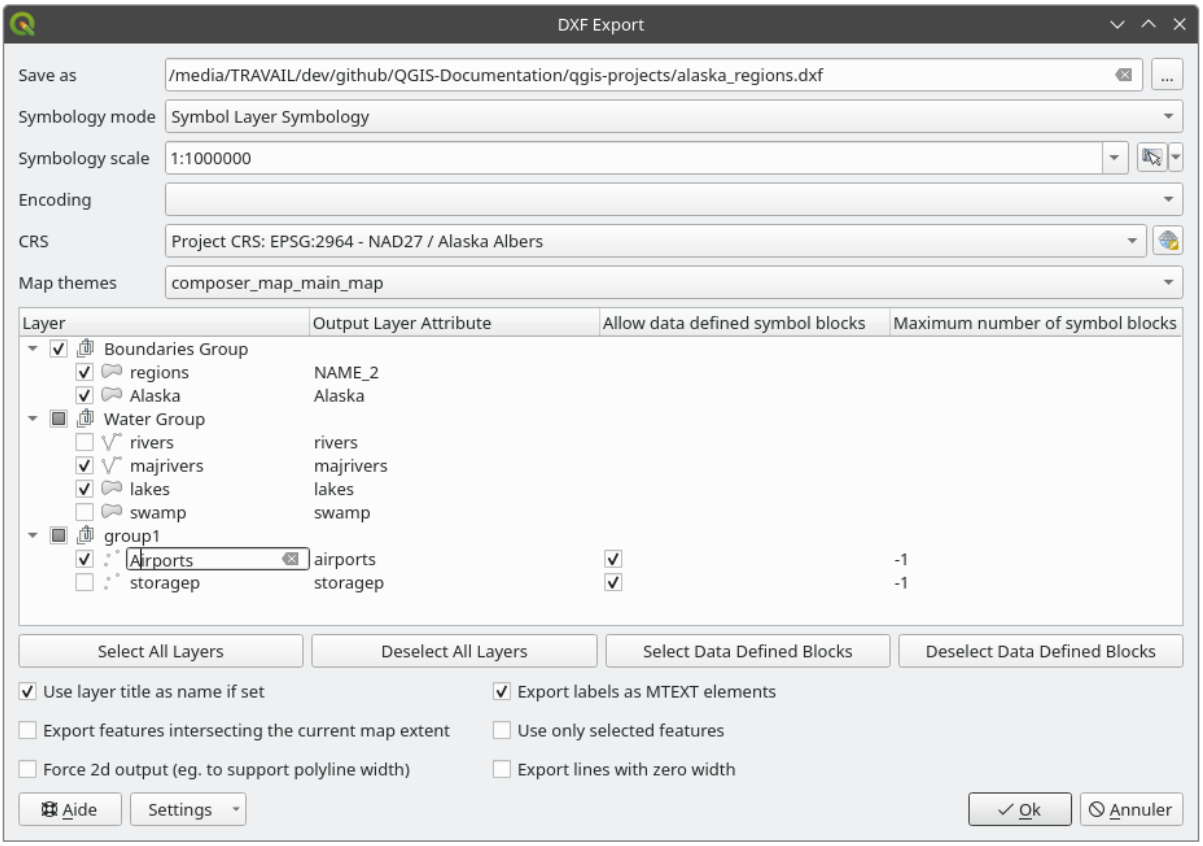


Figura 11.34: Diálogo Exportar un proyecto a DXF

En el diálogo *Exportar DXF*

1. Proporciona el archivo destinatario.
2. Escoge el modo y la escala de simbología (ver la nota *OGR Feature Styles*), si es aplicable.
3. Selecciona la *Codificación* de los datos
4. Selecciona el *CRS* a aplicar: las capas seleccionadas se volverán a proyectar al CRS dado.
5. Seleccione las capas para incluir en los archivos DXF, ya sea marcándolas en el widget de tabla o seleccionándolas automáticamente de un *tema de mapa* existente. Los botones *Seleccionar todo* y :guilabel:` Deseleccionar todo` pueden ayudar a configurar rápidamente los datos para exportar.

   Para cada capa, puede:

   - Anule el nombre de la capa de salida sin alterar la capa original del proyecto. Para ello, haga clic en *Nombre de capa* en el cuadro de diálogo y escriba el nombre de salida que desea utilizar.
   - *Atributo de capa de salida*: Elija si desea exportar todas las características en una sola capa DXF o confiar en un campo cuyos valores se utilizan para dividir las características en capas en la salida DXF. En este último caso, cada capa tomará su nombre del valor del campo correspondiente.
   - *Permitir bloques de símbolos definidos por datos*:

- *Número máximo de bloques de símbolos*: crea bloques de símbolos hasta el límite especificado, empezando por los que contienen el mayor número de referencias. Los demás símbolos se escriben tal cual. `-1` significa que no hay limitación.

Opcionálmente, puede además elegir:

- ☐ *Utilizar el título de la capa como nombre si se establece* en lugar del propio nombre de la capa: el título se toma de los *metadatos* o *server* propiedades de la capa;
- ☐ *Exportar objetos espaciales que intersectan la extensión actual del mapa*;
- ☐ *Forzar salida 2d (por ejemplo, para admitir el ancho de polilínea)*;
- ☑ *Exporta la etiqueta como elementos MTEXT* o TEXT;
- ☐ *Utilizar sólo los objetos espaciales seleccionados*;
- ☐ *Exportar líneas con anchura cero*: todas las líneas se exportan con anchura mínima `0` (hairline) si está activado. Esto ayuda a mantener las líneas mínimas en el archivo, independientemente del nivel de zoom, y puede ser útil para hacer más CAD-edición con el dxf exportado, especialmente si hay muchas características junto a la otra en el mapa.

Nota

La precedencia para definir el nombre de la capa de salida es la siguiente:

1. El valor del campo de *Atributo de capa de salida*.
2. El nombre sobrescrito en la columna *Capa*
3. La opción *Utilizar el título de la capa como nombre si está establecido*.
4. El nombre de la capa

La configuración actual definida en el cuadro de diálogo *Exportar DXF* puede almacenarse en un archivo XML para reutilizarla en otras sesiones. Para ello, el cuadro combinado *Settings* tiene dos opciones: *Cargar configuración desde archivo…* y *Guardar configuración en archivo…*.

11.2.4 Creando nuevas capas desde el portapapeles

Las entidades que están en el portapapeles se pueden pegar en una nueva capa. Para hacer esto, seleccione algunas entidades, cópielas en el portapapeles y luego péguelas en una nueva capa usando *Editar -> Pegar entidades como ->* y eligiendo:

- *Nueva capa vectorial…*: aparece el cuadro de diálogo *Guardar capa vectorial como…* (ver *Creando nuevas capas desde una capa existente* para ver los parámetros)
- o *Capa Borrador temporal…*: necesitas proporcionar un nombre a la capa

Se crea una nueva capa, llena de entidades seleccionadas y sus atributos (y se agrega al lienzo del mapa).

Nota

Es posible crear capas desde el portapapeles con características seleccionadas y copiadas dentro de QGIS, así como características de otra aplicación, siempre que sus geometrías se definan utilizando texto conocido (WKT).

11.2.5 Creating SQL Query Layers

Beside loading an entire layer in a project or creating new layers from scratch or pasted features, you can also load layers generated on the fly from other layer(s). They are the result of a more or less advanced filter using SQL language, applied to stored layers regardless of their data provider or their availability in the active project. This means that temporary layers are not compatible with this feature. Depending on the provider, one or more layers can be used to write the query. The generated layer remains dependent on the layer(s) involved in the query and is loaded with the  Filter icon next to it.

This feature is accessible:

- From the *Browser* panel, right-click on a supported data (plain layer, database connection, schema or table) and select *Execute SQL…* entry in the contextual menu.
- From the *Layers* panel, select a loaded layer, right-click and select *Execute SQL…* entry in the contextual menu.

This opens a window with a central text box widget where you can write SQL queries.

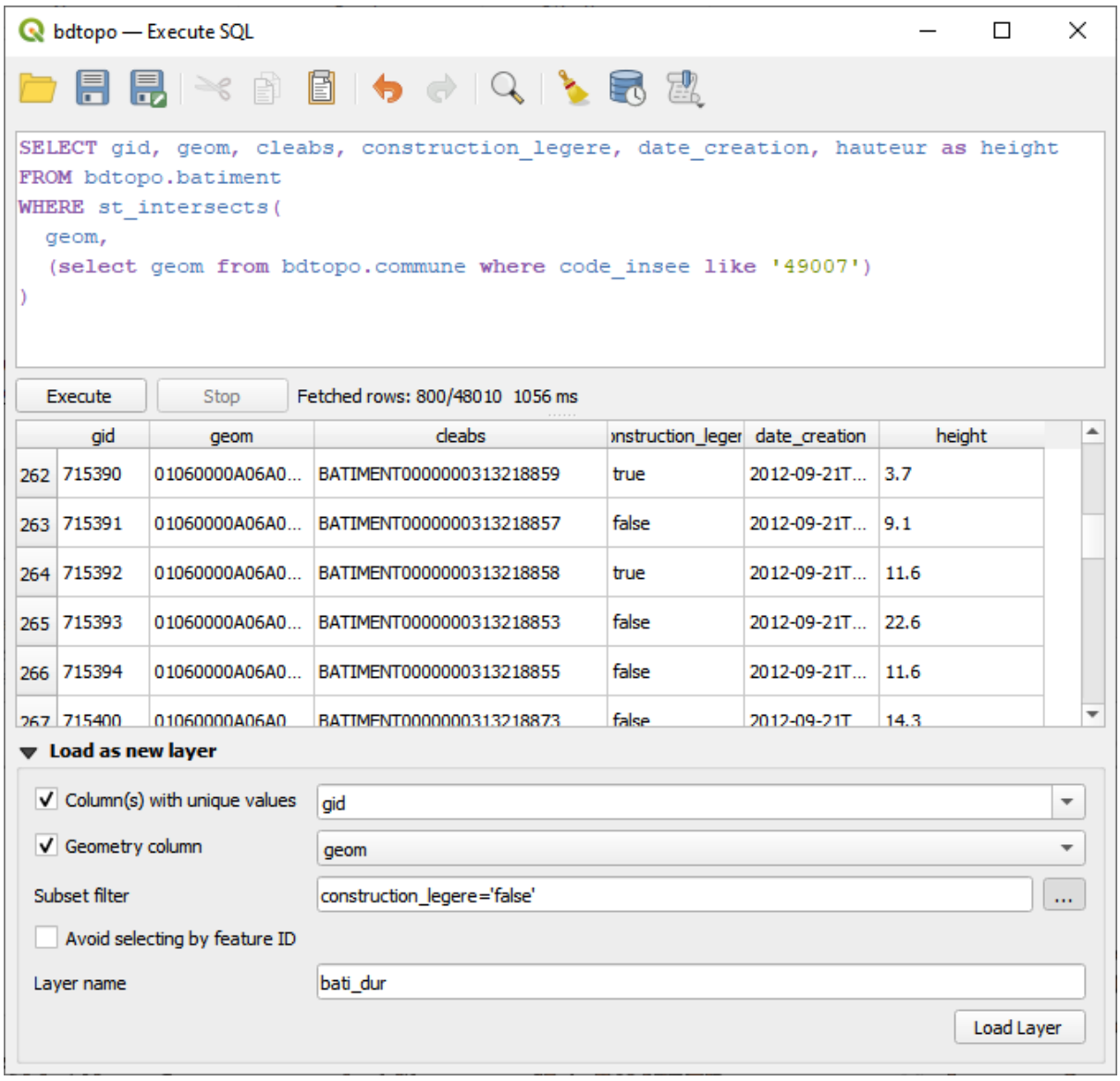


Figura 11.35: Executing SQL queries in the Execute SQL window

At the top of the dialog, a toolbar provides a set of tools to create, store and manipulate your queries:

- Open Queries…: fills the text editor widget with contents from an existing `.sql` file
- Save Queries… and Save Queries as… help you store the written query to a `.sql` file
- Statements of the query can be adjusted using the Cut, Copy and Paste buttons. Likewise, you can Undo or Redo your changes.
- The Find & Replace enables, at the bottom of the dialog, a widget allowing to look for a particular string in your SQL code. The search can be case sensitive, affect partial or whole word, rely on a regular expression. It is then possible to navigate through the found strings, replacing them one by one or all in a row.
- Use Clear SQL Editor to wipe the text editor.
- The History button opens a dialog storing previously run queries. More at *Query History*.
- As previously mentioned, queries can be saved as an `.sql` file stored on disk. Using the Store Current Query button, they can also be stored:
  - In the active *User Profile*, in the associated `QGIS3.ini` file, thus accessible in subsequent projects
  - or as part of the *Current Project*.

  Clicking an entry from the stored queries drop-down menu inserts that query in the expression being written. The stored entry can also be deleted, from the drop-down menu.

Running and loading queries as layer

In the central part of the *Execute SQL* dialog, you build your query using the SQL syntax supported by the underlying provider (e.g., OGR, GeoPackage, PostgreSQL).

By default, if opened from a layer entry, a sample SQL query is provided. Editing tools to select, cut, copy, paste, undo and redo are available as well from the contextual menu.

> **Truco**
>
> **Finding the right SQL syntax for your dataset**
>
> To find the appropriate SQL dialect for your dataset, press the … button next to the *Subset filter* option in the bottom part of the dialog.

When ready, pressing the *Execute* button below the text area will run the query. It is possible to highlight a portion of the SQL to only execute that portion when pressing `Ctrl+R` or clicking the *Execute selection* button. Use the *Stop* button to abort the execution.

A successful query execution will display a table at the bottom of the dialog with returned features. You can select specific cells in the result set. Use the `Ctrl+C` shortcut to copy the selected cells to the clipboard. The copied data is available as a formatted table. This allows you to paste the data into other applications, such as spreadsheet where it will show up as a table.

The returned table can be loaded in QGIS expanding the *Load as new layer* group and configuring parameters (their availability depends on the layer provider):

- *Column(s) with unique values* to indicate primary key of the data,
- *Geometry column*: check the box to load the layer as a spatial one, and indicate the geometry field name
- *Subset filter*: allows to filter the results using a `WHERE` clause. It can be written in the text box, or built using the *Query builder* after you pressed …. Make sure to use fields that are available in the SQL layer.
- *Avoid selecting by feature ID*

- *Layer name* in the project

At any time, in the *Layers panel*, you can adjust the output layer by right-clicking and selecting *Update SQL expression…*. The *Update SQL* dialog opens, prefilled with the applied query that you can edit as you wish. Once ready, press *Update layer* and the layer will be modified in-place.

Query History

The *Query History* dialog displays all the previously run queries, sorted by date and data provider type. It is accessible from the *Database* ► *Query History…* menu, or pressing the History button from the *Execute SQL* dialog.

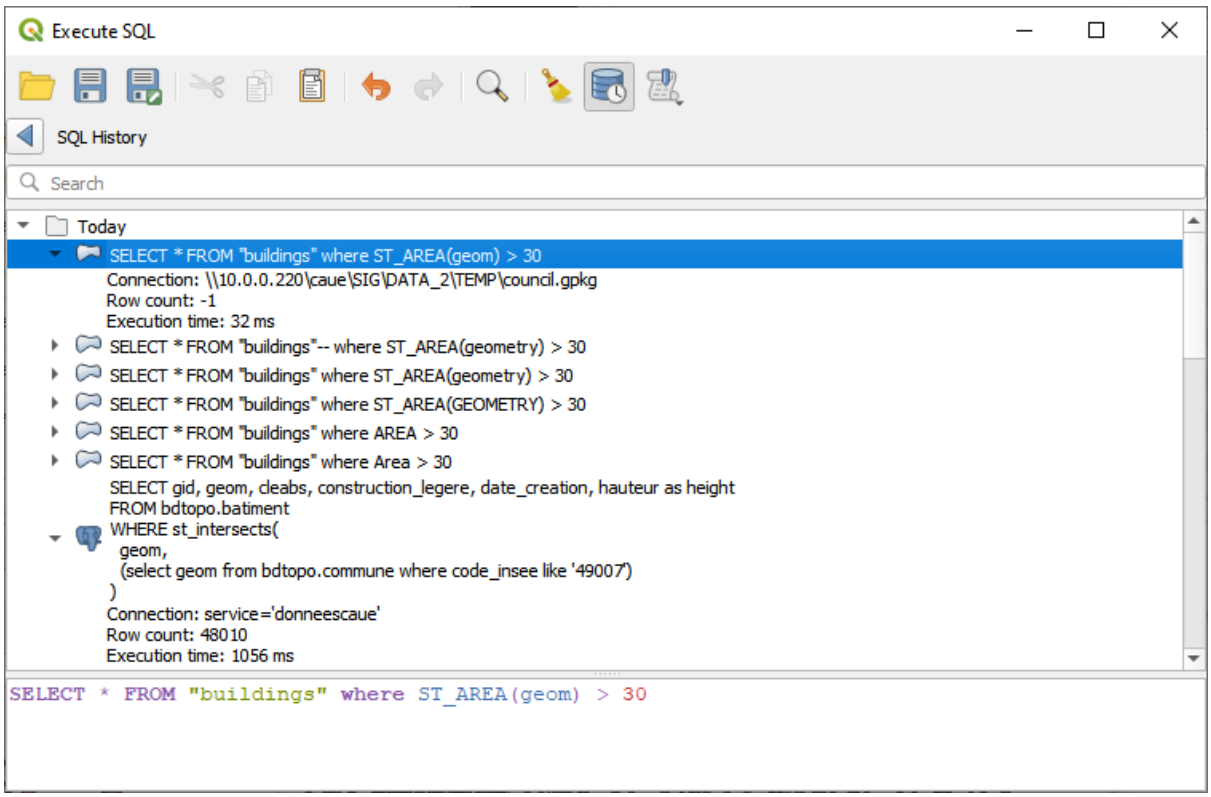


Figura 11.36: History of the executed SQL queries

At the top of the dialog, a search box can be used to filter out the queries. The search is done through every strings displayed in the query history. If the dialog is opened from within the *Execute SQL* dialog, the search box is preceded by the *Execute SQL toolbar*.

Below, collapsible and sorted date entries list their related queries. Each query is identifiable by its SQL command, and when expanded, reveals:

- The *Connection*: the source of the data the query is based on
- The *Row count*: number of returned features
- The *Execution time*

Hover over an entry and a tooltip displays the full query. Likewise, selecting an entry also displayed it in the bottom part of the dialog. You can interact with the text, copying all or part of it.

Double-click the SQL command entry and the *Execute SQL* dialog opens, filled with the clicked query. Any query being written is overwritten. Right-click the SQL command entry and you can either:

- *Load SQL Command…*, loads the target command into the *Execute SQL* dialog, replacing any existing query. It is the same as double clicking the entry.

- *Copy SQL Command* and paste it wherever you want.

11.2.6 Creando capas virtuales

Una capa virtual es un tipo especial de capa vectorial. Le permite definir una capa como resultado de una consulta SQL que involucra cualquier cantidad de otras capas vectoriales que QGIS es capaz de abrir. Las capas virtuales no portan datos por sí mismas y pueden verse como vistas.

Para crear una capa virtual, abre el diálogo de capa virtual mediante:

- escogiendo la entrada  *Añadir/Editar Capa Virtual* en el menú *Capa* ► *Añadir Capa* ►;
- habilitando la pestaña  *Agregar capa virtual* en el diálogo *Administrador de fuentes de datos*;
- o usando el diálogo del árbol :guilabel:Àdministrador de DB`.

El diálogo le permite especificar un *Nombre de capa* y una *Consulta* SQL. La consulta puede usar el nombre (o id) de las capas vectoriales cargadas como tablas, así como sus nombres de campo como columnas.

Por ejemplo, si tiene una capa llamada `airports`, puede crear una nueva capa virtual llamada `public_airports` con una consulta SQL como:

```
SELECT *
FROM airports
WHERE USE = "Civilian/Public"
```

La consulta SQL se ejecutará, independientemente del proveedor subyacente de la capa de `airports`, incluso si este proveedor no admite directamente consultas SQL.

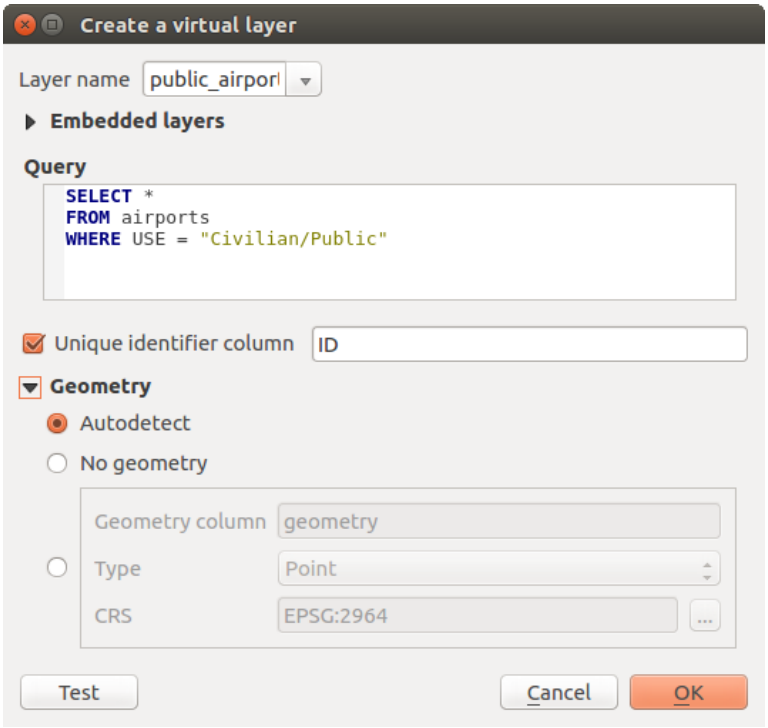


Figura 11.37: Diálogo Crear Capas Virtuales

También se pueden crear combinaciones y consultas complejas, por ejemplo, para unir información sobre aeropuertos y países:

```
SELECT airports.*, country.population
FROM airports
JOIN country
ON airports.country = country.name
```

Nota

También es posible crear capas virtuales usando la ventana SQL de *Complemento de Administración de BBDD*.

Incorporación de capas para su uso en consultas

Además de las capas vectoriales disponibles en el lienzo del mapa, el usuario puede agregar capas a la lista *Capas incrustadas*, que se pueden usar en consultas sin necesidad de que se muestren en el lienzo del mapa o en el panel Capas.

Para incrustar una capa, haz click en *Agregar* y proporciona *Nombre local*, *Proveedor*, *Codificación* y la ruta a la *Fuente*.

El botón *Importar* permite agregar capas en el lienzo del mapa en la lista de capas incrustadas. A continuación, esas capas se pueden eliminar del panel Capas sin interrumpir las consultas existentes.

Lenguaje soportado de consultas

El motor subyacente utiliza SQLite y SpatiaLite para operar.

Significa que puede usar todo el SQL que comprenda su instalación local de SQLite.

Las funciones de SQLite y las funciones espaciales de SpatiaLite también se pueden utilizar en una consulta de capa virtual. Por ejemplo, se puede crear una capa de puntos a partir de una capa de solo atributo con una consulta similar a:

```
SELECT id, MakePoint(x, y, 4326) as geometry
FROM coordinates
```

Funciones de expresiones QGIS también puede utilizarse en una consulta de capa virtual.

Para referirse a la columna de geometría de una capa, use el nombre `geometry`.

A diferencia de una consulta SQL pura, todos los campos de una consulta de capa virtual deben tener un nombre. No olvide usar la palabra clave `as` para nombrar sus columnas si son el resultado de un cálculo o una llamada a una función.

Problemas de Rendimiento

Con los parámetros predeterminados, el motor de capa virtual hará todo lo posible para detectar el tipo de las diferentes columnas de la consulta, incluido el tipo de columna de geometría, si existe una.

Esto se hace con una introspección de la consulta cuando sea posible o obteniendo la primera fila de la consulta (LIMIT 1) como último recurso. Obtener la primera fila del resultado solo para crear la capa puede no ser deseable por razones de rendimiento.

Los parámetros del diálogo creación:

- *Columna de identificador único*: especifica un campo de la consulta que representa valores enteros únicos que QGIS puede usar como identificadores de fila. De forma predeterminada, se utiliza un valor entero que aumenta automáticamente. La definición de una columna de identificador único acelera la selección de filas por id.
- *Sin geometría*: obliga a la capa virtual a ignorar cualquier campo de geometría. La capa resultante es una capa de solo atributos.

- *Columna* Geometría: especifica el nombre de la columna de geometría.
- *Tipo* de Geometría: especifica el tipo de geometría.
- *CRS* de Geometría: especifica el sistema de referencia de coordenadas de la capa virtual.

Comentarios Especiales

El motor de capa virtual intenta determinar el tipo de cada columna de la consulta. Si falla, se recupera la primera fila de la consulta para determinar los tipos de columna.

El tipo de una columna en particular se puede especificar directamente en la consulta mediante algunos comentarios especiales.

La sintaxis es la siguiente: `/*:type*/`. Debe colocarse justo después del nombre de una columna. `type` puede ser `int` para números enteros, `real` para números de coma flotante o `text`.

Por ejemplo:

```
SELECT id+1 as nid /*:int*/
FROM table
```

El tipo y sistema de referencia de coordenadas de la columna de geometría también se puede configurar gracias a comentarios especiales con la siguiente sintaxis `/*:gtype:srid*/` donde `gtype` es el tipo de geometría (`punto`, `línea`, `polígono`, `multipunto`, `multilínea` o `multipolígono`) y `srid` un número entero que representa el código EPSG de un sistema de referencia de coordenadas.

Uso de índices

Al solicitar una capa a través de una capa virtual, los índices de la capa fuente se utilizarán de las siguientes formas:

- Si se usa un predicado = en la columna de clave principal de la capa, se le pedirá al proveedor de datos subyacente una identificación particular (FilterFid)
- para cualquier otro predicado (``>``, `` <= ``, ``!=``, etc.) o en una columna sin una clave primaria, se utilizará una solicitud construida a partir de una expresión para solicitar el proveedor de datos vectoriales. Significa que los índices se pueden utilizar en proveedores de bases de datos, si existen.

Existe una sintaxis específica para manejar los predicados espaciales en las solicitudes y desencadena el uso de un índice espacial: existe una columna oculta llamada `_search_frame_` para cada capa virtual. Esta columna se puede comparar por igualdad con un cuadro delimitador. Ejemplo:

```
SELECT *
FROM vtab
WHERE _search_frame_=BuildMbr(-2.10,49.38,-1.3,49.99,4326)
```

Los predicados binarios espaciales como `ST_Intersects` se aceleran significativamente cuando se usan junto con esta sintaxis de índice espacial.

11.3 Georreferenciador

El georreferenciador es una herramienta para alinear capas ráster o vectoriales no referenciadas con sistemas de coordenadas conocidos utilizando puntos de control terrestre (GCP). Soporta la exportación de rásters transformados (GeoTIFF) y vectores (shapefiles, geopackages, etc.) o la escritura de archivos World (sólo para rásters). El método básico para georreferenciar una capa consiste en localizar en ella puntos cuyas coordenadas puedan determinarse con precisión.

Entidades

Tabla 11.1: Georeferencer Tools

Icono	Propósito	Icono	Propósito
	Abrir ráster		Abrir vector
	Comenzar georrefeerenciado		Generar Script de GDAL
	Cargar puntos GCP		Guardar Puntos GCP como
	Configuración de transformación		
	Agregar puntos GCP		Eliminar punto GCP
	Mover punto GCP		Snapping Type
	Show Advanced Digitizing Dock		Mover
	Acercar Zoom		Alejar Zoom
	Zoom a capa		Zoom último
	Zoom siguiente		Enlazar Gerreferenciador a QGIS
	Enlazar QGIS a georreferenciador		Estiramiento de Histograma Total
	Estiramiento de Histograma Local		

11.3.1 Capa ráster de georreferenciación

Como coordenadas X e Y (DMS (dd mm ss.ss), DD (dd.dd) o coordenadas proyectadas (mmmm.mm)), que se corresponden con el punto seleccionado en la imagen, se pueden utilizar dos procedimientos alternativos:

- El mismo ráster a veces proporciona cruces con coordenadas «escritas» en la imagen. En este caso, puede ingresar las coordenadas manualmente.
- Utilizando capas ya georreferenciadas. Estos pueden ser datos vectoriales o ráster que contienen los mismos objetos/entidades que tiene en la imagen que desea georreferenciar y con la proyección que desea para su imagen. En este caso, puede ingresar las coordenadas haciendo clic en el conjunto de datos de referencia cargado en el lienzo del mapa de QGIS.

El procedimiento habitual para georreferenciar una imagen implica seleccionar varios puntos en el ráster, especificar sus coordenadas y elegir un tipo de transformación relevante. Basado en los parámetros de entrada y los datos, el Georreferenciador calculará los parámetros del archivo mundial. Cuantas más coordenadas proporcione, mejor será el resultado.

El primer paso es iniciar QGIS y hacer clic en *Capa* ► *Georeferenciador*, que aparece en la barra de menús de QGIS. El diálogo Georreferenciador aparece como se muestra en Figura 11.38.

Para este ejemplo, estamos usando una hoja topográfica de Dakota del Sur de SDGS. Posteriormente se puede visualizar junto con los datos de la localización GRASS `spearfish60`. Puede descargar la hoja top aquí: https://grass.osgeo.org/sampledata/spearfish_toposheet.tar.gz.

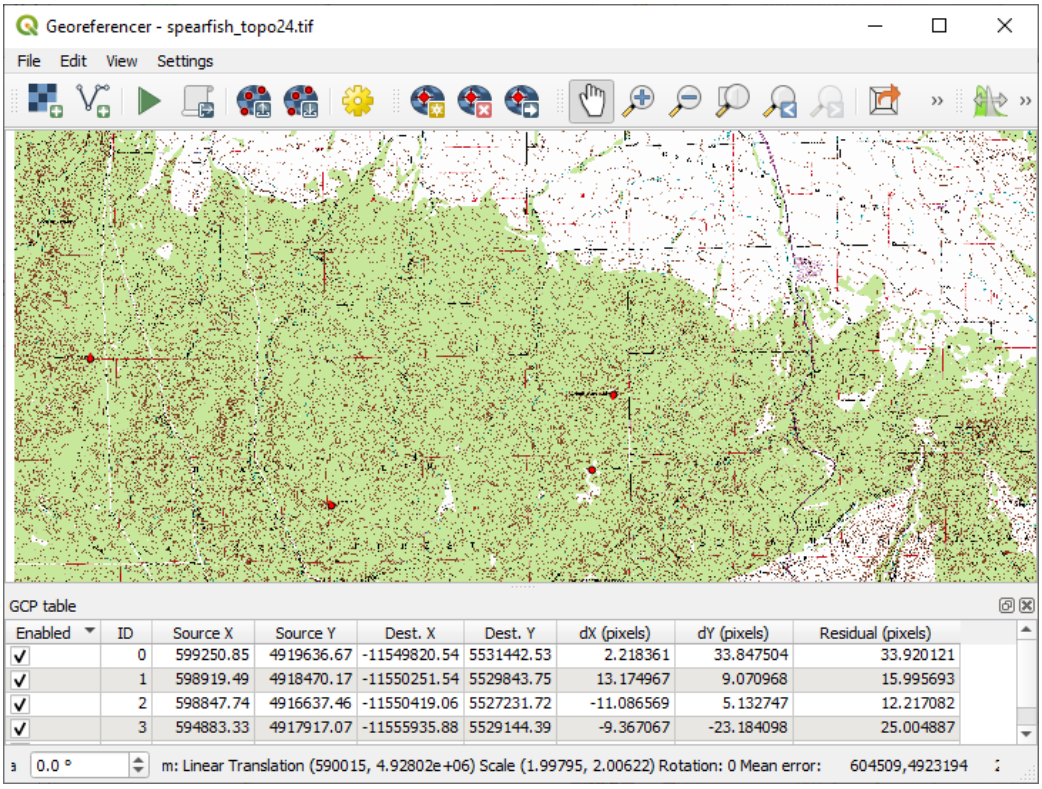


Figura 11.38: Diálogo Georreferenciador

Entrar puntos de control terrestre (GCP)

1. Para comenzar a georreferenciar un ráster sin referencia, debemos cargarlo usando el botón . El ráster se mostrará en el área de trabajo principal del cuadro de diálogo. Una vez cargado el ráster, podemos comenzar a ingresar puntos de referencia.

2. Usando el botón Añadir Punto PCT, añada puntos al área de trabajo principal e introduzca sus coordenadas (ver Figura Figura 11.39). Para este procedimiento dispone de las siguientes opciones:

   - Haga clic en un punto en la imagen ráster e ingrese las coordenadas X e Y manualmente, junto con el CRS del punto.
   - Haga clic en un punto de la imagen ráster y elija el botón From map canvas para agregar las coordenadas X e Y con la ayuda de un mapa georreferenciado ya cargado en el lienzo del mapa de QGIS. El CRS se configurará automáticamente.
   - Al introducir los PCTs desde el lienzo principal del mapa, tiene la opción de ocultar la ventana del georreferenciador mientras selecciona los puntos desde el lienzo principal. Si la casilla *Ocultar automáticamente la ventana del georreferenciador* está marcada, después de hacer clic en *Desde el lienzo del mapa*, la ventana principal del georreferenciador se ocultará hasta que se añada un punto en el lienzo del mapa. El cuadro de diálogo *Introducir coordenadas del mapa* permanecerá abierto. Si la casilla no está marcada, ambas ventanas permanecerán abiertas mientras se selecciona un punto en el lienzo del mapa. Esta opción sólo tiene efecto cuando la ventana del georreferenciador no está anclada en la interfaz principal.

3. Continúe ingresando puntos. Debe tener al menos cuatro puntos, y cuantas más coordenadas pueda proporcionar, mejor será el resultado. Hay herramientas adicionales para hacer zoom y paneo del área de trabajo con el fin de ubicar un conjunto relevante de puntos GCP.

> **Truco**
>
> Para evitar el cambio constante entre los botones Pan, Añadir punto PCT y Mover punto GCP, puede utilizar convenientemente las teclas de flecha del teclado para mover y la rueda del ratón para escalar el mapa georreferenciado.

4. Después de proporcionar algunos puntos, puede utilizar los botones Enlazar QGIS a Georeferencer y/o Enlazar Georeferencer a QGIS que ajustarán, respectivamente, la extensión del mapa de la ventana principal de QGIS a la vista actual en Georeferencer y/o viceversa.

5. Con la herramienta , puede mover los GCP tanto en el lienzo como en la ventana de georreferenciación, si necesita corregirlos.

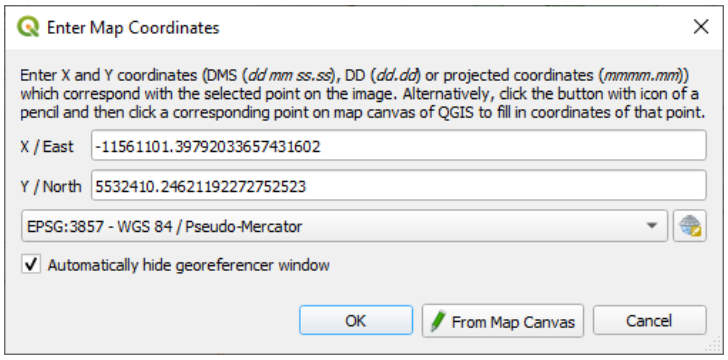


Figura 11.39: Añadir puntos a la imagen ráster

Los puntos que se agregan al mapa se almacenarán en un archivo de texto separado (`[filename].points`) generalmente junto con la imagen rasterizada. Esto nos permite reabrir el Georreferenciador en una fecha posterior y agregar nuevos puntos o eliminar los existentes para optimizar el resultado. El archivo de puntos contiene valores de la forma: `mapX, mapY, pixelX, pixelY`. Puedes usar los botones Cargar puntos GCP y Guardar puntos GCP como para administrar los archivos.

Definiendo la configuración de transformación

Una vez que haya agregado sus GCP a la imagen ráster, debe definir la configuración de transformación para el proceso de georreferenciación.

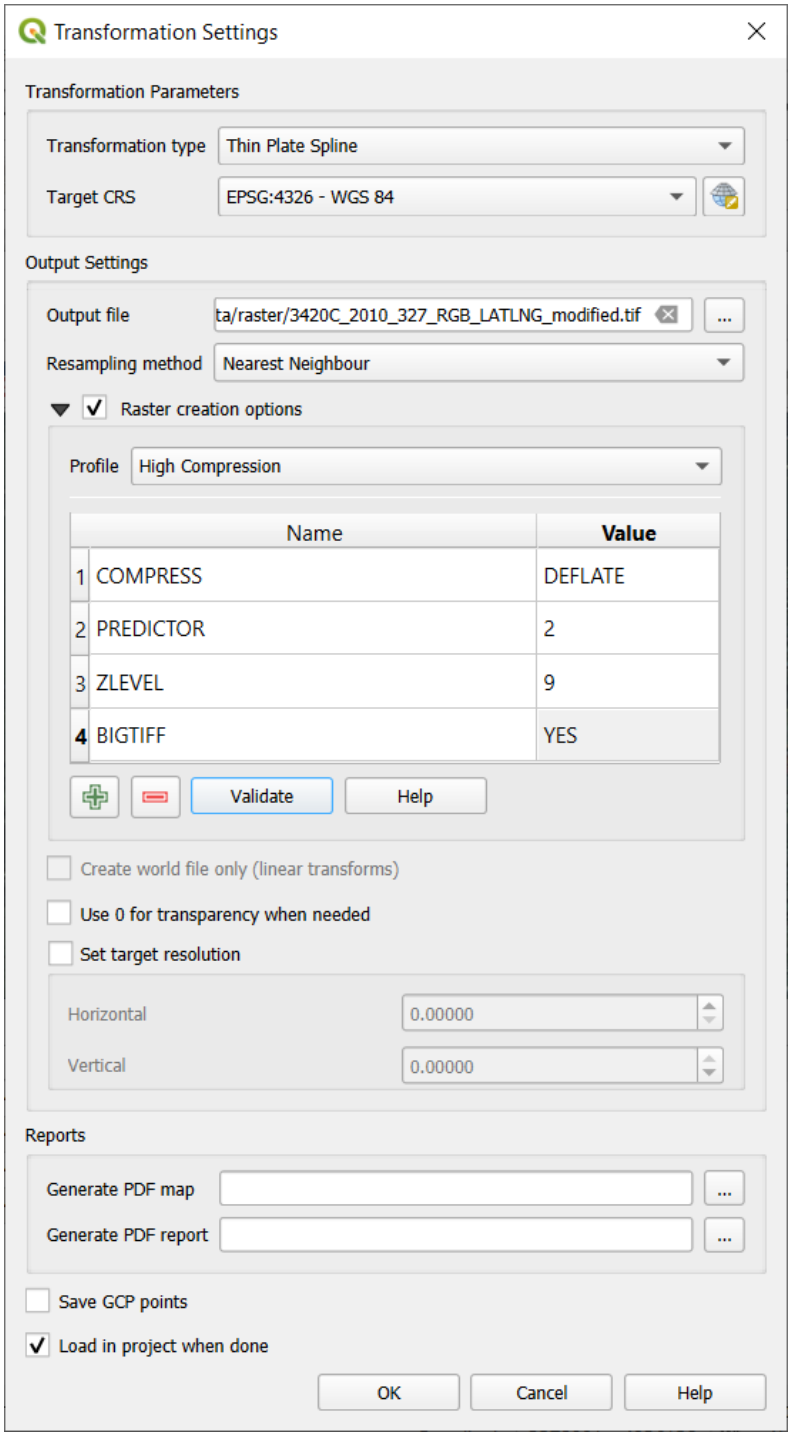


Figura 11.40: Definición de la configuración de transformación del georreferenciador

Algoritmos de transformación disponibles

Hay varios algoritmos de transformación disponibles, según el tipo y la calidad de los datos de entrada, la naturaleza y la cantidad de distorsión geométrica que desee introducir en el resultado final y la cantidad de puntos de control terrestre (GCP).

Actualmente, los siguientes *Tipos de transformación* están disponibles:

- El algoritmo **Lineal** se usa para crear un archivo mundial y es diferente de los otros algoritmos, ya que en realidad no transforma los píxeles de trama. Permite posicionar (traducir) la imagen y escalar uniformemente, pero no rotar ni otras transformaciones. Es el más adecuado si su imagen es un mapa ráster de buena calidad, en un CRS conocido, pero solo le falta información de georreferenciación. Se necesitan al menos 2 GCP.
- La transformación de **Helmert** también permite la rotación. Es particularmente útil si su ráster es un mapa local de buena calidad o una imagen aérea ortorrectificada, pero no está alineado con la cuadrícula de su CRS. Se necesitan al menos 2 GCP.
- El algoritmo **Polinomio 1** permite una transformación afín más general, en particular también una cizalla uniforme. Las líneas rectas permanecen rectas (es decir, los puntos colineales permanecen colineales) y las líneas paralelas permanecen paralelas. Esto es especialmente útil para cartogramas de datos de georreferenciación, que pueden haberse trazado (o recopilado datos) con diferentes tamaños de píxeles en diferentes direcciones. Se requieren al menos 3 GCP.
- Los algoritmos de **polinomios** 2-3 utilizan polinomios de segundo o tercer grado más generales en lugar de solo transformaciones afines. Esto les permite tener en cuenta la curvatura u otras deformaciones sistemáticas de la imagen, por ejemplo, mapas fotografiados con bordes curvos. Se requieren al menos 6 (respectivamente 10) GCP. Los ángulos y la escala local no se conservan ni se tratan de manera uniforme en toda la imagen. En particular, las líneas rectas pueden volverse curvas y puede haber una distorsión significativa introducida en los bordes o lejos de cualquier GCP que surja de extrapolar demasiado los polinomios ajustados a los datos.
- El algoritmo **Proyectivo** generaliza el Polinomio 1 de una manera diferente, permitiendo transformaciones que representan una proyección central entre 2 planos no paralelos, la imagen y el lienzo del mapa. Las líneas rectas permanecen rectas, pero no se conserva el paralelismo y la escala a lo largo de la imagen varía de manera constante con el cambio de perspectiva. Este tipo de transformación es más útil para georreferenciar fotografías en ángulo (en lugar de escaneos planos) de mapas de buena calidad o imágenes aéreas oblicuas. Se requiere un mínimo de 4 GCP.
- Por último, el algoritmo **Thin Plate Spline** (TPS) «hoja de goma» el ráster usando múltiples polinomios locales para coincidir con los GCP especificados, con la curvatura general de la superficie minimizada. Las áreas alejadas de los GCP se moverán en la salida para adaptarse a la coincidencia de GCP, pero de lo contrario se deformarán localmente de forma mínima. TPS es más útil para georreferenciar mapas dañados, deformados o ligeramente inexactos, o antenas mal ortorrectificadas. También es útil para la georreferenciación aproximada y la reproyección implícita de mapas con un tipo de proyección o parámetros desconocidos, pero donde una cuadrícula regular o un conjunto denso de GCP ad-hoc pueden coincidir con una capa de mapa de referencia. Técnicamente requiere un mínimo de 10 GCP, pero generalmente más para tener éxito.

En todos los algoritmos excepto TPS, si se especifican más de los GCP mínimos, los parámetros se ajustarán de manera que se minimice el error residual general. Esto es útil para minimizar el impacto de los errores de registro, es decir, ligeras imprecisiones en los clics del puntero o las coordenadas escritas, u otras pequeñas deformaciones de la imagen local. En ausencia de otros GCP para compensar, tales errores o deformaciones podrían traducirse en distorsiones significativas, especialmente cerca de los bordes de la imagen georreferenciada. Sin embargo, si se especifican más de los GCP mínimos, solo coincidirán aproximadamente en la salida. Por el contrario, TPS coincidirá con precisión con todos los GCP especificados, pero puede introducir deformaciones significativas entre los GCP cercanos con errores de registro.

Definir el método de remuestreo

El tipo de remuestreo que elija probablemente dependerá de sus datos de entrada y del objetivo final del ejercicio. Si no desea cambiar las estadísticas del ráster (aparte de lo implícito en la escala geométrica no uniforme si se utilizan transformaciones distintas a las lineales, de Helmert o polinómicas 1), puede elegir “Nearest neighbour”. Por el contrario, el “cubic resampling”, por ejemplo, generalmente generará un resultado visualmente más suave.

Es posible elegir entre cinco métodos de remuestreo diferentes:

1. Vecino mas próximo
2. Bilineal (núcleo 2x2)
3. Cúbico (núcleo 4x4)
4. B-Spline cúbica (núcleo 4x4)
5. Lanczos (núcleo 6x6)

Define the Raster creation options

When exporting a raster, *Raster creation options* allows you to define additional options that control how the output file is structured and compressed. See more at *Raster driver options*.

> **Truco**
>
> Select an empty entry if you want to create your own custom combination of parameters.

Definir la configuración de transformación

Hay varias opciones que deben definirse para el ráster de salida georreferenciado.

- La casilla de verificación *Crear archivo mundial* solo está disponible si decide utilizar el tipo de transformación lineal, porque esto significa que la imagen ráster en realidad no se transformará. En este caso, el campo *Ráster de salida* no está activado, porque solo se creará un nuevo archivo mundial.
- Para todos los demás tipos de transformación, debe definir un *ráster de salida*. De forma predeterminada, se creará un nuevo archivo ([filename] _modified) en la misma carpeta junto con la imagen ráster original.
- El siguiente paso es definir el *SRC de destino para el ráster georreferenciado (véase :ref:`label_projections*).
- Si lo desea, puede **generar un mapa en pdf** y también **un informe en pdf**. El informe incluye información sobre los parámetros de transformación utilizados, una imagen de los residuos y una lista con todos los GCP y sus errores de RMS.
- Además, puede activar la casilla de verificación *Establecer resolución de destino* y defina la resolución de píxeles del ráster de salida. La resolución horizontal y vertical predeterminada es 1.
- La *Usar 0 para transparencia cuando sea necesario* se puede activar, si los píxeles con el valor 0 deben visualizarse transparentes. En nuestra hoja superior de ejemplo, todas las áreas blancas serían transparentes.
- La casilla *Guardar Puntos PCT* almacenará los Puntos PCT en un archivo junto al ráster de salida.
- Por último, *Cargar en proyecto al terminar* carga el ráster de salida automáticamente en el lienzo de mapa de QGIS cuando la transformación está terminada.

Ejecutando la transformación

Una vez que se hayan recopilado todos los GCP y se hayan definido todas las configuraciones de transformación, simplemente presione el botón Iniciar georreferenciación para crear el nuevo ráster georreferenciado.

11.3.2 Capa vectorial de georreferenciación

La georreferenciación de capas vectoriales funciona de forma similar a la georreferenciación de rásters, pero en lugar de hacer coincidir píxeles de imagen, se hacen coincidir geometrías vectoriales (puntos, líneas o polígonos) con referencias espaciales conocidas.

El procedimiento estándar es el mismo que para la georreferenciación ráster: abra QGIS y añada una capa al lienzo del mapa para utilizarla como referencia. Puede ser una capa ráster o vectorial georreferenciada, o una capa WMS.

Abra el diálogo Georreferenciador desde *Capa* ► *Georreferenciador*.

Inicie la georreferenciación siguiendo estos pasos (en este ejemplo, utilizamos un `alaska.shp` sin referenciar):

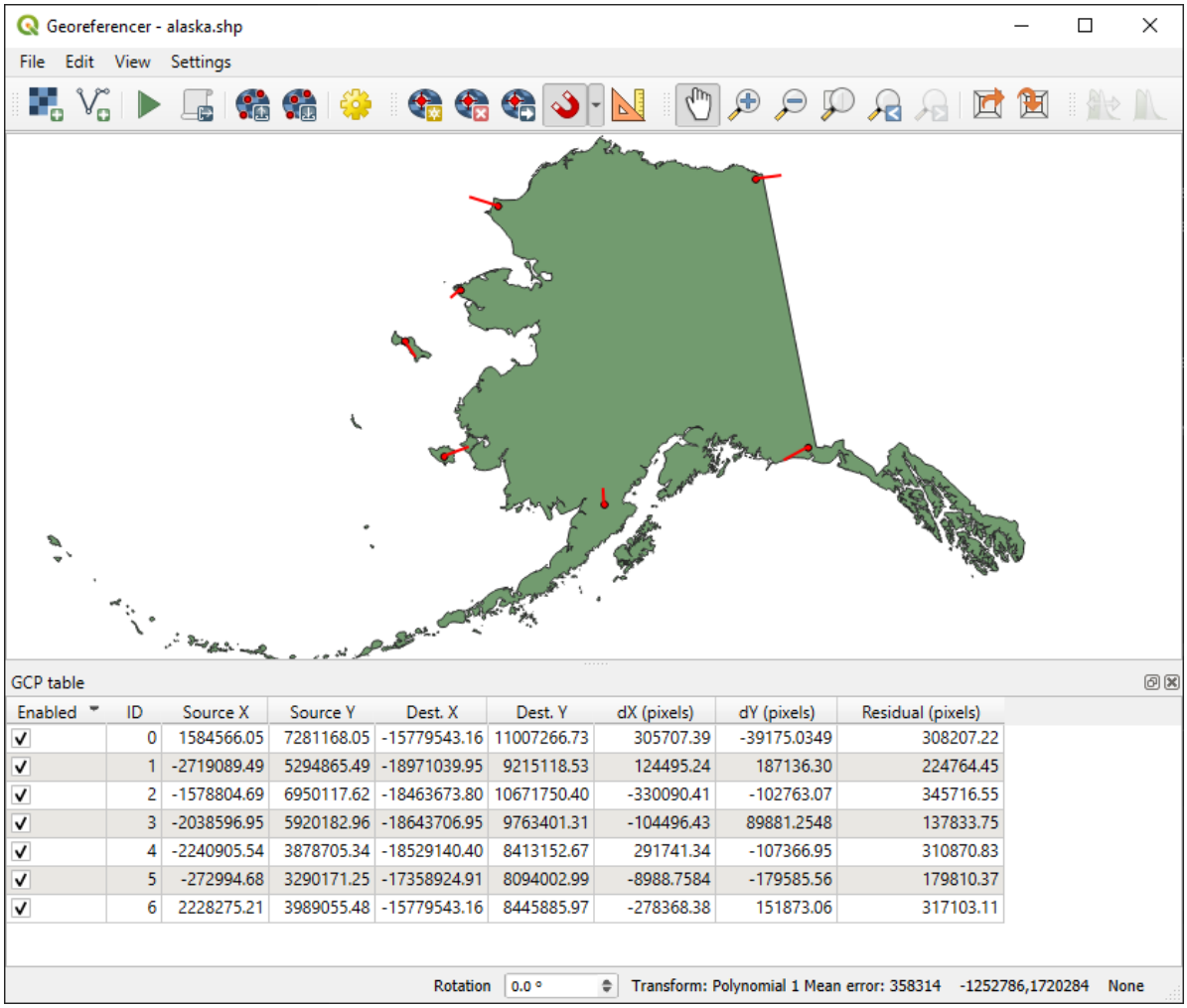


Figura 11.41: Diálogo del georreferenciador vectorial

1. Cargue la capa vectorial no referenciada utilizando el botón . La capa vectorial aparecerá en el área de trabajo principal del diálogo.

2. Activate snapping by clicking the button and selecting the desired snapping type(s). This enables snapping to the reference layer when placing GCPs.

You can also use the  Advanced Digitizing dock to ensure high-precision point selection. For more information, refer to the *Advanced Digitizing* section.

3. Utilice el botón  Añadir punto PCT para añadir un punto al área de trabajo. Introduzca sus coordenadas manualmente y establezca el SRC, o haga clic en el botón  Desde el lienzo del mapa para seleccionar las coordenadas de una capa georreferenciada en el lienzo principal del mapa de QGIS. En ese caso, el SRC se establecerá automáticamente.
4. Definir los ajustes de transformación:
   - Seleccione la *Tipo de transformación*. Los algoritmos de transformación son los mismos que para la georreferenciación ráster. Véase *Definiendo la configuración de transformación* para más detalles.
   - Define el *SRC de destino* (Sistema de Referencia de Coordenadas) para el vector georreferenciado (véase *Trabajar con Proyecciones*).
   - Establezca el formato y la ruta del archivo de salida (por ejemplo, GeoPackage, Shapefile). Por defecto, se creará un nuevo archivo con el sufijo `_modified` en la misma carpeta que el archivo vectorial original.
   - Opcionalmente, active **Generar mapa PDF** y **Generar informe PDF**. El informe incluye los parámetros de transformación, los residuos PCT y un resumen de los errores RMS.
   - Habilite  *Guardar Puntos PCT* para almacenar los PCTs en un archivo junto a la capa vectorial de salida.
   - Active  *Cargar en proyecto al terminar* para añadir el resultado directamente al lienzo del mapa.
5. Haga clic en  Iniciar georreferenciación para ejecutar la transformación y generar la capa vectorial georreferenciada.

11.3.3 Mostrar y adaptar las propiedades de las capas

Al hacer clic en la opción *Propiedades de origen* del menú *Configuración* se abre la *Propiedades de la capa raster* o *Propiedades de capa vectorial* dependiendo del tipo de capa que esté georreferenciando.

11.3.4 Configurar el georreferenciador

Puede personalizar el comportamiento del georreferenciador en *Configuración* ► *Configurar georreferenciador* (o utilice el atajo de teclado `Ctrl+P`).

- En *Consejo de punto* puede utilizar las casillas de verificación para alternar la visualización de los ID de PCT y las coordenadas X/Y tanto en la ventana del Georreferenciador como en el lienzo principal del mapa.
- *Unidades residuales* controla si las unidades residuales se dan en píxeles o en unidades de mapa
- *Informe PDF* permite establecer el tamaño del margen en mm para la exportación del informe
- *Mapa PDF* le permite elegir un tamaño de papel para la exportación del mapa
- Por último, puede activar la casilla *Mostrar ventana del Georreferenciador acoplada*. Esto anclará la ventana del Georreferenciador en la ventana principal de QGIS en lugar de mostrarla como una ventana separada que puede ser minimizada.

11.4 Explorando campos y formatos de datos

11.4.1 Datos Raster

Los datos ráster GIS son matrices de celdas discretas que representan entidades/fenómenos sobre, encima o debajo de la superficie terrestre. Cada celda de la cuadrícula ráster tiene el mismo tamaño y las celdas suelen ser rectangulares (en QGIS siempre serán rectangulares). Los datasets ráster típicos incluyen datos de teledetección, como fotografías aéreas, o imágenes de satélite y datos modelados, como elevación o temperatura.

A diferencia de los datos vectoriales, los datos ráster normalmente no tienen un registro de base de datos asociado para cada celda. Están geocodificados por resolución de píxeles y la coordenada X / Y de un píxel de esquina de la capa ráster. Esto permite que QGIS coloque los datos correctamente en el lienzo del mapa.

El formato GeoPackage es conveniente para almacenar datos ráster cuando se trabaja con QGIS. El popular y poderoso formato GeoTiff es una buena alternativa.

QGIS utiliza información de georeferencia dentro de la capa ráster (por ejemplo, GeoTiff) o un *archivo mundial* asociado para mostrar los datos correctamente.

11.4.2 Datos Vectoriales

Many of the features and tools available in QGIS work the same, regardless of the vector data source. However, because of the differences in format specifications (GeoPackage, ESRI Shapefile, MapInfo and MicroStation file formats, AutoCAD DXF, PostgreSQL, SpatiaLite, Oracle Spatial, MS SQL Server, SAP HANA Spatial databases and many more), QGIS may handle some of their properties differently. Support is provided by the GDAL vector drivers. This section describes how to work with some of these specifics.

> **Nota**
>
> QGIS admite tipos de entidades (multi)punto, (multi)línea, (multi)polígono, cadena circular, Curva Compuesta, Curva Poligonal, MultiCurva, MultiSuperficie, todos opcionalmente con valores Z y/o M.
>
> También debe tener en cuenta que algunos controladores no admiten algunos de estos tipos de entidades, como Cadena Circular, Curva Compuesta, Curva Poligonal, MultiCurva, MultiSuperficie. QGIS los convertirá.

GeoPackage

El formato GeoPackage (GPKG) es independiente de la plataforma, se implementa como un contenedor de base de datos SQLite y se puede utilizar para almacenar datos vectoriales y ráster. El formato fue definido por el Consorcio Geoespacial Abierto (OGC) y fue publicado en 2014.

GeoPackage puede ser usado para almacenar lo siguiente en una base de datos SQLite:

- entidades **vectoriales**
- **conjuntos de matrices de teselas de imágenes** y mapas **ráster**
- atributos (datos no espaciales)
- extensiones

Desde la versión 3.8 de QGIS, GeoPackage también puede almacenar proyectos de QGIS. Las capas GeoPackage pueden tener campos JSON.

GeoPackage es el formato de datos predeterminado para datos vectoriales en QGIS.

Formato de archivo de forma ESRI

El formato de archivo de forma ESRI sigue siendo uno de los formatos de archivo vectorial más utilizados, incluso si tiene algunas limitaciones en comparación con, por ejemplo, GeoPackage y SpatiaLite.

Un conjunto de datos en formato de archivo de forma ESRI consta de varios archivos. Los tres siguientes son requeridos:

1. archivo contenedor de las geometrías de las entidades `.shp`
2. archivo contenedor de los atributos en formato dBase `.dbf`
3. archivo del índice `.shx`

Un conjunto de datos del formato ESRI Shapefile puede también incluir un archivo con una extensión `.prj`, que contiene información de proyección. Aunque es muy útil tener un archivo de proyección, no es obligatorio. Un conjunto de datos del formato Shapefile puede contener archivos adicionales. Para más detalles vea la **`especificación técnica ESRI `<https://www.esri.com/content/dam/esrisites/sitecore-archive/Files/Pdfs/library/whitepapers/pdfs/shapefile.pdf>`_**.

GDAL tiene soporte de lectura-escritura para el formato comprimido ESRI Shapefile (`shz` y `shp.zip`).

Mejora del rendimiento para conjuntos de datos en formato ESRI Shapefile

Para mejorar el rendimiento del dibujo para un dataset de formato ESRI Shapefile, puede crear un índice espacial. Un índice espacial mejorará la velocidad tanto del zoom como de la panorámica. Los índices espaciales utilizados por QGIS tienen una extensión `.qix`.

Usa estos pasos para crear un índice:

1. Carga un conjunto de datos de formato de archivo de forma ESRI (ver *El panel Navegador*)
2. Abre el diálogo *Propiedades de capa* mediante doble-click en el nombre de la capa en la leyenda o con click derecho y eligiendo *Propiedades…* del menú contextual
3. En la pestaña *Fuente*, click en el botón *Crear índice Espacial*

Problema cargando un archivo .prj

Si carga un conjunto de datos en formato ESRI Shapefile con un archivo `.prj` y QGIS no puede leer el sistema de referencia de coordenadas de ese archivo, deberá definir la proyección adecuada manualmente en :menuselection:`

Propiedades de capa -> Pestaña Fuente` de la capa haciendo click en el botón  Seleccionar CRS. Esto se debe al hecho de que los archivos `.prj` a menudo no proporcionan los parámetros de proyección completos que se utilizan en QGIS y que se enumeran en el diálogo *CRS*.

Por la misma razón, si crea un nuevo conjunto de datos en formato ESRI Shapefile con QGIS, se crean dos archivos de proyección diferentes: un archivo `.prj` con parámetros de proyección limitados, compatible con el software ESRI, y un archivo `. qpj`, que proporciona todos los parámetros del CRS. Siempre que QGIS encuentre un archivo `.qpj`, se usará en lugar del archivo `.prj`.

Archivos de Texto delimitado

Los archivos de texto delimitados son muy comunes y se usan ampliamente debido a su simplicidad y legibilidad: los datos se pueden ver y editar en un editor de texto sin formato. Un archivo de texto delimitado son datos tabulares con columnas separadas por un carácter definido y filas separadas por saltos de línea. La primera fila generalmente contiene los nombres de las columnas. Un tipo común de archivo de texto delimitado es un CSV (valores separados por comas), con columnas separadas por comas. Los archivos de texto delimitados también pueden contener información posicional (ver *Almacenamiento de información de geometría en archivos de texto delimitados*).

QGIS le permite cargar un archivo de texto delimitado como una capa o una tabla ordinaria (ver *El panel Navegador* o *Importando un archivo de texto delimitado*). Primero verifique que el archivo cumpla con los siguientes requisitos:

1. El archivo debe tener una fila de encabezado delimitada de nombres de campo. Esta debe ser la primera línea de los datos (idealmente la primera fila en el archivo de texto).

2. Si se debe habilitar la geometría, el archivo debe contener los campos que definen la geometría. Estos campos pueden tener cualquier nombre.
3. Los campos de coordenadas X e Y (si la geometría está definida por coordenadas) deben especificarse como números. El sistema de coordenadas no es importante.
4. Si tiene un archivo CSV con columnas que no sean cadenas, puede tener un archivo CSVT adjunto (véase la sección *Usar archivo CSVT para controlar el formato de campo*).

El archivo de datos de puntos de elevación `elevp.csv` en el conjunto de datos de muestra de QGIS (ver sección :ref:` label_sampledata`) es un ejemplo de un archivo de texto válido:

```
X;Y;ELEV
-300120;7689960;13
-654360;7562040;52
1640;7512840;3
[...]
```

Algunas cosas a tener en cuenta sobre el archivo de texto:

1. El archivo de texto de ejemplo usa `;` (punto y coma) como delimitador (se puede usar cualquier carácter para delimitar los campos).
2. La primera fila es la fila de encabezado. Contiene los campos `X`, `` Y `` y `` ELEV ``.
3. No se utilizan comillas (`` « ``) para delimitar campos de texto
4. Las coordenadas X están contenidas en el campo `X`
5. Las coordenadas Y están contenidas en el campo `Y`

Almacenamiento de información de geometría en archivos de texto delimitados

Los archivos de texto delimitados pueden contener información de geometría en dos formas principales:

- Como coordenadas en columnas separadas (por ejemplo, `Xcol`, `` Ycol `` …), para datos de geometría de puntos;
- Como representación de texto conocido (WKT) de geometría en una sola columna, para cualquier tipo de geometría.

Se admiten las funciones con geometrías curvas (Cadena circular, PolígonoCurvo y CurvaCompuesta). A continuación, se muestran algunos ejemplos de tipos de geometría en un archivo de texto delimitado con geometrías codificadas como WKT:

```
Label;WKT_geom
LineString;LINESTRING(10.0 20.0, 11.0 21.0, 13.0 25.5)
CircularString;CIRCULARSTRING(268 415,227 505,227 406)
CurvePolygon;CURVEPOLYGON(CIRCULARSTRING(1 3, 3 5, 4 7, 7 3, 1 3))
CompoundCurve;COMPOUNDCURVE((5 3, 5 13), CIRCULARSTRING(5 13, 7 15,
  9 13), (9 13, 9 3), CIRCULARSTRING(9 3, 7 1, 5 3))
```

Los archivos de texto delimitados también admiten coordenadas Z y M en geometrías:

```
LINESTRINGZ(10.0 20.0 30.0, 11.0 21.0 31.0, 11.0 22.0 30.0)
```

Usar archivo CSVT para controlar el formato de campo

Al cargar archivos CSV, el controlador GDAL asume que todos los campos son cadenas (es decir, texto) a menos que se le indique lo contrario. Puedes crear un archivo CSVT para indicar a GDAL (y a QGIS) el tipo de datos de las diferentes columnas:

Tipo	Nombre	Ejemplo
Número entero	Entero	4
Boolean	Integer(Boolean)	true
Número decimal	Real	3.456
Fecha	Date (YYYY-MM-DD)	2016-07-28
Tiempo	Time (HH:MM:SS+nn)	18:33:12+00
Fecha y hora	DateTime (YYYY-MM-DD HH:MM:SS+nn)	2016-07-28 18:33:12+00
CoordX	CoordX	8.8249
CoordY	CoordY	47.2274
Point(X)	Point(X)	8.8249
Point(Y)	Point(Y)	47.2274
WKT	WKT	POINT(15 20)

El archivo CSVT es un archivo de texto sin formato de ** UNA línea ** con los tipos de datos entre comillas y separados por comas, por ejemplo

```
"Integer","Real","String"
```

Puedes incluso especificar anchura y precisión para cada columna, p.ej.:

```
"Integer(6)","Real(5.5)","String(22)"
```

Este archivo es guardado en la misma carpeta que el archivo `.csv`, con el mismo nombre, pero con extensión `csvt`.

Puede encontrar más información en GDAL CSV Driver.

Truco

Detectar tipos de campo

En lugar de utilizar un archivo CSVT para indicar los tipos de datos, QGIS ofrece la posibilidad de detectar automáticamente los tipos de campo y cambiar los tipos de campo asumidos.

PostgreSQL Database

In order to process geographical information in a PostgreSQL database, the PostGIS extension must first be installed. PostGIS extends the capabilities of the PostgreSQL relational database by adding support for storing, indexing, and querying geospatial data.

To enable PostGIS in your database you have to activate the extension in your database (open database and run `CREATE EXTENSION postgis;`). PostGIS enabled databases are also often named «PostGIS layer». Using PostGIS, vector functions such as select and identify work more accurately than they do with GDAL layers in QGIS.

Truco

Capas PostGIS

Normalmente, una capa PostGIS se identifica mediante una entrada en la tabla geometry_columns. QGIS puede cargar capas que no tienen una entrada en la tabla geometry_columns. Esto incluye tanto tablas como vistas. Consulte su manual de PostgreSQL para obtener información sobre cómo crear vistas.

Esta sección contiene algunos detalles sobre cómo QGIS accede a las capas de PostgreSQL. La mayoría de las veces, QGIS simplemente debería proporcionarle una lista de tablas de base de datos que se pueden cargar, y las cargará a pedido. Sin embargo, si tiene problemas para cargar una tabla de PostgreSQL en QGIS, la siguiente información puede ayudarlo a comprender los mensajes de QGIS y brindarle instrucciones para modificar la tabla de PostgreSQL o la definición de vista para permitir que QGIS la cargue.

Nota

Una base de datos PostgreSQL también puede almacenar proyectos QGIS.

Clave primaria

QGIS requiere que las capas de PostgreSQL contengan una columna que se pueda utilizar como clave única para la capa. Para las tablas, esto generalmente significa que la tabla necesita una clave principal o una columna con una restricción única. En QGIS, esta columna debe ser de tipo int4 (un número entero de 4 bytes). Alternativamente, la columna ctid se puede utilizar como clave principal. Si una tabla carece de estos elementos, se utilizará la columna oid en su lugar. El rendimiento mejorará si la columna está indexada (tenga en cuenta que las claves principales se indexan automáticamente en PostgreSQL).

QGIS ofrece una casilla de verificación **Seleccionar en id** que está activada de forma predeterminada. Esta opción obtiene los identificadores sin los atributos, lo que es más rápido en la mayoría de los casos.

Ver

Si la capa de PostgreSQL es una vista, existe el mismo requisito, pero las vistas no siempre tienen claves primarias o columnas con restricciones únicas. Debe definir un campo de clave principal (debe ser un número entero) en el cuadro de diálogo de QGIS antes de poder cargar la vista. Si no existe una columna adecuada en la vista, QGIS no cargará la capa. Si esto ocurre, la solución es modificar la vista para que incluya una columna adecuada (un tipo de número entero y una clave primaria o con una restricción única, preferiblemente indexada).

En cuanto a la tabla, una casilla de verificación **Seleccionar en id** está activada de forma predeterminada (consulte más arriba el significado de la casilla de verificación). Puede tener sentido desactivar esta opción cuando utiliza vistas caras.

Nota

Tabla extrerna de PostgreSQL

Las tablas externas de PostgreSQL no son compatibles explícitamente con el proveedor de PostgreSQL y se manejarán como una vista.

Copia de seguridad de base de datos y tabla de estilo de capa de QGIS

If you want to make a backup of your PostgreSQL database using the `pg_dump` and `pg_restore` commands, and the default layer styles as saved by QGIS fail to restore afterwards, you need to set the XML option to `DOCUMENT` before the restore command:

1. Hacer una copia de seguridad PLANA de la tabla *layer_style*`
2. Abrir el archivo en un editor de texto
3. Cambiar la línea `SET xmloption = content;` a `SET XML OPTION DOCUMENT;`
4. Guardar el archivo
5. Utilice psql para restaurar la table en la nueva base de datos

Filtrar el lado de la base de datos

QGIS permite filtrar funciones que ya están en el lado del servidor. Compruebe *Configuración -> Opciones -> Fuentes de datos ->*  *Ejecute expresiones en el lado del servidor si es posible* para hacerlo. Solo las expresiones admitidas se enviarán a la base de datos. Las expresiones que utilicen operadores o funciones no admitidas recurrirán con gracia a la evaluación local.

Tipos de archivos soportados por PostgreSQL

Los tipos de datos admitidos por el proveedor de PostgreSQL incluyen: integer, float, boolean, binary object, varchar, geometry, timestamp, array, hstore y json.

Importando Datos en PostgreSQL

Data can be imported into PostgreSQL using several tools, including the Browser, DB Manager plugin and the command line tools shp2pgsql and ogr2ogr.

Administrador de BBDD

QGIS viene con un complemento principal llamado  Administrador de bases de datos. Se puede usar para cargar datos e incluye soporte para esquemas. Consulte la sección *Complemento de Administración de BBDD* para obtener más información.

shp2pgsql

PostGIS incluye una utilidad llamada **shp2pgsql**, que se puede utilizar para importar conjuntos de datos en formato Shapefile a una base de datos habilitada para PostGIS. Por ejemplo, para importar un conjunto de datos en formato Shapefile llamado `lakes.shp` a una base de datos PostgreSQL llamada `gis_data`, use el siguiente comando:

```
shp2pgsql -s 2964 lakes.shp lakes_new | psql gis_data
```

Esto crea una nueva capa llamada `lakes_new` en la base de datos `gis_data`. La nueva capa tendrá un identificador de referencia espacial (SRID) de 2964. Consulte la sección *Trabajar con Proyecciones* para obtener más información sobre los sistemas de referencia espacial y las proyecciones.

> **Truco**
>
> **Exporting datasets from PostgreSQL**
>
> There is also a tool for exporting postgreSQL datasets wth geographic data to Shapefile format: **pgsql2shp**. It is shipped within your PostGIS distribution.

ogr2ogr

In addition to **shp2pgsql** and **DB Manager**, there is another tool for feeding geographical data in PostgreSQL: **ogr2ogr**. It is part of your GDAL installation.

To import a Shapefile format dataset into PostgreSQL, do the following:

```
ogr2ogr -f "PostgreSQL" PG:"dbname=postgis host=myhost.de user=postgres
password=topsecret" alaska.shp
```

This will import the Shapefile format dataset `alaska.shp` into the database *postgis* using the user *postgres* with the password *topsecret* on the host server *myhost.de*.

Note that GDAL must be built with PostgreSQL. You can verify this by typing (in ):

```
ogrinfo --formats | grep -i post
```

Si prefiere utilizar el comando **COPY** de PostgreSQL en lugar del método **INSERT INTO** predeterminado, puede exportar la siguiente variable de entorno (al menos disponible en  y ):

```
export PG_USE_COPY=YES
```

ogr2ogr no crea índices espaciales como lo hace **shp2pgsl**. Debe crearlos manualmente, utilizando el comando SQL normal ** CREATE INDEX ** después, como un paso adicional (como se describe en la siguiente sección *Mejorar el rendimiento*).

Mejorar el rendimiento

Retrieving features from a PostgreSQL database can be time-consuming, especially over a network. You can improve the drawing performance of PostgreSQL layers by ensuring that a spatial index exists on each layer in the database. PostGIS supports creation of a GiST (Generalized Search Tree) index to speed up spatial searching (GiST index information is taken from the PostGIS documentation available at https://postgis.net).

Truco

Puede utilizar Administrador de BBDD para crear un índice para su capa. Primero debe seleccionar la capa y hacer clic en *Tabla-> Editar tabla*, ir a la pestaña :menuselection:` Indexes` y hacer click en *Añadir índice Espacial*.

La sintaxis para crear un índice GiST es:

```
CREATE INDEX [indexname] ON [tablename]
  USING GIST ( [geometryfield] GIST_GEOMETRY_OPS );
```

Tenga en cuenta que para tablas grandes, la creación del índice puede llevar mucho tiempo. Una vez creado el índice, debe realizar un `` ANÁLISIS DE VACÍO ``. Consulte la documentación de PostGIS (POSTGIS-PROJECT en literatura_y_web) para obtener más información.

El siguiente ejemplo crea un índice GiST:

```
gsherman@madison:~/current$ psql gis_data
Welcome to psql 8.3.0, the PostgreSQL interactive terminal.

Type:  \copyright for distribution terms
       \h for help with SQL commands
       \? for help with psql commands
       \g or terminate with semicolon to execute query
       \q to quit
```

(continúe en la próxima página)

(proviene de la página anterior)

```
gis_data=# CREATE INDEX sidx_alaska_lakes ON alaska_lakes
gis_data-# USING GIST (geom GIST_GEOMETRY_OPS);
CREATE INDEX
gis_data=# VACUUM ANALYZE alaska_lakes;
VACUUM
gis_data=# \q
gsherman@madison:~/current$
```

Capas SpatiaLite

Si desea guardar una capa vectorial usando el formato SpatiaLite, puede hacerlo siguiendo las instrucciones en *Creando nuevas capas desde una capa existente*. Seleccione `` SpatiaLite`` como *Formato* e ingrese ambos :guilabel:` Nombre de archivo` y *Nombre de capa*.

También puede seleccionar `SQLite` como formato y añadir `SPATIALITE=YES` en el campo *Opciones personalizadas* ► *Origen de datos*. Esto le dice a GDAL que cree una base de datos SpatiaLite. Véase también https://gdal.org/en/latest/drivers/vector/sqlite.html.

QGIS también admite vistas editables en SpatiaLite. Para la gestión de datos de SpatiaLite, también puede utilizar el complemento principal *DB Manager*.

Si desea crear una nueva capa de SpatiaLite, consulte la sección *Crear una nueva capa SpatiaLite*.

parámetros específicos GeoJSON

Cuando *exporting layers* a GeoJSON, hay algunas *Opciones de Capa* disponibles. Estas opciones provienen de GDAL que es responsable de la redacción del archivo:

- *COORDINATE_PRECISION* el número máximo de dígitos después del separador decimal para escribir en coordenadas. El valor predeterminado es 15 (nota: para las coordenadas Lat Lon, 6 se considera suficiente). Se producirá un truncamiento para eliminar los ceros finales.
- *RFC7946* por defecto se utilizará GeoJSON 2008. Si se establece en YES, se utilizará el estándar actualizado RFC 7946. Por defecto es NO (por tanto GeoJSON 2008). Véase https://gdal.org/en/latest/drivers/vector/geojson.html#rfc-7946-write-support para las principales diferencias, en resumen: sólo se permite EPSG:4326, otros crs serán transformados, los polígonos se escribirán de forma que sigan la regla de la mano derecha para la orientación, los valores de una matriz «bbox» son [west, south, east, north], no [minx, miny, maxx, maxy]. Algunos nombres de miembros de extensión están prohibidos en los objetos FeatureCollection, Feature y Geometry, la precisión de coordenadas por defecto es de 7 dígitos decimales
- *WRITE_BBOX* establécelo en SÍ para incluir el cuadro delimitador de las geometrías en el nivel de entidad y colección de entidades

Además de GeoJSON también hay una opción para exportar a «GeoJSON - Newline Delimited» (ver https://gdal.org/en/latest/drivers/vector/geojsonseq.html). En lugar de una FeatureCollection con Features, puede transmitir un tipo (probablemente sólo Features) secuencialmente separado con nuevas líneas.

GeoJSON - Newline Delimited también tiene algunas opciones de capa específicas disponibles:

- *COORDINATE_PRECISION* ver arriba (igual que para GeoJSON)
- *RS* al iniciar registros con el carácter RS = 0x1E. La diferencia es cómo se separan las entidades: solo por un carácter de nueva línea (LF) (JSON delimitado por nueva línea, geojsonl) o también anteponiendo un carácter separador de registros (RS) (dando GeoJSON Text Sequences, geojsons). Por defecto a NO. Los archivos reciben la extensión `.json` si no se proporciona la extensión.

Capas espaciales de SAP HANA

Esta sección contiene algunos detalles sobre cómo QGIS accede a las capas de SAP HANA. La mayoría de las veces, QGIS simplemente debería proporcionarle una lista de tablas y vistas de la base de datos que se pueden cargar, y las cargará a pedido. Sin embargo, si tiene problemas para cargar una tabla o vista de SAP HANA en QGIS, la siguiente información puede ayudarlo a comprender la causa raíz y ayudarlo a resolver el problema.

Identificación de características

Si desea utilizar todas las capacidades de edición de características de QGIS, QGIS debe poder identificar sin ambigüedades cada característica en una capa. Internamente, QGIS usa un entero con signo de 64 bits para identificar características, mientras que el rango negativo está reservado para propósitos especiales.

Por lo tanto, el proveedor de SAP HANA requiere una clave única que se pueda asignar a un número entero positivo de 64 bits para admitir completamente las capacidades de edición de características de QGIS. Si no es posible crear dicha asignación, aún puede ver las características, pero es posible que la edición no funcione.

Agregar tablas

Al agregar una tabla como capa, el proveedor de SAP HANA usa la clave principal de la tabla para asignarla a una identificación de función única. Por lo tanto, para tener compatibilidad completa con la edición de funciones, debe tener una clave principal para la definición de su tabla.

El proveedor de SAP HANA admite claves principales de varias columnas, pero si desea obtener el mejor rendimiento, su clave principal debe ser una sola columna de tipo `INTEGER`.

Agregar vistas

Al añadir una vista como capa, el proveedor de SAP HANA no puede identificar automáticamente las columnas que identifican inequívocamente un objeto espacial. Además, algunas vistas son de solo lectura y no se pueden editar.

Para tener soporte completo de edición de características, la vista debe ser actualizable (marque la columna `IS_READ_ONLY` en la vista del sistema `SYS.VIEWS` para la vista en cuestión) y debe proporcionar manualmente a QGIS una o más columnas que identifiquen un rasgo. Las columnas se pueden proporcionar usando *Layer* ► *Add Layer* ► *Add SAP HANA Spatial Layer* y luego seleccionando las columnas en la columna *Feature id*. Para un mejor rendimiento, el valor *Feature id* debe ser una única columna `INTEGER`.

11.4.3 Capas que cruzan los 180° de longitud

Muchos paquetes SIG no envuelven las capas con un sistema de referencia geográfica (lat/lon) que cruza la línea de longitud de 180 grados. Como resultado, si abrimos una capa de este tipo en QGIS, podríamos ver dos ubicaciones muy separadas, que deberían aparecer cerca la una de la otra. En Figura 11.42, el pequeño punto en el extremo izquierdo del lienzo del mapa (Islas Chatham) debería estar dentro de la cuadrícula, a la derecha de las islas principales de Nueva Zelanda.



Figura 11.42: Mapa en lat/lon cruzando la línea de longitud de 180

Solving in PostgreSQL

A work-around is to transform the longitude values using PostgreSQL and the ST_ShiftLongitude function. This function reads every point/vertex in every component of every feature in a geometry, and shifts its longitude coordinate from -180..0° to 180..360° and vice versa if between these ranges. This function is symmetrical so the result is a 0..360° representation of a -180..180° data and a -180..180° representation of a 0..360° data.

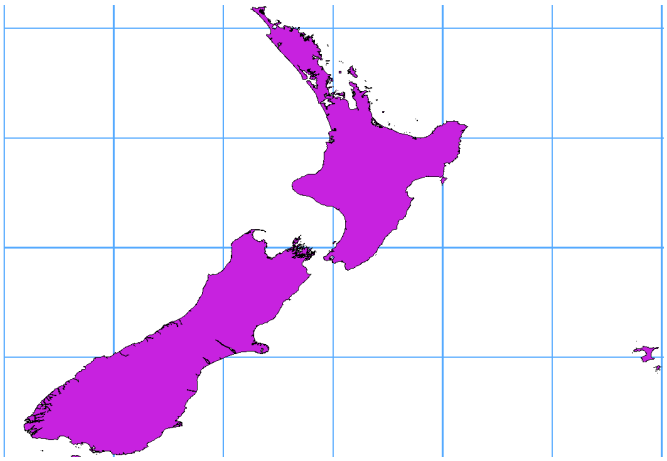


Figura 11.43: Cruce de 180° de longitud aplicando la función **ST_ShiftLongitude**.

1. Import data into PostgreSQL (*Importando Datos en PostgreSQL*) using, for example, the DB Manager plugin.
2. Use the PostgreSQL command line interface to issue the following command:

```
-- In this example, "TABLE" is the actual name of your PostgreSQL table
update TABLE set geom=ST_ShiftLongitude(geom);
```

3. Si todo salió bien, debería recibir una confirmación sobre la cantidad de funciones que se actualizaron. Entonces podrá cargar el mapa y ver la diferencia (*Figure_vector_crossing_map*).

CAPÍTULO 12

Trabajar con Datos Vectoriales

12.1 El Diálogo de las Propiedades del Vector

El cuadro de diálogo *Propiedades de la capa* para una capa vectorial proporciona una configuración general para administrar la apariencia de los elementos de la capa en el mapa (simbología, etiquetas, diagramas), interacción con el mouse (acciones, textos de aviso del mapa, diseño de formularios). También proporciona información de la capa.

Para acceder al diálogo:guilabel:*Propiedades de la capa*:

- En el panel *Capas* , doble-click en la capa o click-derecho y selecciona *Propiedades…* en el menú emergente;
- Vaya al menú *Capa* ► *Propiedades de capa…* cuando la capa este seleccionada.

El diálogo vectorial *Propiedades de capa* proporciona las siguientes secciones:

Information	*Source*	*Symbology* ([1])
Labels ([1])	*Mask* ([1])	*3D View* ([1])
Diagramas	*Fields*	*Attributes Form*
Uniones	*Auxiliary Storage*	*Actions*
Visualización	*Renderizando*	*Temporal*
Variables	*Elevación*	*Metadata*
Dependencies	*Leyenda*	*QGIS Server*
Digitizing	*External plugins* ([2]) tabs	

[1] Also available in the *Layer styling panel*
[2] *External plugins* you install can optionally add tabs to this dialog. Those are not presented in this document. Refer to their documentation.

Truco

Comparta propiedades completas o parciales de los estilos de capa

El menú *Style* en la parte inferior del cuadro de diálogo le permite importar o exportar estas o parte de estas propiedades desde / hacia varios destinos (archivo, portapapeles, base de datos). Ver *Administrando Estilos Personalizados*.

Nota

Debido a que las propiedades (simbología, etiqueta, acciones, valores predeterminados, formas …) de capas incrustadas (ver *Incorporación de capas de proyectos externos*) se extraen del archivo del proyecto original y para evitar cambios que puedan romper este comportamiento, el diálogo de propiedades de capa no está disponible para estas capas.

12.1.1 Propiedades de información

La pestaña  *Information* es de solo lectura y representa un lugar interesante para obtener rápidamente información resumida y metadatos para la capa actual. La información proporcionada es:

- general, como el nombre en el proyecto, la ruta de origen, la lista de archivos auxiliares, el tiempo y el tamaño del último guardado, el proveedor utilizado
- propiedades personalizadas, utilizadas para almacenar en el proyecto activo información adicional sobre la capa. Las propiedades personalizadas por defecto pueden incluir *notas de capa*, *legend widgets*, *variables de capa*, *propiedades del formulario*… Se pueden crear y gestionar más propiedades personalizadas utilizando PyQGIS, concretamente a través del método setCustomProperty().
- según el proveedor de la capa: formato de almacenamiento, tipo de geometría, codificación de la fuente de datos, extensión, recuento de características…
- el sistema de referencia de coordenadas: nombre, unidades, método, precisión, referencia (es decir, si es estático o dinámico)
- seleccionado de los *filled metadata*: acceso, extensiones, enlaces, contactos, historial…
- y relacionados con su geometría (extensión espacial, CRS…) o sus atributos (número de campos, características de cada uno…).

12.1.2 Propiedades de fuente

Use esta pestaña para definir configuraciones generales para la capa vectorial.

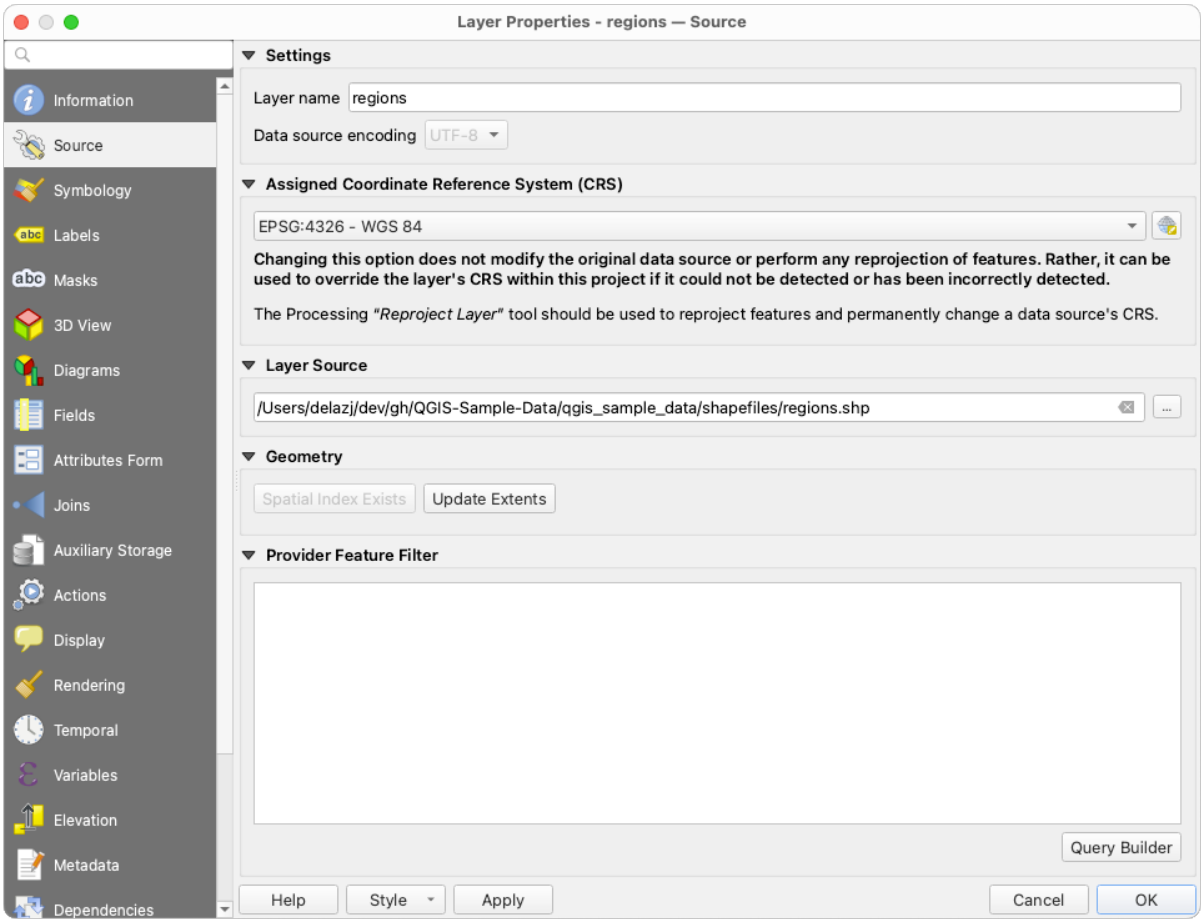


Figura 12.1: Pestaña fuente en el diálogo de propiedades de capa vectorial

Configuración

- Establezca un *Nombre de capa* diferente del nombre de archivo de la capa que se usará para identificar la capa en el proyecto (en el *Panel de capas*, con expresiones, en la leyenda del diseño de impresión, …)
- Dependiendo del formato de datos, seleccione *Codificación de fuente de datos* si QGIS no lo detecta correctamente.

Coordinate Reference System

- Muestra la capa *Assigned Coordinate Reference System (CRS)*. Puede cambiar el CRS de la capa, seleccionando uno usado recientemente en la lista desplegable o haciendo clic en el botón Select CRS (ver *Selector del Sistema de Coordenadas de Referencia*). Use este proceso solo si el CRS aplicado a la capa es incorrecto o si no se aplicó ninguno. Si desea reproyectar sus datos en otro CRS, utilice algoritmos de reproyección de capas de Processing o *Save it into another layer*.

Layer source and Geometry

- Depending on the data provider, a *Layer source* group indicates the path to the source of the dataset and allows for replacing the loaded layer:
  - When the layer is stored as file on disk, edit the path shown in the text box or press … Browse to select another file on the disk. Both layers do not need to share attribute fields, geometry type or file formats.
  - When the layer is provided by an ArcGIS Feature service, it is possible to modify its *authentication settings*, while keeping unchanged the details for connecting to the service.
- *Create spatial index* (only for OGR-supported formats): helps speeding layer rendering and features" geometry retrieval.
- *Actualizar extensión* información para una capa.

Constructor de Consulta

Se puede acceder al cuadro de diálogo Generador de consultas a través del botón *Constructor de consultas* situado en la parte inferior de la pestaña *Fuente* del cuadro de diálogo Propiedades de capa, en el grupo *Filtro de objeto espacial del proveedor*.

El generador de consultas proporciona una interfaz que le permite definir un subconjunto de las características de la capa utilizando una cláusula WHERE similar a SQL y mostrar el resultado en la ventana principal. Mientras la consulta esté activa, solo las características correspondientes a su resultado estarán disponibles en el proyecto.

Puede utilizar uno o más atributos de capa para definir el filtro en el `Generador de consultas`. El uso de más de un atributo se muestra en Figura 12.2. En el ejemplo, el filtro combina los atributos

- `toa` (`DateTime` campo: `cast("toa" as character) > '2017-05-17'` y``cast(«toa» as character) < "2019-12-24T18:00:00"```),
- `name` (`String` campo: `"name" > 'S'`) y
- `FID` (`Integer` campo: `FID > 10`)

utilizando los operadores y paréntesis AND, OR y NOT. Esta sintaxis (incluido el formato DateTime para el campo `toa`) funciona para los conjuntos de datos GeoPackage.

El filtro se realiza a nivel del proveedor de datos (OGR, PostgreSQL, MS SQL Server…). Por lo tanto, la sintaxis depende del proveedor de datos (por ejemplo, el formato ESRI Shapefile no admite DateTime). La expresión completa:

```
cast("toa" as character) > '2017-05-17' AND
cast("toa" as character) < '2019-12-24T18:00:00' AND
NOT ("name" > 'S' OR FID > 10)
```

También puede abrir el diálogo *Constructor de Consulta* usando la opción *Filtro…* desde el menú *Layer* o el menú contextual de la capa. Las secciones *Campos*, *Valores* y *Operadores* en el cuadro de diálogo le ayuda a construir la consulta tipo SQL expuesta en el cuadro *Proveedor de expresión de filtrado específico*.

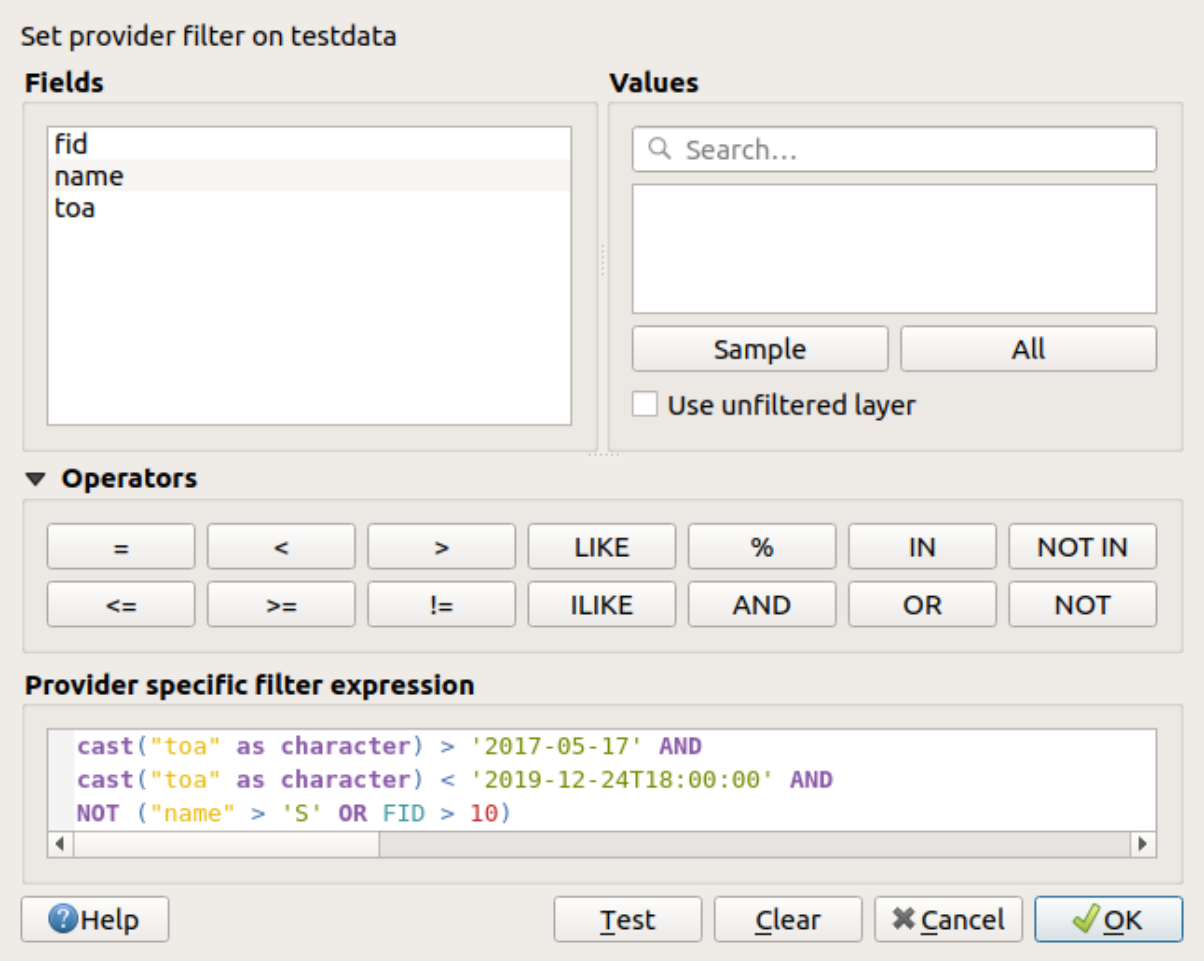


Figura 12.2: Constructor de Consulta

La lista **Campos** contiene todos los campos de la capa. Para agregar una columna de atributos al campo de la cláusula WHERE de SQL, haga doble clic en su nombre o simplemente escríbalo en el cuadro SQL.

El cuadro **Valores** enumera los valores del campo seleccionado actualmente. Para enumerar todos los valores únicos de un campo, haga clic en el botón: guilabel:*Todos*. Para enumerar los primeros 25 valores únicos de la columna, haga clic en el botón:guilabel:*Muestra*. Para agregar un valor al campo de la cláusula WHERE de SQL, haga doble clic en su nombre en la lista Valores. Puede usar el cuadro de búsqueda en la parte superior del marco de Valores para navegar fácilmente y encontrar valores de atributos en la lista.

La sección **Operadores** contiene todos los operadores utilizables. Para agregar un operador al campo de la cláusula SQL WHERE, haga clic en el botón correspondiente. Operadores relacionales (`` = , ``> ``, ...), operador de comparación de cadenas ( LIKE``) y operadores lógicos (`` AND``, `` OR``, …) están disponibles.

El botón *Probar* le ayuda a comprobar su consulta y muestra un cuadro de mensaje con el número de objetos espaciales que satisfacen la consulta actual. Utilice el botón *Limpiar* para borrar la consulta SQL y devolver la capa a su estado original (es decir, cargar completamente todas los objetos espaciales). Es posible *Guardar…* la consulta como un archivo `.QQF`, o *Cargar…* la consulta desde un archivo en el diálogo.

Cuando se aplica un filtro, QGIS trata el subconjunto resultante como si fuera la capa completa. Por ejemplo, si aplicó el filtro anterior para “Borough” (`` «TYPE_2» = “Borough”), no puede mostrar, consultar, guardar o editar `` Anchorage, porque ese es un “Municipio’y por lo tanto, no forma parte del subconjunto.

Truco

Las capas filtradas se indican en el Panel de capas

En el panel *Capas*, la capa filtrada aparece con un icono  Filtro al lado que indica la consulta utilizada cuando el ratón pasa por encima del botón. Al hacer doble clic en el icono se abre el cuadro de diálogo *Constructor de consultas* para su edición. También puede hacerse a través del menú *Capa* ► *Filtro…*.

12.1.3 Propiedades de simbología

La pestaña  *Simbología* le proporciona una herramienta completa para representar y simbolizar sus datos vectoriales. Puede utilizar herramientas comunes a todos los datos vectoriales, así como herramientas especiales de simbolización diseñadas para los distintos tipos de datos vectoriales. Sin embargo, todos los tipos comparten la siguiente estructura de diálogo: en la parte superior, tiene un widget que le ayuda a preparar la clasificación y el símbolo a utilizar para las características y en la parte inferior el widget *Reproducción de capas*.

Truco

Cambia rápidamente entre diferentes representaciones de capa

Usando el menú *Styles* ► *Add* en la parte inferior del diálogo *Layer Properties*, Puede guardar tantos estilos como sea necesario. Un estilo es la combinación de todas las propiedades de una capa (como simbología, etiquetado, diagrama, forma de campos, acciones …) como desee. Luego, simplemente cambie entre estilos desde el menú contextual de la capa en *Panel de capas* para obtener diferentes representaciones de sus datos automaticamente.

Truco

Exportar simbología vectorial

Tiene la opción de exportar simbología vectorial de QGIS a Google *.kml, *.dxf y archivos MapInfo *.tab. Simplemente abra el menú derecho del mouse de la capa y haga clic en *Save As…* para especificar el nombre del archivo de salida y su formato. En el cuadro de diálogo, use el menú *Symbology export* para guardar la simbología ya sea como *Feature symbology* ► o como *Symbol layer symbology* ►. Si ha usado capas de símbolos, se recomienda usar la segunda configuración.

Representación de elementos

El renderizador se encarga de dibujar un elemento junto con el símbolo correcto. Independientemente del tipo de geometría de la capa, existen cuatro tipos comunes de renderizadores: símbolo único, categorizado, graduado y basado en reglas. Para las capas de puntos, hay disponibles renderizadores de desplazamiento de puntos, cluster de puntos y mapa de calor, mientras que las capas de polígonos también pueden renderizarse con los renderizadores de características fusionadas, polígonos invertidos y 2,5 D.

No hay un renderizador de color continuo, porque de hecho es solo un caso especial del renderizador graduado. Los renderizadores categorizados y graduados se pueden crear especificando un símbolo y una rampa de color; establecerán los colores para los símbolos de manera adecuada. Para cada tipo de datos (puntos, líneas y polígonos), hay disponibles tipos de capas de símbolos vectoriales. Dependiendo del renderizador elegido, el diálogo proporciona diferentes secciones adicionales.

Nota

Si cambia el tipo de renderizador al establecer el estilo de una capa vectorial, se mantendrá la configuración que realizó para el símbolo. Tenga en cuenta que este procedimiento solo funciona para un cambio. Si repite el cambio del tipo de renderizador, la configuración del símbolo se perderá.

Representador de Símbolo Único

El renderizador de *Símbolo único* se usa para representar todas las entidades de la capa usando un solo símbolo definido por el usuario. Ver *EL selector de símbolo* para más información sobre la representación de símbolos.

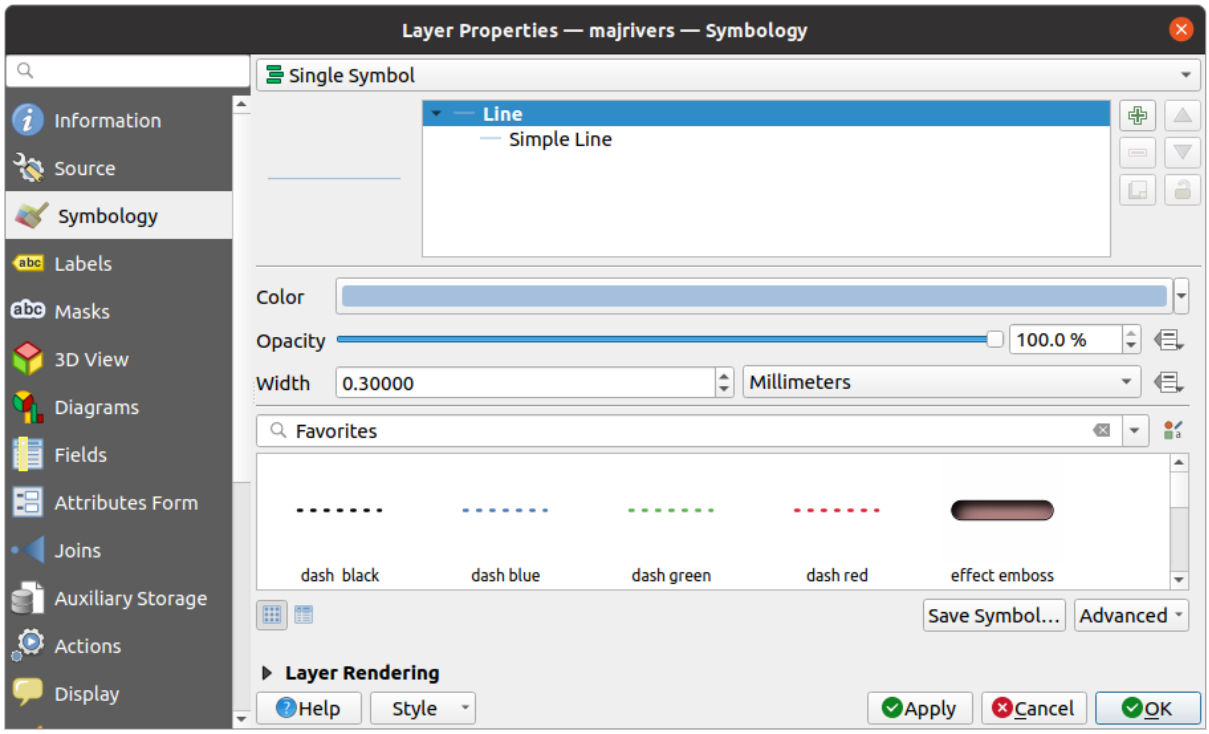


Figura 12.3: Propiedades de línea con símbolo único

Sin representación de símbolos

El renderizador *Sin Símbolos* es un caso de uso especial del renderizador Símbolo Único ya que aplica el mismo renderizado a todas las entidades. Con este renderizador, no se dibujará ningún símbolo para las entidades, pero se seguirán mostrando etiquetas, diagramas y otras partes que no sean símbolos.

Todavía se pueden hacer selecciones en la capa en el lienzo y las entidades seleccionadas se representarán con un símbolo predeterminado. También se mostrarán las entidades que se están editando.

Esto pretende ser un atajo útil para capas para las que solo desea mostrar etiquetas o diagramas, y evita la necesidad de representar símbolos con un relleno / borde totalmente transparente para lograr esto.

Representador Categorizado

El representador *Categorizado* se usa para representar las entidades de una capa, usando un símbolo definido por el usuario cuyo aspecto refleja los valores discretos de un campo o una expresión.

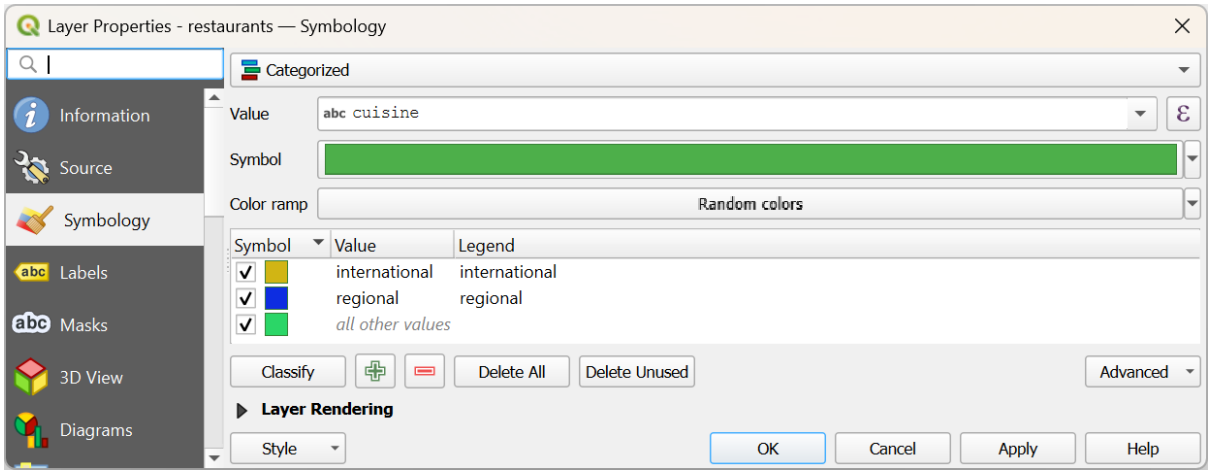


Figura 12.4: Opciones de simbolización categorizadas

Para usar la simbología categorizada para una capa:

1. Seleccione el *Valor* de clasificación: puede ser un campo existente o una *expresión* puede escribir en la casilla o construir utilizando el botón asociado . El uso de expresiones para categorizar evita la necesidad de crear un campo sólo para fines de simbología (por ejemplo, si sus criterios de clasificación se derivan de uno o más atributos).

   La expresión utilizada para clasificar características puede ser de cualquier tipo; por ejemplo, puede:

   - ser una comparación. En este caso, QGIS devuelve los valores `` 1`` (**Verdadero**) y `` 0`` (**Falso**). Algunos ejemplos:

     ```
     myfield >= 100
     $id = @atlas_featureid
     myfield % 2 = 0
     within( $geometry, @atlas_geometry )
     ```

   - combinar diferentes campos:

     ```
     concat( field_1, ' ', field_2 )
     ```

   - ser un cálculo en campos:

     ```
     myfield % 2
     year( myfield )
     field_1 + field_2
     substr( field_1, -3 )
     ```

   - ser usada para trasnformar valores lineales a clases discretas, p.ejemplo:

     ```
     CASE WHEN x > 1000 THEN 'Big' ELSE 'Small' END
     ```

   - combinar muchos valores discretos en una categoría simple, p.ejemplo:

     ```
     CASE
     WHEN building IN ('residence', 'mobile home') THEN 'residential'
     WHEN building IN ('commercial', 'industrial') THEN 'Commercial and␣
     ↪Industrial'
     END
     ```

> **Truco**
>
> Si bien puede usar cualquier tipo de expresión para categorizar características, para algunas expresiones complejas puede ser más sencillo de usar *rule-based rendering*.

2. Comfigure el *Symbol*, que se usará como símbolo base para todas las clases;
3. Indicar la *Rampa de colores*, i.e. la gama de colores de la que se selecciona el color aplicado a cada símbolo.

   Además de las opciones comunes de la *color ramp widget*, puede aplicar una *Random Color Ramp* a las categorías. Puede clickar en la entrada *Shuffle Random Colors* para regenerar un nuevo conjunto de colores aleatorios si no está satisfecho.
4. Luego clicke en el botón *Classify* para crear clases a partir de los distintos valores del campo o expresión proporcionados.
5. For each class, you can edit the *Legend* column to a more meaningful label (used in the *Layers* panel and the print layout).

   > **Nota**
   >
   > Keep in mind that some values may use widgets that do not display the actual value stored in the field. For example, a checkbox widget may store `1` and `0` for checked and unchecked states, while displaying `True` and `False` labels. In this case, to categorize features based on the checkbox state, you need to use the stored values (`1` and `0`) in the expression. QGIS will automatically use the display value for the legend column.

6. *Apply* los cambios ie la *live update* no está en uso y cada entidad en el lienzo del mapa se representará con el símbolo de su clase.

   Por defecto, QGIS agrega una clase:guilabel:*todos los demás valores* a la lista. Si bien está vacía al principio, esta clase se usa como una clase predeterminada para cualquier entidad que no se encuentre en las otras clases (por ejemplo, cuando crea entidades con nuevos valores para la clasificación de campo / expresión.

Se pueden hacer más ajustes a la clasificación predeterminada:

- You can Add new categories, Remove selected categories, *Delete All* of them or *Delete Unused* categories.
- Se puede deshabilitar una clase desmarcando la casilla a la izquierda del nombre de la clase; las entidades correspondientes serán ocultadas en el mapa.
- Arrastre y pegue las filas para reordenar las clases
- Para cambiar el símbolo, el valor o la leyenda de una clase, haga doble click en el elemento.

Click-derecho sobre el elemento seleccionado(s) muestra un menú contextual a:

- *Copy Symbol* y *Paste Symbol*, una forma conveniente de aplicar la representación del artículo a otros
- *Change Color…* de el símbolo(s) seleccionado
- *Change Opacity…* de el símbolo(s) seleccionado
- *Change Output Unit…* de el símbolo(s) seleccionado
- *Change Width…* de el símbolo(s) seleccionado
- *Change Size…* de el símbolo(s) seleccionado
- *Change Angle…* de el símbolo(s) seleccionado
- *Merge Categories*: Agrupa múltiples categorías seleccionadas en una sola. Esto permite un estilo más simple de capas con una gran cantidad de categorías, donde es posible agrupar numerosas categorías distintas en un conjunto de categorías más pequeño y manejable que se aplica a múltiples valores.

> **Truco**
>
> Dado que el símbolo que se mantiene para las categorías fusionadas es una de las categorías seleccionadas en la parte superior de la lista, es posible que desee mover la categoría cuyo símbolo desea reutilizar a la parte superior antes de fusionar.

- *Unmerge Categories* que fue previamente mezclado

Las clases creadas también aparecen en una jerarquía de árbol en el panel *Capas*. Haga doble clic en una entrada de la leyenda del mapa para editar el símbolo asignado. Haga clic con el botón derecho y obtendrá algunas *más opciones*.

El menú *Advanced* da acceso a opciones para acelerar la clasificación o afinar la representación de símbolos:

- *Match to saved symbols*: Usando la *symbols library*, asigna a cada categoría un símbolo cuyo nombre representa el valor de clasificación de la categoría
- *Match to symbols from file…*: Proporciona un archivo con símbolos, asigna a cada categoría un símbolo cuyo nombre representa el valor de clasificación de la categoría.
- *Symbol levels…* para definir el orden de representación de símbolos.

Representador Graduado

El representador graduado  *Graduated* se usa para representar todas las entidades de una capa, utilizando un símbolo definido por el usuario cuyo color o tamaño refleja la asignación del atributo de una entidad seleccionada a una clase.

Al igual que el Renderizador categorizado, el Renderizador graduado le permite definir la rotación y la escala de tamaño de las columnas especificadas.

Además, de forma análoga al Renderizador categorizado, le permite seleccionar:

- El *Valor* de la clasificación: puede ser un campo existente o una *expresión* puede escribir en la casilla o construir utilizando el botón asociado  . El uso de expresiones para la graduación evita la necesidad de crear un campo sólo para fines de simbología (por ejemplo, si sus criterios de clasificación se derivan de uno o más atributos).
- El símbolo (usando el cuadro de diálogo Selector de símbolo)
- El formato de leyenda y la precisión
- El método a usar para cambiar el símbolo: color o tamaño
- Los colores (usando la lista de Rampas de color) si se selecciona el método de color
- El tamaño (usando el dominio de tamaño y su unidad)

Luego puede usar la pestaña Histograma que muestra un histograma interactivo de los valores del campo o expresión asignados. Los saltos de clase se pueden mover o agregar usando el widget de histograma.

> **Nota**
>
> Puede usar el panel Resumen estadístico para obtener más información sobre su capa vectorial. Ver *Panel de resumen estadístico*.

De vuelta a la pestaña Clases, puede especificar el número de clases y también el modo para clasificar las características dentro de las clases (usando la lista Modo). Los modos disponibles son:

- Recuento Equitativo (Cuantil): cada clase tendrá el mismo número de elementos (la idea de un diagrama de caja).
- Intervalo Equitativo: cada clase tendrá el mismo tamaño (por ejemplo, con los valores de 1 a 16 y cuatro clases, cada clase tendrá un tamaño de cuatro).

- Intervalo fijo: cada clase tendrá un intervalo fijo de valores (por ejemplo, con los valores de 1 a 16 y un tamaño de intervalo de 4, las clases serán 1-4, 5-8, 9-12 y 13-16).
- Escala logarítmica: adecuada para datos con una amplia gama de valores. Clases estrechas para valores bajos y clases amplias para valores grandes (por ejemplo, para números decimales con rango [0..100] y dos clases, la primera clase será de 0 a 10 y la segunda clase de 10 a 100).
- Rupturas naturales(Jenks): la variación dentro de cada clase se minimiza mientras que la variación entre clases se maximiza.
- Rupturas Agraciadas: calcula una secuencia de aproximadamente n+1 valores agradables igualmente espaciados que cubren el rango de los valores en x. Los valores se eligen para que sean 1, 2 o 5 veces una potencia de 10. (basado en pretty del entorno estadístico R https://www.rdocumentation.org/packages/base/topics/pretty).
- Desviación estándar: las clases se crean según la desviación estándar de los valores.

El cuadro de lista en el centro de la pestaña *Simbology* enumera las clases junto con sus rangos, etiquetas y símbolos que se mostrarán.

Haga clic en el botón **Classify** para crear clases usando el modo elegido. Cada clase se puede desactivar desmarcando la casilla de verificación a la izquierda del nombre de la clase.

Para cambiar el símbolo, el valor y/o la etiqueta de la clase, simplemente haga doble clic en el elemento que desea cambiar.

Click-derecho sobre el elemento seleccionado(s) muestra un menú contextual a:

- *Copy Symbol* y *Paste Symbol*, una forma conveniente de aplicar la representación del artículo a otros
- *Change Color…* de el símbolo(s) seleccionado
- *Change Opacity…* de el símbolo(s) seleccionado
- *Change Output Unit…* de el símbolo(s) seleccionado
- *Change Width…* de el símbolo(s) seleccionado
- *Change Size…* de el símbolo(s) seleccionado
- *Change Angle…* de el símbolo(s) seleccionado

El ejemplo en Figura 12.5 muestra el cuadro de diálogo de renderizado graduado para la capa major_rivers del conjunto de datos de muestra de QGIS.

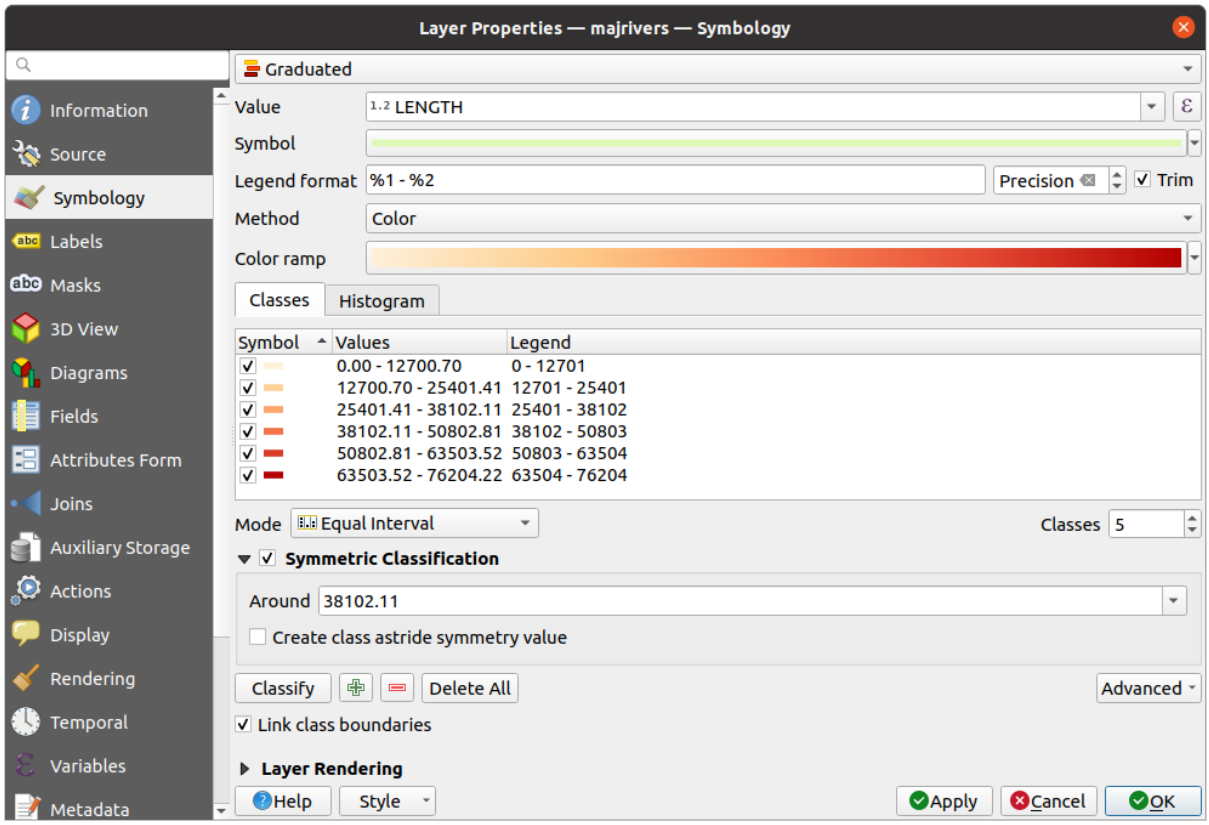


Figura 12.5: Opciones de Simbolización Graduada

Las clases creadas también aparecen en una jerarquía de árbol en el panel *Capas*. Haga doble clic en una entrada de la leyenda del mapa para editar el símbolo asignado. Haga clic con el botón derecho y obtendrá algunas *más opciones*.

Símbolo proporcional y Análisis Multivariante

El símbolo proporcional y el análisis multivariado no son tipos de representación disponibles en la lista desplegable de representación de simbología. Sin embargo con el *data-defined override* opciones aplicadas sobre cualquiera de las opciones de representación anteriores, QGIS le permite mostrar sus datos de punto y línea con dicha representación.

Crear símbolo proporcional

Para aplicar un renderizado proporcional:

1. Primero aplique a la capa el *single symbol renderer*.
2. Luego establezca el el símbolo a aplicar a las entidades.
3. Seleccione el elemento en el nivel superior del árbol de símbolos y use el Data-defined override *button* próximo a la opción *Size* (para capa de puntos) o *Width* (para capa de lineas).
4. Seleccione un campo o ingrese una expresión, y para cada característica, QGIS aplicará el valor de salida a la propiedad y redimensionará proporcionalmente el símbolo en el lienzo del mapa.

   Si fuera necesario, use la opción *Size assistant…* de | dataDefine | menú para aplicar alguna transformación (exponencial, flannery …) al cambio de tamaño del símbolo(ver *Usar la interfaz de asistente definida por datos* fpara mas detalles).

Puede elegir mostrar los símbolos proporcionales en el *Layers panel* y el *print layout legend item*: desplegue el *Advanced* lista desplegable en la parte inferior del cuadro de diálogo principal de la pestaña *Symbology* y seleccione **Data-defined size legend…** para configurar los elementos de la leyenda (ver *Leyenda de tamaño definido por datos* para detalles).

Crear análisis multivariable

Un análisis multivariado le ayuda a evaluar la relación entre dos o más variables, por ejemplo, una puede representarse con una rampa de color y la otra con un tamaño.

El modo mas simple para crear análisis multivariable en QGIS es:

1. Primero aplique una representación categorizada o graduada en una capa, utilizando el mismo tipo de símbolo para todas las clases.
2. Luego, aplique una simbología proporcional en las clases:
    1. Click en el botón *Change* sobre el cuadro de la clasificación: accederá al diálogo *EL selector de símbolo*.
    2. Cambie la escala del tamaño o el ancho de la capa de símbolo utilizando el widget *data defined override* como se muestra encima.

Al igual que el símbolo proporcional, la simbología escalada se puede agregar al árbol de capas, encima de los símbolos de clases graduadas o categorizadas utilizando la función *data defined size legend*. Y ambas representaciones también están disponibles en el elemento de leyenda de diseño de impresión.

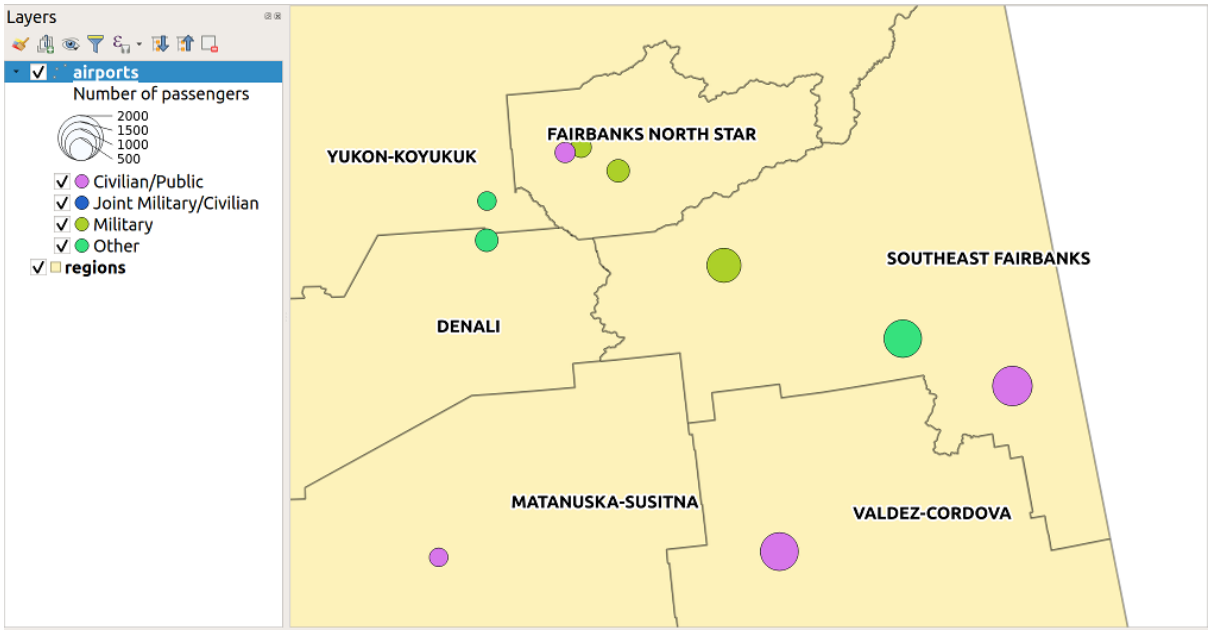


Figura 12.6: Ejemplo multivariable con leyenda de tamaño escalado

Renderizado basado en reglas

Las reglas son QGIS *expressions* utilizadas para discriminar características según sus atributos o propiedades para aplicarles configuraciones de representación específicas. Las reglas se pueden anidar y las entidades pertenecen a una clase si pertenecen a todos los niveles superiores de anidamiento.

El *Rule-based* está diseñado para renderizar todas las características de una capa, utilizando símbolos cuyo aspecto refleja la asignación de una característica seleccionada a una clase de grano fino.

Para crear una regla:

1. Active una fila existente haciendo doble clic en ella (de manera predeterminada, QGIS agrega un símbolo sin una regla cuando el modo de representación está habilitado) o haga clic en Edit rule or Add rule.
2. En el diálogo *Edit Rule* que se abre, puede definir una etiqueta para ayudarlo a identificar cada regla. Esta es la etiqueta que se mostrará en *Layers Panel* y también en la leyenda del compositor de impresión.

3. Ingrese manualmente una expresión en el cuadro de texto junto a la opción  *Filter* o presione la | expresión | botón al lado para abrir el diálogo del generador de cadenas de expresión.
4. Utilice las funciones proporcionadas y los atributos de capa para construir una *expression* para filtrar las entidades que le gustaría recuperar. Presione el botón *Test* para verificar el resultado de la consulta.
5. Puede introducir una etiqueta mas larga para completar la descripción de la regla.
6. Puede usar la opción  :guilabel:`Scale Range`para establecer escalas a las que se debe aplicar la regla.
7. Finalmente, configure el *symbol to use* para estas entidades.
8. Y presione *OK*.

Se agrega una nueva fila que resume la regla al cuadro de diálogo Propiedades de capa. Puede crear tantas reglas como sea necesario siguiendo los pasos anteriores o copiar y pegar una regla existente. Arrastre y suelte las reglas una encima de la otra para anidarlas y refinar las características de la regla superior en las subclases.

El renderizador basado en reglas se puede combinar con renderizadores categorizados o graduados. Seleccionando una regla, puedes organizar sus características en subclases usando el menú desplegable *Refine selected rules*. Las clases refinadas aparecen como subelementos de la regla, en una jerarquía de árbol y, al igual que su padre, puede establecer la simbología y la regla de cada clase. El refinamiento de reglas automatizado se puede basar en:

- **scales**: dada una lista de escalas, esta opción crea un conjunto de clases a las que se aplican los diferentes rangos de escala definidos por el usuario. Cada nueva clase basada en escalas puede tener su propia simbología y expresión de definición. Esto puede, p. ser una forma conveniente de mostrar las mismas características con varios símbolos en diferentes escalas, o mostrar solo un conjunto de características según la escala (por ejemplo, aeropuertos locales a gran escala frente a aeropuertos internacionales a pequeña escala).
- **categories**: aplica un *categorized renderer* a las características que caen en la regla seleccionada.
- o **ranges**: aplica un *graduated renderer* a las características que caen en la regla seleccionada.

Las clases refinadas aparecen como subelementos de la regla, en una jerarquía de árbol y, como arriba, puede establecer la simbología de cada clase. Los símbolos de las reglas anidadas se apilan uno encima del otro, así que tenga cuidado al elegirlos. También es posible desmarcar  *Symbols* en el cuadro de diálogo *Edit rule* para evitar mostrar un símbolo en particular en la pila.

En el diálogo *Edit rule* , puedes evitar escribir todas las reglas y hacer uso de la opción  *Else* para capturar todas las características que no coinciden con ninguna de las otras reglas, en el mismo nivel. Esto también se puede lograr escribiendo `` Else`` en la columna * Rule * del diálogo *Layer Properties* ► *Symbology* ► *Rule-based*.

Click-derecho sobre el elemento seleccionado(s) muestra un menú contextual a:

- *Copy* y *Paste*, Una forma conveniente de crear nuevos artículos basados en los artículos existentes
- *Copy Symbol* y *Paste Symbol*, una forma conveniente de aplicar la representación del artículo a otros
- *Change Color…* de el símbolo(s) seleccionado
- *Change Opacity…* de el símbolo(s) seleccionado
- *Change Output Unit…* de el símbolo(s) seleccionado
- *Change Width…* de el símbolo(s) seleccionado
- *Change Size…* de el símbolo(s) seleccionado
- *Change Angle…* de el símbolo(s) seleccionado
- *Refine Current Rule*: abre un submenú que permite refinar la regla actual con **scales**, **categories** or **Ranges**. Igual que seleccionar el *corresponding menu* en la parte inferior del cuadro de diálogo.

Al desmarcar una fila en el cuadro de diálogo del renderizador basado en reglas, se ocultan en el lienzo del mapa las características de la regla específica y las anidadas.

Las reglas creadas también aparecen en una jerarquía de árbol en la leyenda del mapa. Haga doble clic en una entrada de la leyenda del mapa para editar el símbolo asignado. Haga clic con el botón derecho y obtendrá algunas *más opciones*.

El ejemplo en Figura 12.7 muestra el cuadro de diálogo de renderizado basado en reglas para la capa de ríos del conjunto de datos de muestra de QGIS.

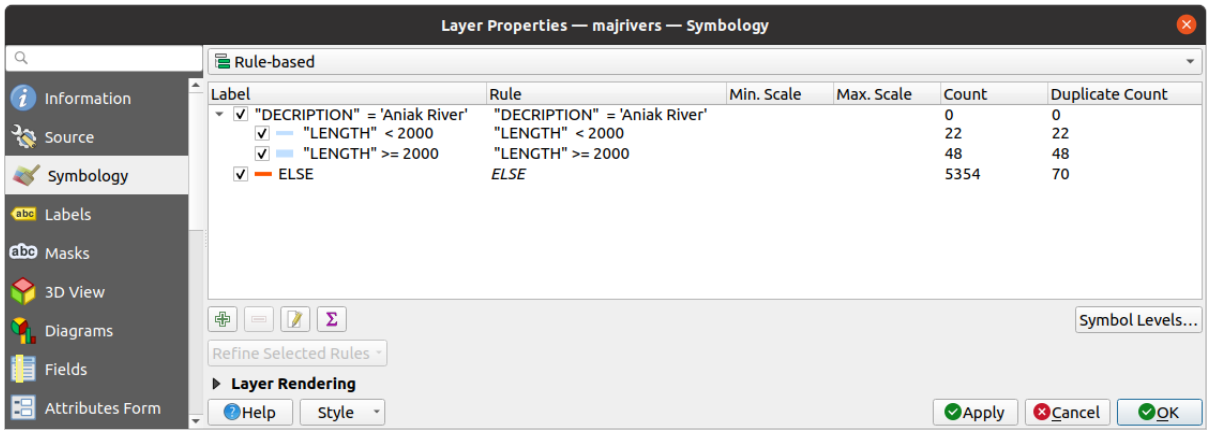


Figura 12.7: Opciones de simbología basada en reglas

Representador con desplazamiento puntual

El renderizador  *DesplazamientoDePuntos* toma los objetos espaciales de puntos que caen en una tolerancia de distancia dada entre sí y coloca sus símbolos alrededor de su baricentro, siguiendo diferentes métodos de colocación. Esta puede ser una forma conveniente de visualizar todas los objetos espaciales de una capa de puntos, incluso si tienen la misma ubicación (por ejemplo, servicios en un edificio).

Para configurar un renderizador de desplazamiento de puntos, tienes que:

1. Establece el *Símbolo del centro*: cómo será el punto virtual en el centro
2. Seleccione el tipo de *Renderizador*: cómo desea clasificar los objetos espaciales en la capa (simples, categorizadas, basadas en reglas…)
3. Pulse el botón *Configuración del renderizador…* para configurar la simbología de las características según el renderizador seleccionado
4. Indica la tolerancia *Distancia* en la que se consideran superpuestas los objetos espaciales cercanos y luego desplazadas sobre el mismo punto virtual. Admite unidades de símbolo comunes.
5. Configure los métodos *Colocación*:
   - **Ring**: coloca todas las entidades en un círculo cuyo radio depende de la cantidad de entidades que se mostrarán.
   - **Concentric rings**: usa un conjunto de círculos concéntricos para mostrar las entidades.
   - **Grid**: genera una malla regular con un símbolo puntual en cada intersección.
6. Los símbolos desplazados se colocan en las *Líneas de desplazamiento*. Aunque el espaciado mínimo de las líneas de desplazamiento depende del renderizador de símbolos de puntos, puede personalizar algunos de sus ajustes, como la *Anchura de trazo*, *Color de trazo* y *Ajuste de tamaño* (por ejemplo, para añadir más espaciado entre los puntos renderizados).
7. Utilice las opciones del grupo *Etiquetas* para realizar el etiquetado de puntos: las etiquetas se colocan cerca del símbolo desplazado, y no en la posición real del objeto espacial.
   1. Seleccione el atributo *Etiqueta*: un campo de la capa a utilizar para el etiquetado
   2. Indicar las propiedades y el tamaño de :guilabel:`Fuente de la etiqueta
   3. Elija un :guilabel:`Color de la etiqueta

4. Establezca un *Factor de distancia de la etiqueta*: para cada objeto espacial punto, desplaza la etiqueta desde el centro del símbolo proporcionalmente al tamaño de la diagonal del símbolo.

5. Active *Utilizar etiquetado dependiente de la escala* si desea mostrar etiquetas sólo en escalas mayores que una determinada *Escala mínima del mapa*.

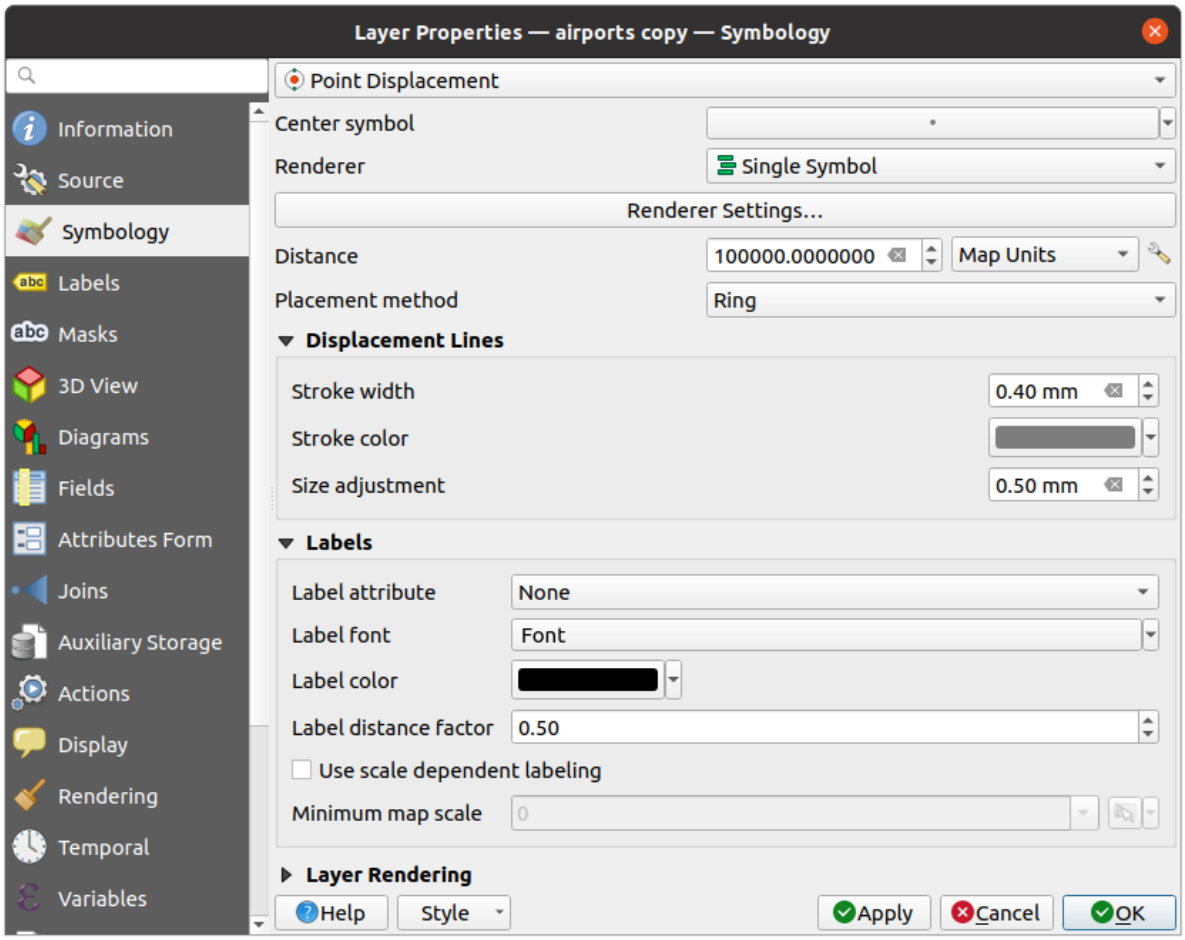


Figura 12.8: Cuadro de diálogo de desplazamiento de punto

Nota

El renderizador de desplazamiento de puntos no altera la geometría de la entidad, lo que significa que los puntos no se mueven de su posición. Todavía se encuentran en su lugar inicial. Los cambios son solo visuales, para fines de representación. Utilice en su lugar el algoritmo de procesamiento *Desplazamiento de puntos* si quiere crear geometrías desplazadas.

Representador de Clúster de punto

A diferencia del renderizador *Desplazamiento de puntos*, que amplía la colocación de los objetos espaciales puntos más cercanos o superpuestos, el renderizador *Cúmulo de puntos* agrupa los puntos cercanos en un único símbolo de marcador renderizado. Los puntos que se encuentran dentro de una distancia especificada entre sí se fusionan en un único símbolo. La agregación de puntos se realiza en función del grupo más cercano que se forme, en lugar de asignarles simplemente el primer grupo dentro de la distancia de búsqueda.

Desde el diálogo principal, puede:

1. Establece el símbolo para representar el cluster de puntos en la *Símbolo de cluster*; la representación por defecto muestra el número de objetos espaciales agregados gracias a la variable ``@cluster_size`. <general_tools_variables>` en la capa Símbolo de marcador de fuente.
2. Seleccione el tipo de *Renderizador*, es decir, cómo desea clasificar los objetos espaciales en la capa (simples, categorizados, basados en reglas…)
3. Pulse el botón *Configuración del renderizador…* para configurar la simbología de los elementos de la forma habitual. Tenga en cuenta que esta simbología sólo es visible en los elementos que no están agrupados, ya que en caso contrario se aplica el símbolo *Agrupamiento*. Además, cuando todas los objetos espaciales punto de un grupo pertenecen a la misma clase de renderizado, y por tanto se les aplicaría el mismo color, ese color representa la variable `@cluster_color` del grupo.
4. Indique la máxima *Distancia* a considerar para agrupar objetos espaciales. Admite unidades de símbolo comunes.

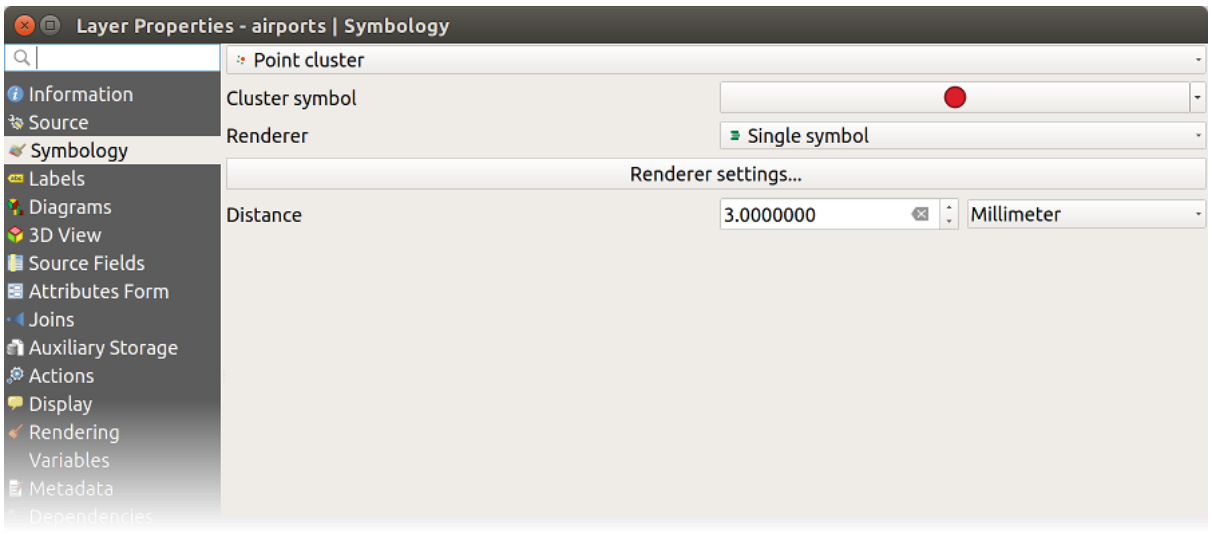


Figura 12.9: Dialogo de Clúster de puntos

Nota

El renderizador de agrupación de puntos no altera la geometría de la entidad, lo que significa que los puntos no se mueven de su posición. Todavía se encuentran en su lugar inicial. Los cambios son solo visuales, para fines de representación. Utilice en su lugar el procesamiento *Agrupación de K-medias* o el algoritmo *Clustering o agrupamiento DBSCAN* si quiere crear entidades basadas en clúster.

Representador de entidades fusionadas

El renderizador *Merged Features* permite que las características de área y línea se «disuelvan» en un solo objeto antes de la representación para garantizar que los símbolos complejos o las características superpuestas se representen mediante un símbolo cartográfico uniforme y contiguo.

Representador de Polígono Invertido

El representador *Inverted Polygon* permite al usuario definir un símbolo para completar fuera de los polígonos de la capa. Como se indicó anteriormente, puede seleccionar subrederes, a saber, Símbolo único, Graduado, Categorizado, Basado en reglas o renderizador 2.5D.

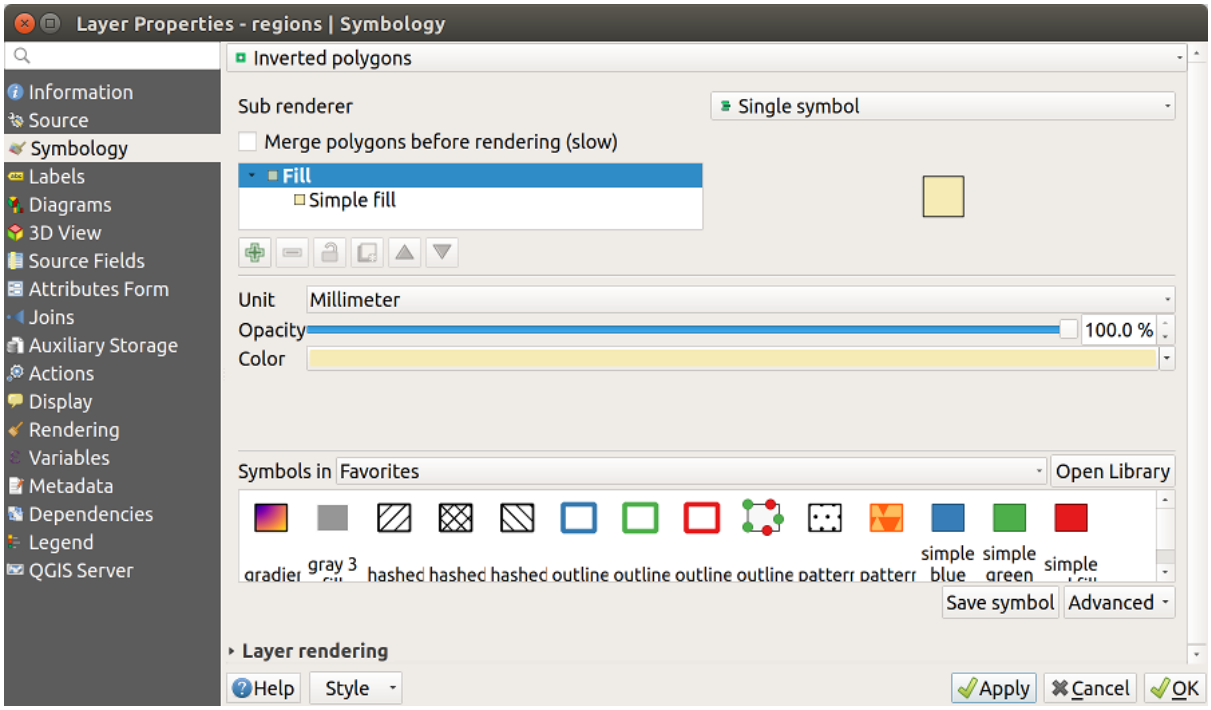


Figura 12.10: Diálogo de polígono invertido

Renderizado de mapa de calor

Con el renderizador *Mapa de calor* puede crear mapas de calor dinámicos para capas (multi)punto. Puede especificar el *Radio* del mapa de calor en milímetros, puntos, píxeles, unidades de mapa o pulgadas, elegir y editar una *Rampa de color* para el estilo del mapa de calor y utilizar un deslizador para seleccionar un compromiso entre velocidad de renderizado y calidad. También puede definir un límite *Maximum value* y *Pesar puntos por* utilizando un campo o una expresión.

Utilice Sustitución definida por datos para controlar dinámicamente *Radio* y *Valor máximo* basándose en los atributos de sus datos. Por ejemplo, el radio de un punto del mapa de calor podría estar determinado por su atributo de población, o el valor máximo podría basarse en un rango temporal.

Al añadir o eliminar una característica, el renderizador del mapa térmico actualiza el estilo del mapa térmico automáticamente. La *Rampa de colores* se mostrará como una barra de leyenda y en la *Configuración de la leyenda* puede establecer las *Etiquetas* para los valores *Máximo* y *Mínimo*. También puede cambiar la orientación y dirección de la leyenda en *Composición*.

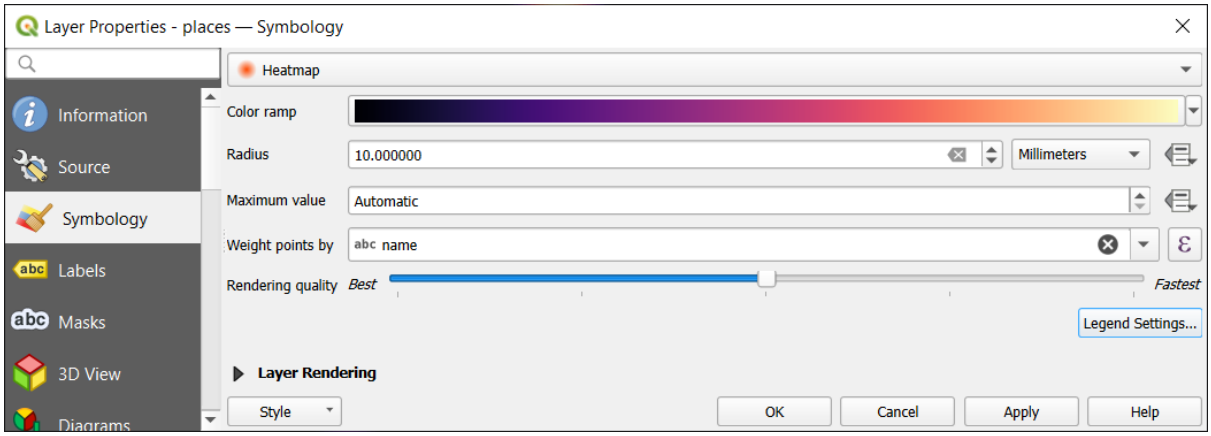


Figura 12.11: Diálogo Mapa de calor

Renderizador 2.5D

Usando el representador *2.5D* Es posible crear un efecto 2.5D en las características de su capa. Empiezas eligiendo un valor *Height* (en unidades de mapa). Para eso, puede usar un valor fijo, uno de los campos de su capa o una expresión. También debe elegir un: guilabel: *Angle* (en grados) para recrear la posición del espectador (0° significa oeste, creciendo en sentido contrario a las agujas del reloj). Utilice las opciones de configuración avanzada para configurar el *Roof Color* y *Wall Color*. Si desea simular la radiación solar en las paredes de las entidades, asegúrese de marcar la opción *Shade walls based on aspect*. También puede simular una sombra estableciendo un *Color* y:guilabel:*Size* (en unidades de mapa).

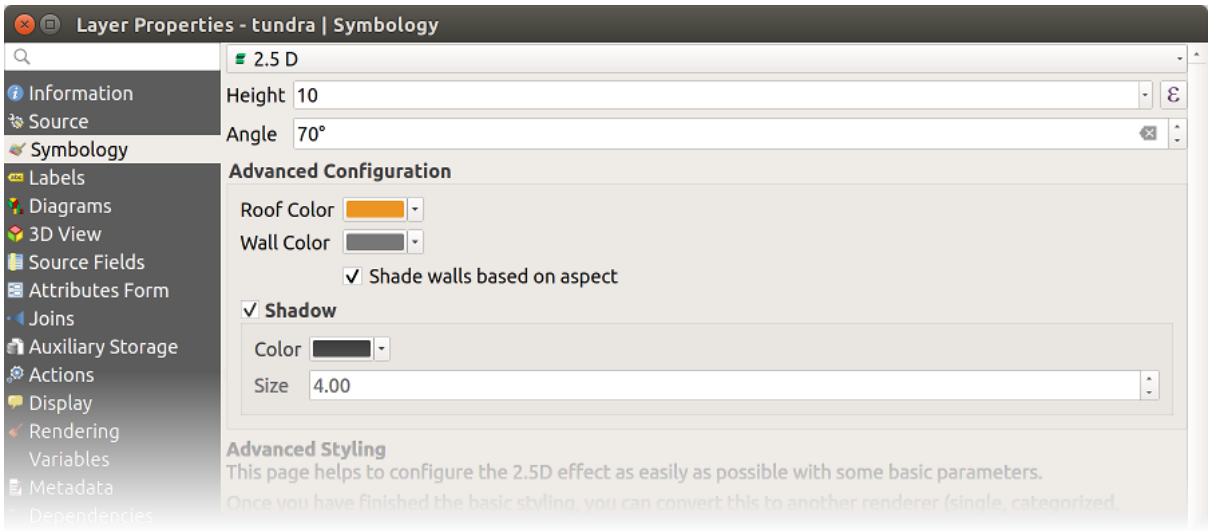


Figura 12.12: Diálogo 2.5D

Truco

Uso del efecto 2.5D con otros renderizadores

Una vez que haya terminado de configurar el estilo básico en el renderizador 2.5D, puede convertirlo a otro renderizador (único, categorizado, graduado). Los efectos 2.5D se mantendrán y todas las demás opciones específicas del renderizador estarán disponibles para que pueda ajustarlos (de esta manera puede tener, por

ejemplo, símbolos categorizados con una buena representación 2.5D o agregar un estilo adicional a sus símbolos 2.5D). Para asegurarse de que la sombra y el «edificio» en sí no interfieran con otras características cercanas, es posible que necesite habilitar los niveles de símbolos (*Avanzado -> Niveles de símbolos …*). Los valores de altura y ángulo 2.5D se guardan en las variables de la capa, por lo que puede editarlo luego en la pestaña de variables del cuadro de diálogo de propiedades de la capa.

Representador integrado

El renderizador *Embedded Symbols* permite mostrar la simbología "native" de una fuente de datos proporcionada. Este es principalmente el caso con los conjuntos de datos `KML` y `TAB` que tienen una simbología predefinida.

Reproducción de capas

Desde la pestaña Simbología, también puede establecer algunas opciones que invariablemente actúan en todas las características de la capa:

- *Opacity*  : Puede hacer que la capa subyacente en el lienzo del mapa sea visible con esta herramienta. Use el control deslizante para adaptar la visibilidad de su capa vectorial a sus necesidades. También puede hacer una definición precisa del porcentaje de visibilidad en el menú junto al control deslizante.
- *Blending mode* en la *Layer* y niveles *Feature*: Puede lograr efectos de renderizado especiales con estas herramientas que anteriormente solo conocía de los programas gráficos. Los píxeles de sus capas superpuestas y subyacentes se mezclan a través de la configuración descrita en *Modos de Mezcla*.
- Aplicar *paint effects* en todas las entidades de la capa con el botón *Draw Effects*.
- *Control feature rendering order* le permite, usando atributos de entidad, para definir el orden z en el que se representarán. Active la casilla de verificación y haga clic en el botón adyacente  . Entonces accederá al diálogo *Define Order* en el cual puede:

  1. Elija un campo o cree una expresión para aplicar a las entidades de capa.
  2. Establezca en qué orden se deben ordenar las entidades recuperadas, es decir, si elige el orden **Ascending**, las entidades con un valor más bajo se representan bajo aquellas con un valor más alto.
  3. Defina cuándo se deben representar las entidades que devuelven el valor NULL: **first** (abajo) o **last** (arriba).
  4. Repita los pasos anteriores tantas veces como las reglas que desee utilizar.

  La primera regla se aplica a todas las entidades en la capa, ordenándolas en z según su valor devuelto. Luego, dentro de cada grupo de entidades con el mismo valor (incluidas aquellas con valor NULO) y, por lo tanto, el mismo nivel z, se aplica la siguiente regla para ordenarlas. Y así…

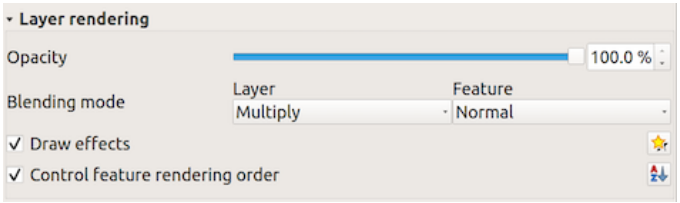


Figura 12.13: Opciones de renderizado de capa

Otras Configuraciones

Niveles de símbolo

Para los renderizadores que permiten capas de símbolo apiladas (solo el mapa de calor no lo permite) hay una opción para controlar el orden de representación de los niveles de cada símbolo.

Para la mayoría de los renderizadores, puede acceder a la opción de niveles de Símbolos haciendo clic en el botón *Advanced* debajo de la lista de símbolos guardados y elegidos *Symbol levels*. Para la *Renderizado basado en reglas* la opción está disponible directamente mediante el botón *Symbols Levels...* , mientras para *Representador con desplazamiento puntual* renderizador el mismo botón está dentro del diálogo *Rendering settings*.

Para activar los niveles de símbolos, seleccione *Enable symbol levels*. Cada fila mostrará una pequeña muestra del símbolo combinado, su etiqueta y la capa de símbolos individuales dividida en columnas con un número al lado. Los números representan el nivel de orden de representación en el que se dibujará la capa de símbolo. Los niveles de valores más bajos se dibujan primero, permanecen en la parte inferior, mientras que los valores más altos se dibujan en último lugar, por encima de los demás.

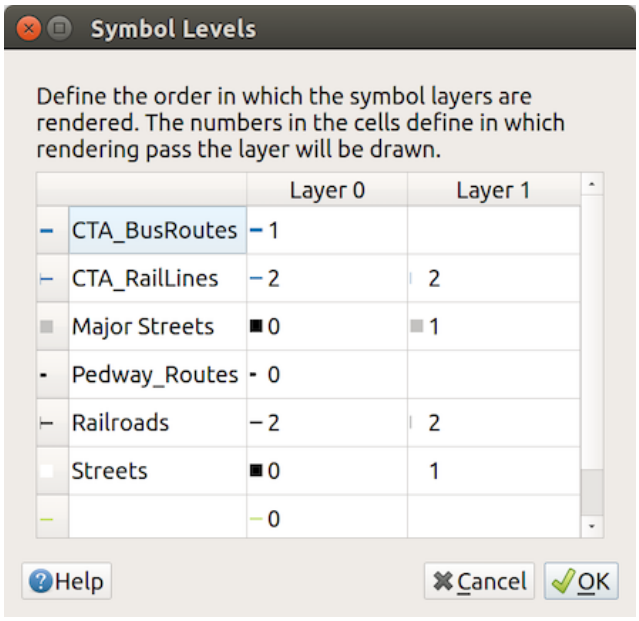


Figura 12.14: Diálogo de niveles de símbolo

Nota

Si los niveles de símbolos están desactivados, los símbolos completos se dibujarán de acuerdo con su respectivo orden de características. Los símbolos superpuestos simplemente se ofuscarán a otros a continuación. Además, los símbolos similares no se «fusionarán» entre sí.

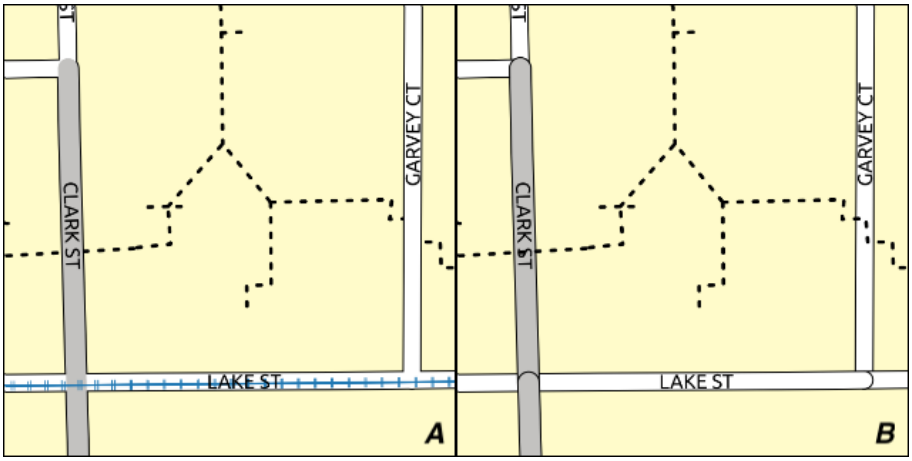


Figura 12.15: Diferencia de niveles de símbolos activados (A) y desactivados (B)

Leyenda de tamaño definido por datos

Cuando una capa es renderizada con la *proportional symbol or the multivariate rendering* o cuando se aplica a la capa un *scaled size diagram* , puede permitir la visualización de los símbolos escalados tanto en el *Layers panel* y en *print layout legend*.

Para activar el diálogo *Data-defined Size Legend* para representar la simbología, seleccione la opción del epónimo en el botón inferior *Advanced* La lista de símbolos guardados. Para los diagramas, la opción está disponible en la pestaña *Legend*. El cuadro de diálogo proporciona las siguientes opciones para:

- seleccione el tipo de leyenda: *Legend not enabled*, *Separated legend items* y *Collapsed legend*. Para la última opción, puede seleccionar si los elementos de la leyenda están alineados en **Bottom** o en el **Center**;
- obtener una vista previa del símbolo *symbol to use* para la representación de la leyenda;
- Inserte el título en la leyenda;
- cambiar el tamaño de las clases para usar: de forma predeterminada, QGIS le proporciona una leyenda de cinco clases (basadas en la opción *Manual size classes*. Use los botones y para configurar sus valores y etiquetas de clases personalizadas.
- Para la leyenda colapsada, es posible:
  - *Align symbols* en el centro o abajo
  - configurar la directriz horizontal *Line symbol* desde el símbolo hasta el texto de la leyenda correspondiente.

Se muestra una vista previa de la leyenda en el panel derecho del cuadro de diálogo y se actualiza a medida que configura los parámetros.

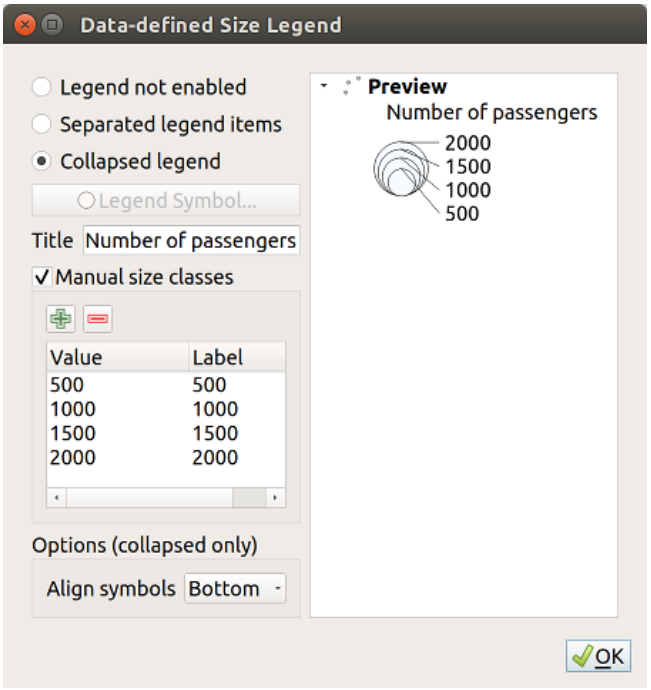


Figura 12.16: Estableciendo la leyenda de tamaño escalado

Nota

Actualmente, la leyenda de tamaño definida por datos para la simbología de capa solo se puede aplicar a la capa de punto usando simbología individual, categorizada o graduada.

Ajustes de animación

Para permitir que cualquier símbolo se convierta en un *símbolo animado*, puede utilizar el panel *Configuración de animación*. En este panel, puede activar la animación para el símbolo y establecer una velocidad de fotogramas específica para el redibujado del símbolo.

1. Empiece por ir al nivel superior de símbolos y seleccione el menú *Avanzado* en la parte inferior derecha del cuadro de diálogo
2. Buscar la opción *Configuración de animación*
3. Marque *Es Animado* para activar la animación del símbolo
4. Configure la *Tasa de fotogramas*, es decir, la velocidad a la que se reproducirá la animación.
5. Ahora puede utilizar la variable `@symbol_frame` en cualquier propiedad definida por los datos del subsímbolo para animar dicha propiedad.

Por ejemplo, establecer la rotación del símbolo a la expresión definida por datos `@symbol_frame % 360` hará que el símbolo rote en el tiempo, con la velocidad de rotación dictada por la velocidad de fotogramas del símbolo:

Extent buffer

You may set an extent buffer for a *symbol*. This means that a buffer is applied to the current map extent so that if a feature is outside of the actual map extent but inside the buffered extent it will still be rendered. This is useful for example with symbols which use the geometry generator where you would like to still see the generated geometries even if the actual feature is outside of the map extent.

To edit the extent buffer you can utilize the *Extent buffer* panel.

1. Empiece por ir al nivel superior de símbolos y seleccione el menú *Avanzado* en la parte inferior derecha del cuadro de diálogo
2. Find *Extent buffer* option
3. In the new panel you can set the buffer distance

The buffer distance units can be changed. You can also control the distance value by using the data defined override widget. For example you can change the value based on the current map scale `if(@map_scale > 50000, 5000, 0)`:

Efectos de dibujo

Para mejorar la representación de capas y evitar (o al menos reducir) el recurso a otro software para la representación final de mapas, QGIS proporciona otra funcionalidad poderosa: las opciones *Draw Effects* , que agrega efectos de pintura para personalizar la visualización de capas vectoriales.

La opción está disponible en el diálogo *Layer Properties* ► *Symbology* , bajo el grupo *Layer rendering* (aplicandolo a la capa entera) o en *symbol layer properties* (aplicándolo a las correspondientes entidades). Puedes combinar ambos usos.

Los efectos de pintura se pueden activar marcando la opción *Dibujar efectos* y hacer click en el botón Personalizar efectos. Eso abrirá el Diálogo *Propiedades Efecto* (ver Figura 12.17). Están disponibles los siguientes tipos de efectos, con opciones personalizadas:

- **Source**: Dibuja el estilo original de la entidad de acuerdo con la configuración de las propiedades de la capa. La *Opacity* de su estilo se puede ajustar, así como el *Blend mode* y *Draw mode*. Estas son propiedades comunes para todo tipo de efectos.

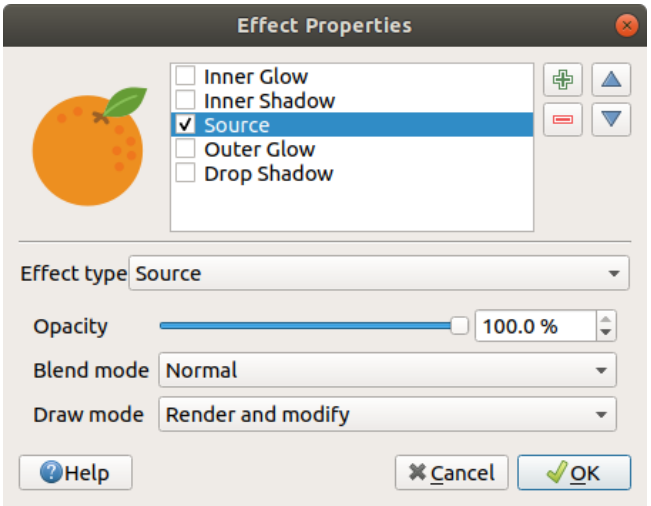


Figura 12.17: Efectos de dibujo: cuadro de diálogo Fuente

- **Blur**: Añade un efecto de difuminado en la capa vectorial. Las opciones personalizadas que puede cambiar son las *Blur type* (*Stack blur (fast)* o *Gaussian blur (quality)*) y la *Blur strength*.

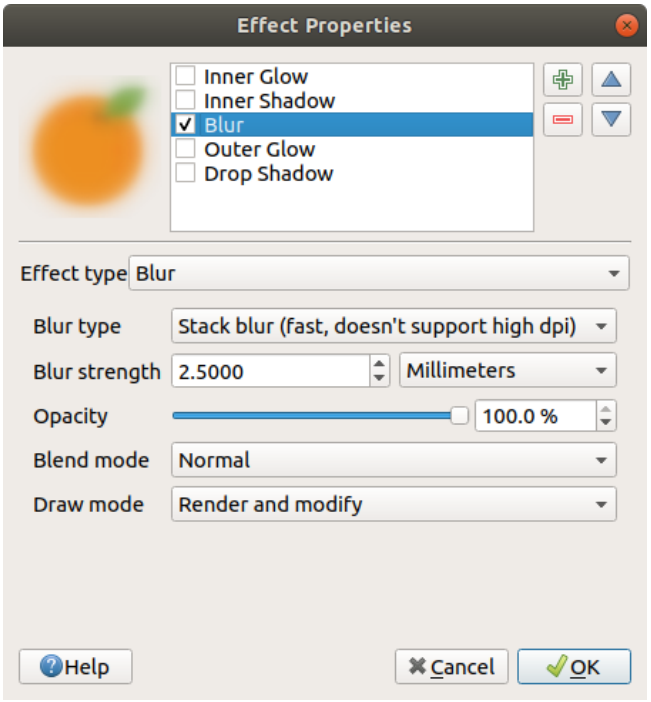


Figura 12.18: Efectos de dibujo: cuadro de diálogo Difuminar

- **Colorise**: Este efecto se puede usar para hacer una versión del estilo usando un solo tono. La base siempre será una versión en escala de grises del símbolo y usted puede:
  - Usar *Grayscale* para seleccionar cómo crearlo: las opciones son “By lightness”, “By luminosity”, “By average” y “Off”.
  - si *Colorise* está seleccionado, será posible mezclar otro color y elegir qué tan fuerte debe ser.
  - controlar *Brightness*, *Contrast* y niveles *Saturation* del símbolo resultante.

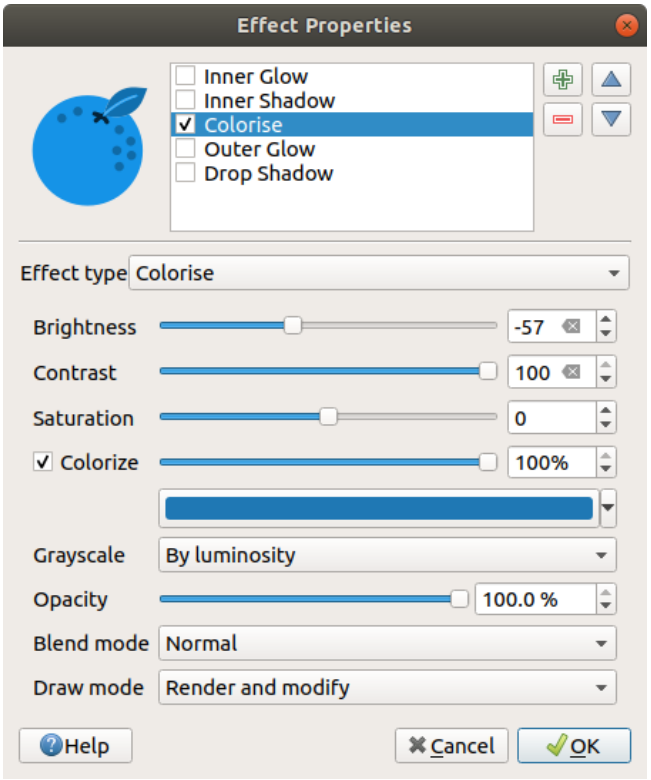


Figura 12.19: Efectos de dibujo: cuadro de diálogo Dar Color

- **Drop Shadow**:El uso de este efecto agrega una sombra en la función, que parece agregar una dimensión adicional. Este efecto se puede personalizar cambiando el ángulo y distancia *Offset* , determinar hacia dónde se desplaza la sombra hacia y la proximidad al objeto de origen. *Drop Shadow* también tiene la opción de cambiar el *Blur radius* y el *Color* del efecto.

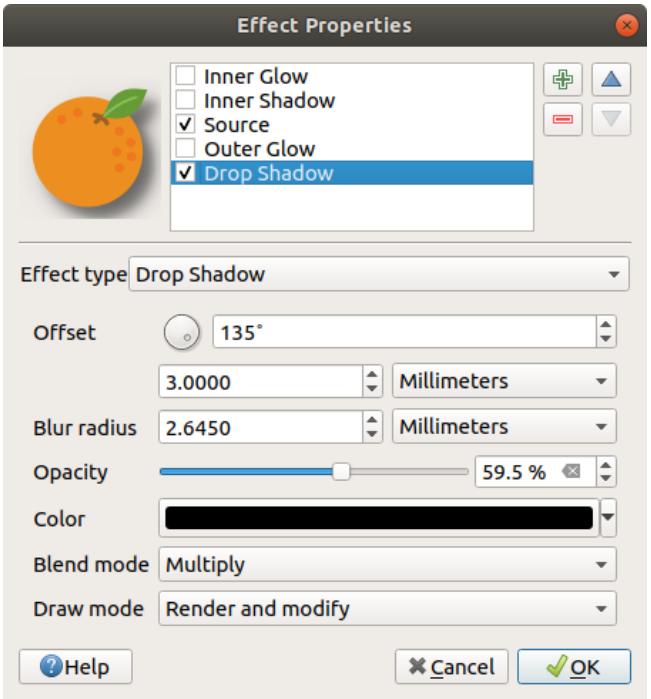


Figura 12.20: Efectos de dibujo: cuadro de diálogo Sombra exterior

- **Inner Shadow**: Este efecto es similar al efecto *Drop Shadow* , pero agrega el efecto de sombra en el interior de los bordes de la entidad. Las opciones disponibles para la personalización son las mismas que las del efecto *Drop Shadow*.

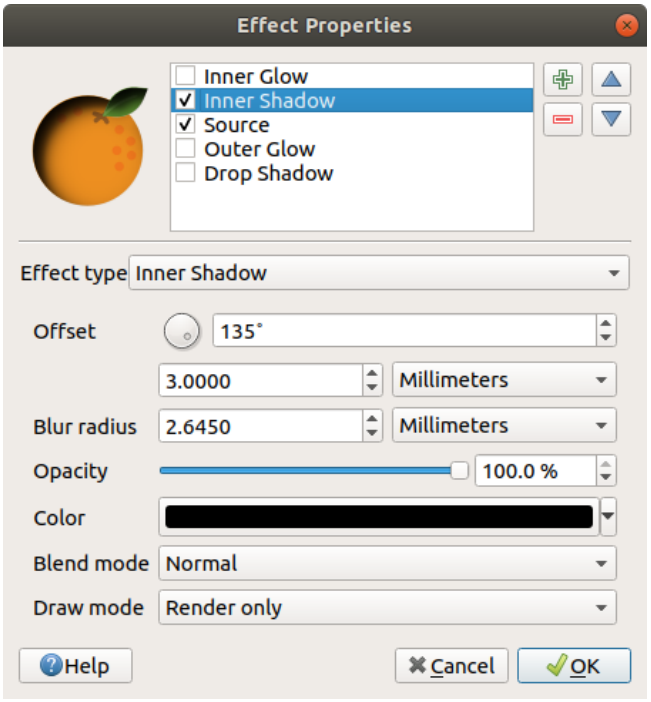


Figura 12.21: Efectos de dibujo: cuadro de diálogo Sombra interior

- **Inner Glow**: Agrega un efecto de brillo dentro de la función. Este efecto se puede personalizar ajustando *Spread* (ancho) del brillo, o *Blur radius*. Este último especifica la proximidad desde el borde de la función donde desea que ocurra cualquier desenfoque. Además, hay opciones para personalizar el color del brillo usando *Single color* o a *Color ramp*.

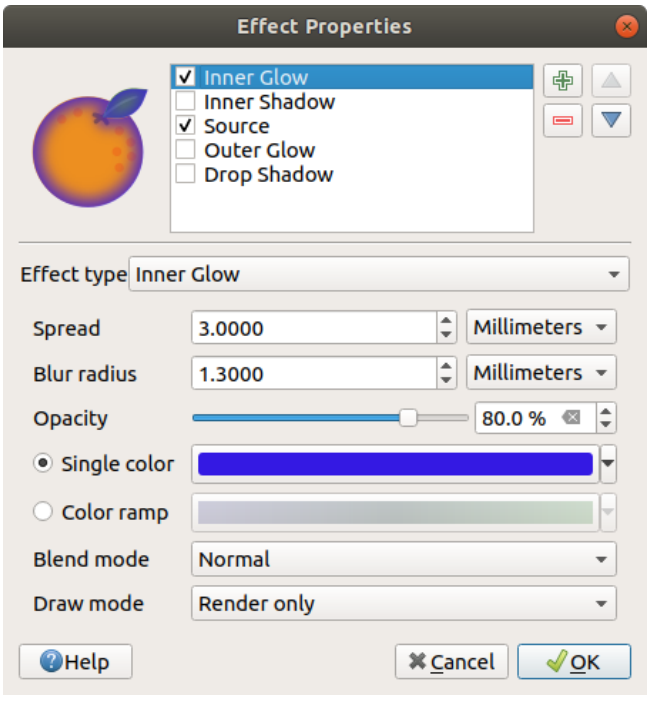


Figura 12.22: Efectos de dibujo: cuadro de diálogo Brillo interior

- **Outer Glow**: Este efecto es similar al efecto *Inner Glow*, pero agrega el efecto de brillo en el exterior de los bordes de la función. Las opciones disponibles para la personalización son las mismas que el efecto *Inner Glow*.

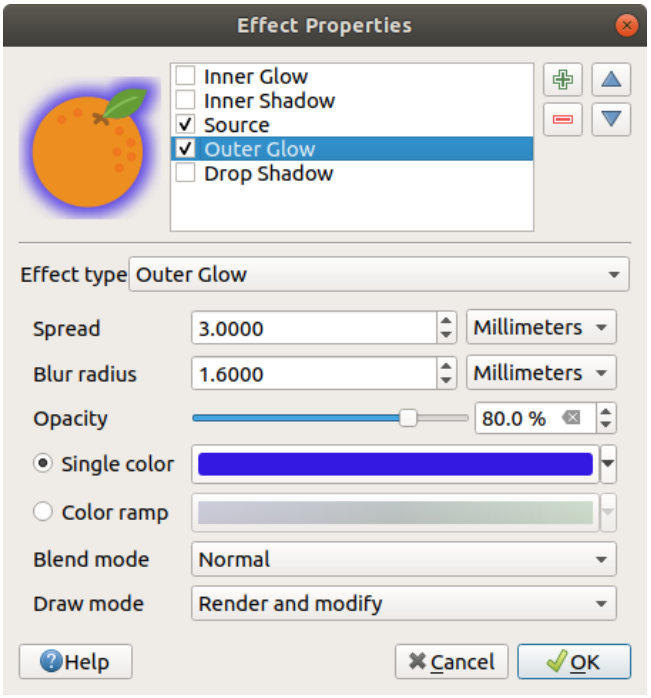


Figura 12.23: Efectos de dibujo: cuadro de diálogo Brillo exterior

- **Transform**: Agrega la posibilidad de transformar la forma del símbolo. Las primeras opciones disponibles para la personalización son *Reflect horizontal* y *Reflect vertical*,que en realidad crean una reflexión sobre los ejes horizontales y / o verticales. Las otras opciones son:
  - *Shear X,Y*: Inclina la entidad a lo largo del eje X y/o Y.
  - *Scale X,Y*: Aumenta o minimiza la función a lo largo del eje X y/o Y en el porcentaje dado.
  - *Rotation*: Gira la entidad alrededor de su punto central.
  - y *Translate X,Y* cambia la posición del elemento en función de una distancia dada en el eje X y/o Y.

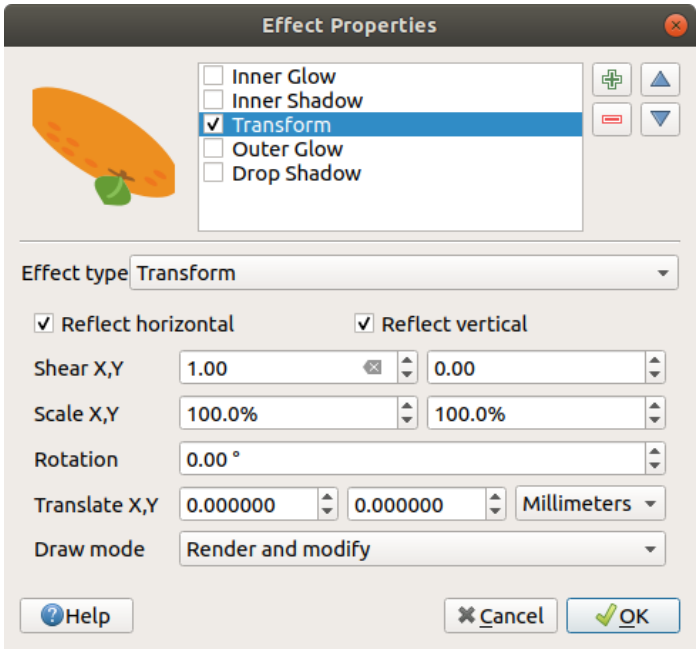


Figura 12.24: Efectos de dibujo: cuadro de diálogo Transformar

Se pueden usar uno o más tipos de efectos al mismo tiempo. Usted (des) activa un efecto usando su casilla de verificación en la lista de efectos. Puede cambiar el tipo de efecto seleccionado utilizando la opción *Effect type*. Puede reordenar los efectos usando los botones Move up y Move down , y también añadir/remover efectos usando los botones Add new effect y Remove effect.

Hay algunas opciones comunes disponibles para todos los tipos de efectos de dibujo. Las opciones *Opacity* y *Blend mode* funcionan de forma similar a las descritas en *Reproducción de capas* y se puede usar en todos los efectos de dibujo, excepto en el de transformación.

También hay una opción disponible *Draw mode* para todos los efectos, y puede elegir entre renderizar y/o modificar el símbolo, siguiendo algunas reglas:

- Los efectos se representan desde arriba a abajo.
- modo *Render only* significa que el efecto será visible.
- modo *Modifier only* significa que el efecto no será visible, pero los cambios que aplique se pasarán al siguiente efecto (el que se encuentra inmediatamente debajo).
- El modo *Render and Modify* hará que el efecto sea visible y pasará cualquier cambio al siguiente efecto. Si el efecto está en la parte superior de la lista de efectos o si el efecto inmediatamente anterior no está en modo de modificación, utilizará el símbolo de fuente original de las propiedades de las capas (similar a la fuente).

12.1.4 Propiedades de etiquetas

Las propiedades *Labels* le proporciona todas las capacidades necesarias y apropiadas para configurar el etiquetado inteligente en capas vectoriales. También se puede acceder a este cuadro de diálogo desde el panel *Layer Styling* , o usando el botón Layer Labeling Options de la **Labels toolbar**.

En la parte superior del diálogo tiene:

- un cuadro combinado para seleccionar el método de etiquetado adecuado para la capa activa
- el botón Configurar reglas de etiquetado del proyecto: le ayuda a controlar las interacciones entre etiquetas y características a través de las capas del proyecto. Más detalles en *Configuración de las reglas de etiquetado de proyectos*.

- el botón Ajustes de colocación automatizada (se aplica a todas las capas): configura propiedades generales sobre la colocación de etiquetas y la resolución de conflictos. Más detalles en *Configurar el motor de colocación automatizado*.

El primer paso es elegir el método de etiquetado de la lista desplegable. Los métodos disponibles son:

- *No labels*: el valor predeterminado, que no muestra etiquetas de la capa
- *Single labels*: Mostrar etiquetas en el mapa usando un solo atributo o una expresión
- *Rule-based labeling*
- y *Blocking*: permite establecer una capa como un obstáculo para las etiquetas de otras capas sin generar ninguna etiqueta propia.

Los siguientes pasos suponen que selecciona la opción *Single labels*, abriendo el siguiente diálogo.

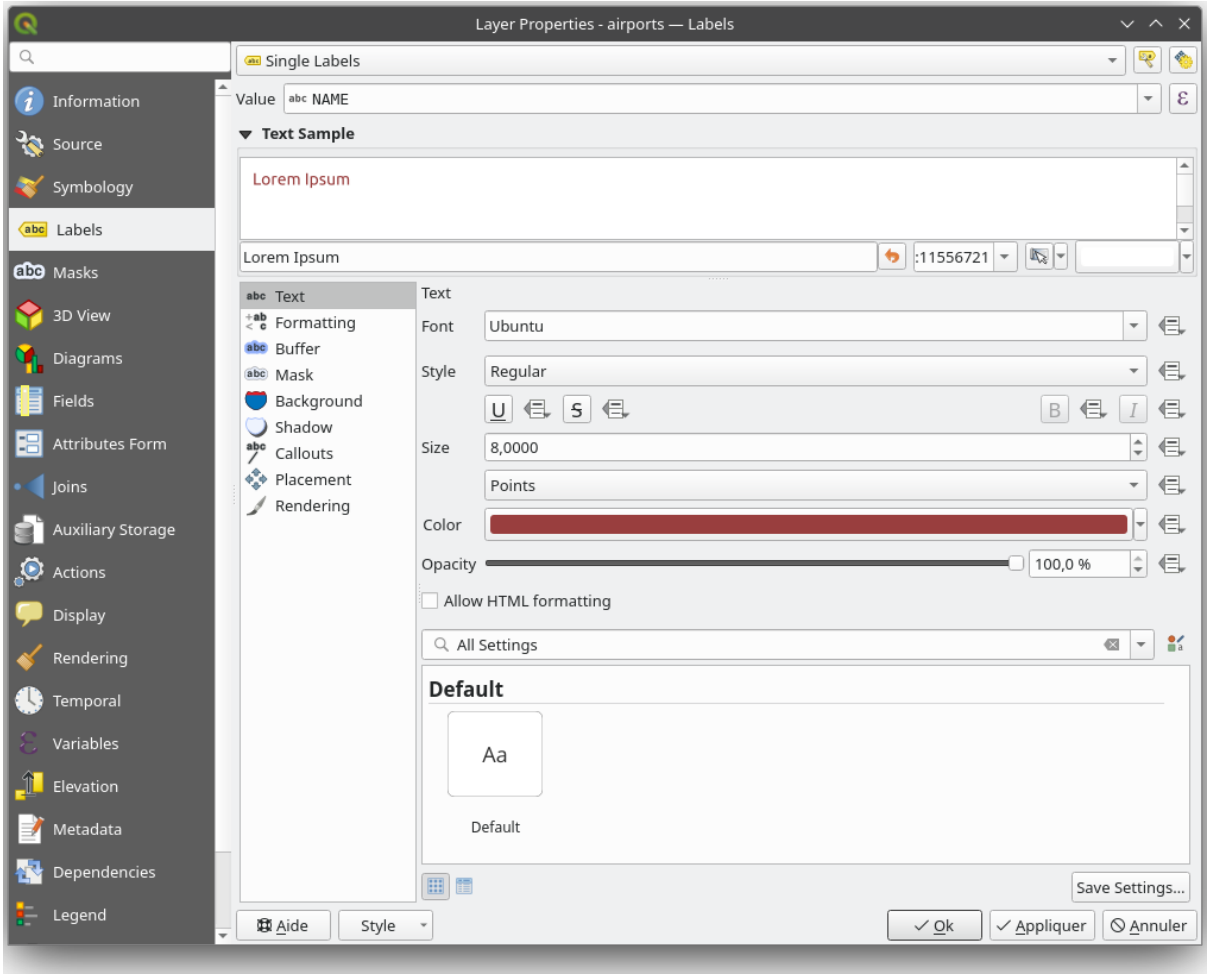


Figura 12.25: Configuración de etiquetado de capa: etiquetas individuales

En la parte superior del diálogo, una lista desplegable de *Value* es activada. Puede seleccionar una columna de atributos para usar en el etiquetado. Por defecto, el *display field* es usado. Click si desea definir etiquetas basadas en expresiones, vea *Definir etiquetas basadas en expresiones*.

Nota

Las etiquetas con su formato se pueden mostrar como entradas en las leyendas, si están habilitadas en la pestaña *Legend*.

A continuación se muestran las opciones para personalizar las etiquetas, en varias pestañas:

- abc *Text*
- *Formatting*
- abc *Buffer*
- abc *Mask*
- *Background*
- *Shadow*
- abc *Callouts*
- *Placement*
- *Rendering*

La descripción de cómo configurar cada propiedad se expone en *Ajustando una etiqueta*.

Configuración de las reglas de etiquetado de proyectos

Pulsando el botón Configurar reglas de etiquetado del proyecto junto al selector desplegable de método de etiquetado, puede crear reglas que controlen cómo las etiquetas de una capa pueden interactuar con etiquetas u objetos espaciales de otra capa.

1. Pulse el botón Añadir regla y en el menú desplegable, seleccione uno de los tipos de regla:
   - *Impedir que las etiquetas se superpongan a los elementos*: impide que las etiquetas se superpongan a elementos de una capa diferente.
   - *Tirar de las etiquetas hacia los objetos espaciales*: evita que las etiquetas se coloquen demasiado lejos de las características de una capa diferente. La distancia máxima puede establecerse en la unidad de su elección.
   - *Alejar etiquetas de elementos*: evita que las etiquetas se coloquen demasiado cerca de elementos de una capa diferente. La distancia mínima puede establecerse en la unidad de su elección, así como la prioridad de la regla . (Es más importante respetar las reglas de mayor prioridad en caso de conflicto de colocación de etiquetas).
   - *Alejar etiquetas de otras etiquetas*: evita que las etiquetas se coloquen demasiado cerca de etiquetas de una capa diferente.

   > **Atención**
   >
   > The last three options are only available on QGIS installed with GEOS >= 3.10 (see *Help* ► *About* menu).

2. Rellene las propiedades a su voluntad; puede dar un nombre más significativo a la regla.
3. Presiona *Aceptar*.
4. Añade tantas reglas como sea necesario.
5. Si es necesario, pulse Editar regla para modificar la regla seleccionada o Eliminar regla para eliminarla del proyecto.

Las reglas establecidas están disponibles desde la pestaña de propiedades de cualquier capa *Etiquetas*, pulsando el botón Configurar reglas de etiquetado del proyecto. Puede activar o desactivar temporalmente cualquiera de ellas, utilizando la casilla situada junto al nombre. Pase el ratón por encima de una regla para previsualizar sus detalles.

Configurar el motor de colocación automatizado

Puede utilizar la configuración de ubicación automatizada para configurar un comportamiento automatizado de las etiquetas a nivel de proyecto. En la esquina superior derecha de la pestaña *Etiquetas*, haga clic en el botón Configuración de ubicación automatizada (se aplica a todas las capas), abriendo un cuadro de diálogo con las siguientes opciones:

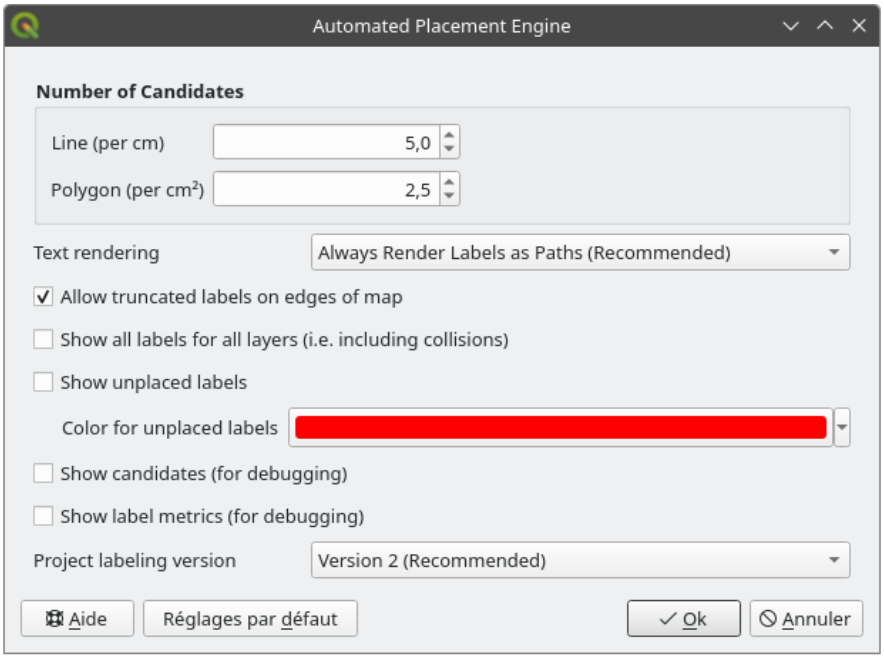


Figura 12.26: El motor de colocación automatizada de etiquetas

- *Número de candidatos*: calcula y asigna a las entidades de línea y polígono el número de posibles ubicaciones de etiquetas en función de su tamaño. Cuanto más larga o ancha es una característica, más candidatos tiene y sus etiquetas pueden colocarse mejor con menos riesgo de colisión.
- *Renderizado de texto*: establece el valor por defecto para los widgets de renderizado de etiquetas al *exportar un lienzo de mapa* o *a layout* a PDF o SVG. Si se selecciona *Renderizar siempre las etiquetas como texto*, las etiquetas pueden editarse en aplicaciones externas (por ejemplo, Inkscape) como texto normal. PERO el efecto secundario es que la calidad de renderizado disminuye, y hay problemas con el renderizado cuando hay ciertos ajustes de texto como los buffers. Por eso se recomienda *Always render labels as paths (recommended)*, que exporta las etiquetas como contornos pero garantiza una compatibilidad completa con todas las opciones de formato disponibles. Con la opción *Preferir renderizado de etiquetas como texto*, las etiquetas se renderizan como objetos de texto, a menos que al hacerlo se produzcan artefactos de renderizado o un renderizado de baja calidad (dependiendo de la configuración del formato de texto).

> **Nota**
>
> Al renderizar etiquetas como texto en un dispositivo basado en vectores (por ejemplo, PDF o SVG), se debe tener cuidado para asegurarse de que las fuentes requeridas están disponibles para los usuarios al abrir los archivos creados, o de lo contrario se utilizarán fuentes por defecto para mostrar la salida en su lugar. (Aunque las exportaciones a PDF PUEDEN incrustar automáticamente algunas fuentes cuando sea posible, dependiendo de la plataforma del usuario).

- *Allow truncated labels on edges of map*: controla si las etiquetas que quedan parcialmente fuera de la extensión del mapa se deben representar. Si está marcada, se mostrarán estas etiquetas (cuando no haya forma de colocarlas completamente dentro del área visible). Si no está marcada, se omitirán las etiquetas parcialmente visibles. Tenga en cuenta que esta configuración no tiene efectos en la visualización de las etiquetas en el *layout map item*.

- *Show all labels for all layers (i.e. including colliding objects)*. Tenga en cuenta que esta opción también se puede configurar por capa (consulte *Pestaña Renderizado*)
- *Show unplaced labels*: permite determinar si faltan etiquetas importantes en los mapas (por ejemplo, debido a superposiciones u otras restricciones). Se muestran con un color personalizable.
- *Show candidates (for debugging)*: controla si se deben dibujar cuadros en el mapa que muestre todos los candidatos generados para la colocación de etiquetas. Como dice la etiqueta, es útil solo para depurar y probar el efecto que tienen las diferentes configuraciones de etiquetado. Esto podría ser útil para una mejor colocación manual con herramientas de *label toolbar*.
- *Mostrar métricas de la etiqueta (para depuración)*: muestra los límites de texto de la etiqueta en rojo y las líneas de base en azul.
- *Versión de etiquetado del proyecto*: QGIS admite dos versiones diferentes de colocación automática de etiquetas:
  - *Versión 1*: el sistema antiguo (utilizado por QGIS versiones 3.10 y anteriores, y al abrir proyectos creados en estas versiones en QGIS 3.12 o posterior). La versión 1 trata las prioridades de las etiquetas y los obstáculos solo como «guías aproximadas», y es posible que una etiqueta de baja prioridad se coloque sobre un obstáculo de alta prioridad en esta versión. En consecuencia, puede ser difícil obtener los resultados de etiquetado deseados cuando se usa esta versión y, por lo tanto, se recomienda solo para compatibilidad con proyectos más antiguos.
  - *Versión 2 (recomendada)*: este es el sistema predeterminado en nuevos proyectos creados en QGIS 3.12 o posterior. En la versión 2, la lógica que dicta cuándo se permite que las etiquetas se superpongan *obstáculos* ha sido reelaborada. La nueva lógica prohíbe que las etiquetas se superpongan a los obstáculos con un peso de obstáculo mayor en comparación con la prioridad de la etiqueta. Como resultado, esta versión da como resultado resultados de etiquetado mucho más predecibles y más fáciles de entender.

Etiquetado basado en reglas

Con el etiquetado basado en reglas, se pueden definir múltiples configuraciones de etiqueta y aplicarlas selectivamente en la base de filtros de expresión y rango de escala, como en *Rule-based rendering*.

Para crear una regla:

1. Seleccione la opción **Etiquetado basado en reglas** en la lista desplegable principal de la pestaña :guilabel:`Labels
2. Haga clic en el botón Añadir regla en la parte inferior del diálogo.
3. Rellena el nuevo diálogo con:
   - *Descripción*: un texto utilizado para identificar la regla en la pestaña *Labels* y como entrada de la leyenda *label* en la leyenda del diseño de impresión
   - *Filtro*: una expresión para seleccionar las características a las que aplicar la configuración de la etiqueta
   - Si hay reglas ya establecidas, la opción *Sino* puede utilizarse para seleccionar todas las características que no coincidan con ningún filtro de las reglas del mismo grupo.
4. Puede establecer un *rango de escala* en la que se debe aplicar la regla de la etiqueta.
5. Las opciones disponibles en el cuadro de grupo *Etiquetas* son los habituales *ajustes de etiquetas*. Configúrelos y pulse *Aceptar*.

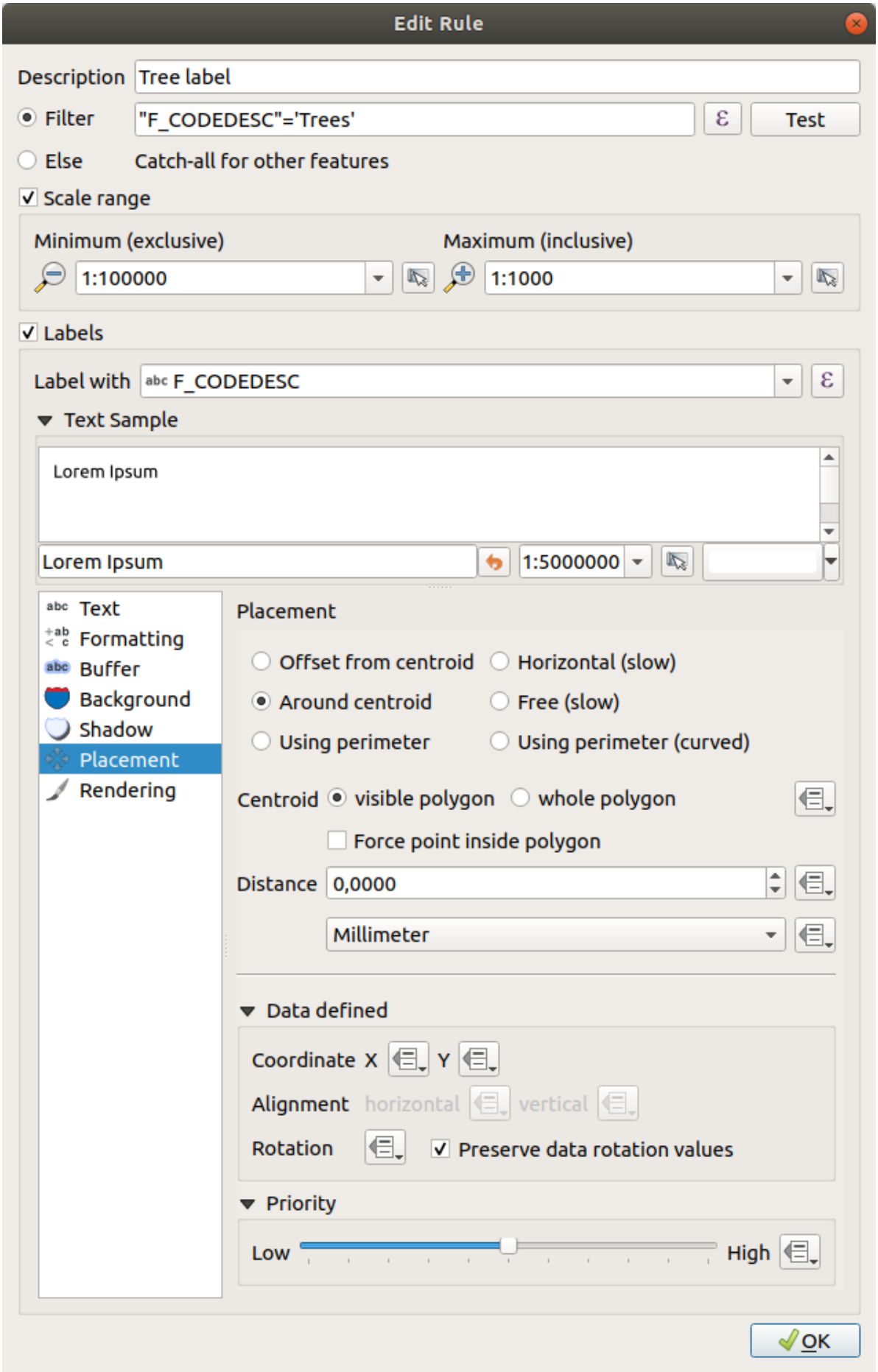


Figura 12.27: Configuración de reglas

Se muestra un resumen de las reglas existentes en el cuadro de diálogo principal (ver Figura 12.28). Puede agregar varias reglas, reordenarlas o imbricarlas con solo arrastrar y soltar. También puede eliminarlos con button or edit them with botón o un doble clic.

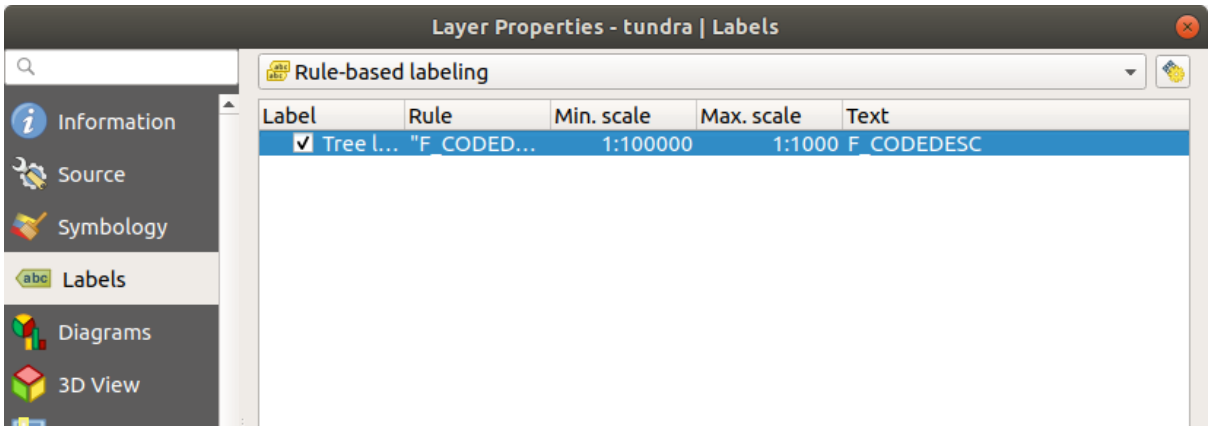


Figura 12.28: Panel de etiquetado basado en reglas

Definir etiquetas basadas en expresiones

Ya sea que elija un tipo de etiquetado único o basado en reglas, QGIS permite el uso de expresiones para etiquetar características.

Suponiendo que está utilizando el método *Single labels* ,click en el botón próximo a la lista desplegable *Value* en la pestaña *Labels* del diálogo de propiedades.

En Figura 12.29, verá una expresión de muestra para etiquetar la capa de árboles de Alaska con el tipo de árbol y el área, basada en el campo "VEGDESC", un texto descriptivo y la función `$area` en combinación con `format_number ()` para que se vea mejor.

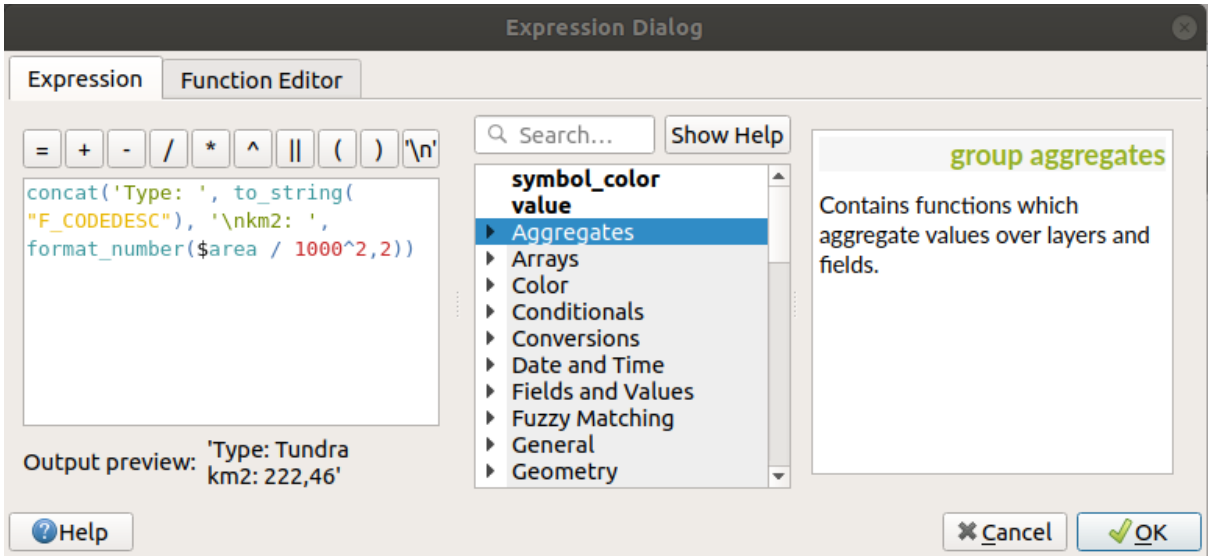


Figura 12.29: Usar expresiones para etiquetado

El etiquetado basado en expresiones es fácil de trabajar. Todo lo que tienes que cuidar es que:

- Es posible que deba combinar todos los elementos (cadenas, campos y funciones) con una función de concatenación de cadenas como `concat`, `+` o `||`. Tenga en cuenta que en algunas situaciones (cuando están involucrados valores nulos o numéricos) no todas estas herramientas se adaptarán a sus necesidades.
- Las cadenas están escritas en “single quotes”.
- Los campos están escritos en «double quotes» o sin ninguna cita.

Veamos algunos ejemplos:

1. Etiqueta basada en dos campos “name” y “place” con una coma como separador:

```
"name" || ', ' || "place"
```

   Devuelve:

```
John Smith, Paris
```

2. Etiqueta basada en dos campos “name” y “place” con otros textos:

```
'My name is ' + "name" + 'and I live in ' + "place"
'My name is ' || "name" || 'and I live in ' || "place"
concat('My name is ', name, ' and I live in ', "place")
```

   Devuelve:

```
My name is John Smith and I live in Paris
```

3. Etqueta basada en dos campos “name” y “place” con otros textos que combinan diferentes funciones de concatenación:

```
concat('My name is ', name, ' and I live in ' || place)
```

   Devuelve:

```
My name is John Smith and I live in Paris
```

   O, si el campo “place” es NULL, devuelve:

```
My name is John Smith
```

4. Etiqueta multilinea basada en dos campos “name” y “place” con un texto descriptivo:

```
concat('My name is ', "name", '\n' , 'I live in ' , "place")
```

   Devuelve:

```
My name is John Smith
I live in Paris
```

5. Etiqueta basada en un campo y la función $area para mostrar el nombre del lugar y el tamaño del área redondeada en una unidad convertida:

```
'The area of ' || "place" || ' has a size of '
|| round($area/10000) || ' ha'
```

   Devuelve:

```
The area of Paris has a size of 10500 ha
```

6. Crear una condición CASE ELSE , si el valor de la población en el campo *population* is <= 50000 es un pueblo, en otro caso es una ciudad:

```
concat('This place is a ',
CASE WHEN "population" <= 50000 THEN 'town' ELSE 'city' END)
```

Devuelve:

```
This place is a town
```

7. Nombre para mostrar de las ciudades y sin etiqueta para las otras entidades (para el contexto de «ciudad», consulte el ejemplo anterior):

```
CASE WHEN "population" > 50000 THEN "NAME" END
```

Devuelve:

```
Paris
```

Como puede ver en el generador de expresiones, tiene cientos de funciones disponibles para crear expresiones simples y muy complejas para etiquetar sus datos en QGIS. Consulte el capítulo *Expresiones* para obtener más información y ejemplos sobre expresiones.

Usando anulación definida por datos para etiquetado

Con la función Data defined override , la configuración para el etiquetado se anula mediante entradas en la tabla de atributos o expresiones fundadas en ellas. Esta función se puede usar para establecer valores para la mayoría de las opciones de etiquetado descritas anteriormente.

Por ejemplo, usando el conjunto de datos de muestra QGIS de Alaska, etiquetemos la capa `airports` con sus nombres, basado en su `USE` militar, p.ej. si el aeropuerto es accesible para:

- personas militares, luego lo muestran en color gris, tamaño 8;
- otros, luego se muestran en color azul, tamaño 10.

Para hacer esto, después de habilitar el etiquetado en el campo ``NAME"" de la capa (consulte:ref:*showlabels*):

1. Active la pestaña *Text*.
2. Click en el icono próximo a la propiedad *Size*.
3. Seleccione *Edit...* y escriba:

```
CASE
  WHEN "USE" like '%Military%' THEN 8 -- because compatible values are
↪'Military'
                                    -- and 'Joint Military/Civilian'
  ELSE 10
END
```

4. Presione *OK* para validar. El diálogo se cierra y el botón se convierte en significando que una regla está siendo aplicada.
5. Luego haga clic en el botón junto a la propiedad de color, escriba la expresión a continuación y valide:

```
CASE
  WHEN "USE" like '%Military%' THEN '150, 150, 150'
  ELSE '0, 0, 255'
END
```

Del mismo modo, puede personalizar cualquier otra propiedad de la etiqueta, de la manera que desee. Ver más detalles sobre los widget's Data-define override descripción y manipulación en la sección *Configuración de anulación definida por datos*.

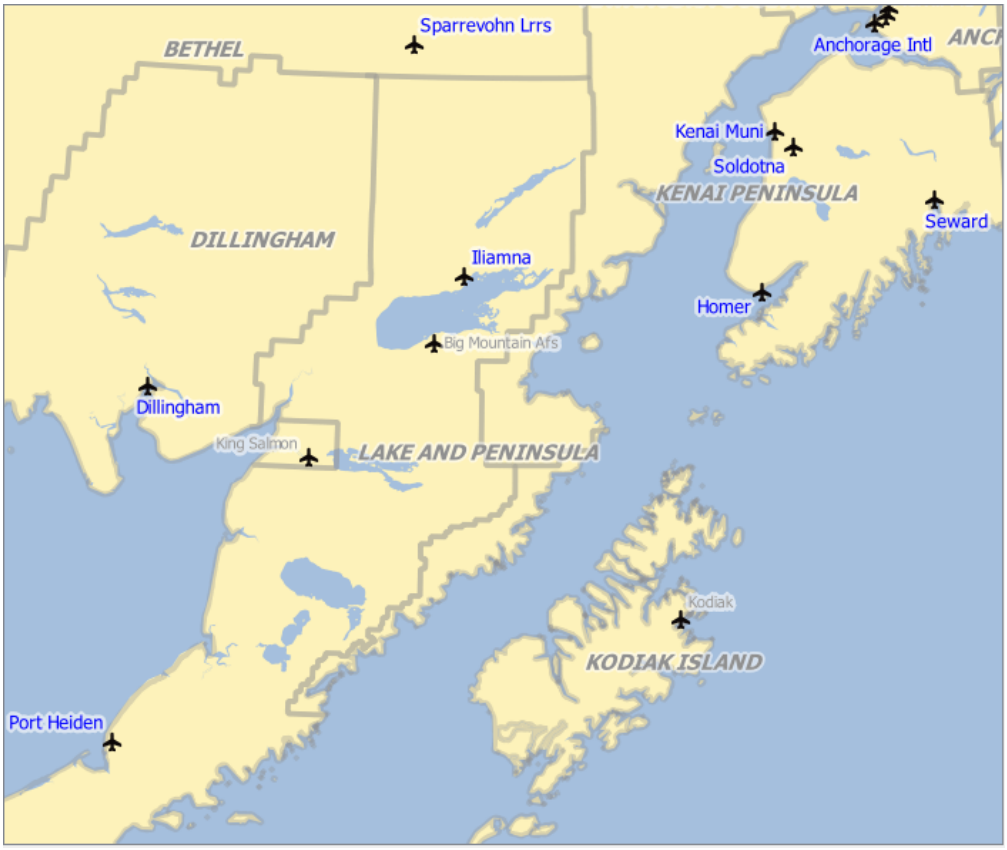


Figura 12.30: Las etiquetas de los aeropuertos están formateadas en función de sus atributos

Truco

Use la anulación definida por datos para etiquetar cada parte de las geometrías de entidades multiparte

Hay una opción para establecer el etiquetado de las características de varias partes independientemente de las propiedades de su etiqueta. Elija *Rendering*, `Feature options`,vaya al botón Data-define override próximo al checkbox *Label every part of multipart-features* y defina las etiquetas como se describe en *Configuración de anulación definida por datos*.

La Barra de Herramientas Etiqueta

El *Label Toolbar* proporciona algunas herramientas para manipular *label* (incluyendo su *callout*) o *diagram* propiedades:



Figura 12.31: La barra de herramientas Etiqueta

- Highlight Pinned Labels, Diagrams and Callouts. Si la capa vectorial del elemento es editable, el resaltado es verde; de lo contrario, es azul.
- Toggle Display of Unplaced Labels: Permite determinar si faltan etiquetas importantes en los mapas (por ejemplo, debido a superposiciones u otras limitaciones). Se muestran con un color personalizable (ver *Configurar el*

motor de colocación automatizado).

- Pin/Unpin Labels and Diagrams. Al hacer clic o arrastrar un área, anclar elementos superpuestos. Si hace clic o arrastra un área sosteniendo `Shift`, los elementos se desanclan. Finalmente, también puede hacer clic o arrastrar un área sosteniendo `Ctrl` para alternar su estado de pin.
- Show/Hide Labels and Diagrams. Si hace clic en los elementos, o hace clic y arrastra un área sosteniendo `Shift`, se ocultan. Cuando un elemento está oculto, solo tiene que hacer clic en la función para restaurar su visibilidad. Si arrastra un área, se restaurarán todos los elementos del área.
- Move a Label, Diagram or Callout: haga clic para seleccionar el elemento y haga clic para moverlo al lugar deseado. Las nuevas coordenadas se almacenan en *auxiliary fields*. Al seleccionar el elemento con esta herramienta y presionar la tecla `Delete`, se eliminará el valor de posición almacenado.
- Rotate a Label. Haga clic para seleccionar la etiqueta y haga clic nuevamente para aplicar la rotación deseada. Asimismo, el nuevo ángulo se almacena en un campo auxiliar. Seleccionar una etiqueta con esta herramienta y presionar la tecla `Delete` eliminará el valor de rotación de esta etiqueta.
- Cambiar las propiedades de Etiqueta. Abre un diálogo para cambiar las propiedades de la etiqueta en la que se hizo click; puede ser la propia etiqueta, sus coordenadas, ángulo, tipo de letra, tamaño, alineación multilínea … siempre que esta propiedad se haya mapeado en un campo. Aquí puede establecer la opción en *Etiqueta cada parte de una entidad*.

Advertencia

Las herramientas de etiquetas sobrescriben los valores de campo actuales

Usando la *Label toolbar* para personalizar el etiquetado, en realidad escribe el nuevo valor de la propiedad en el campo asignado. Por lo tanto, tenga cuidado de no reemplazar accidentalmente los datos que pueda necesitar más adelante.

Nota

El mecanismo *Propiedades de almacenamiento auxiliar* puede usarse para personalizar el etiquetado (posición, etc.) sin modificar la fuente de datos subyacente.

Personaliza las etiquetas desde el lienzo del mapa

En combinación con *Label Toolbar*, la configuración de anulación de datos definidos le ayuda a manipular las etiquetas en el lienzo del mapa (mover, editar, rotar). Ahora describimos un ejemplo usando la función de anulación definida por datos para la función Move Label, Diagram or Callout (ver Figura 12.32).

1. Importe `lakes.shp` desde el conjunto de datos de muestra de QGIS.
2. Doble-click en la capa para abrir as propiedades de capa. Click en *Labels* y *Placement*. Seleccione *Offset from centroid*.
3. Busque las entradas *Data defined*. Click en el icono para definir el tipo de campo para la *Coordinate*. Escoja `xlabel` para X y `ylabel` para Y. Los iconos ahora están resaltados en amarillo.

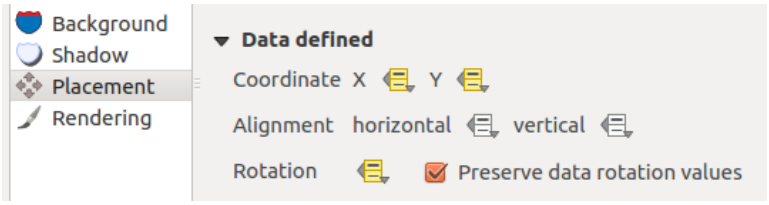


Figura 12.32: Etiquetado de capas de polígonos vectoriales con anulación definida por datos

4. Acercamiento a un lago.
5. Establezca la capa editable utilizando el botón Toggle Editing.
6. Vaya a la barra de herramientas Etiqueta y haga click en el icono . Ahora puede cambiar la etiqueta manualmente a otra posición (ver Figura 12.33). La nueva posición de la etiqueta se guarda en las columnas `xlabel` y `ylabel` de la tabla de atributos.
7. Es posible agregar un lúnea conectando cada lago a su etiqueta desplazada usando:
   - la *propiedad llamada* de la etiqueta
   - o el *generador de geometría para capa de símbolo* con la expresión de abajo:

```
make_line( centroid( $geometry ), make_point( "xlabel", "ylabel" ) )
```

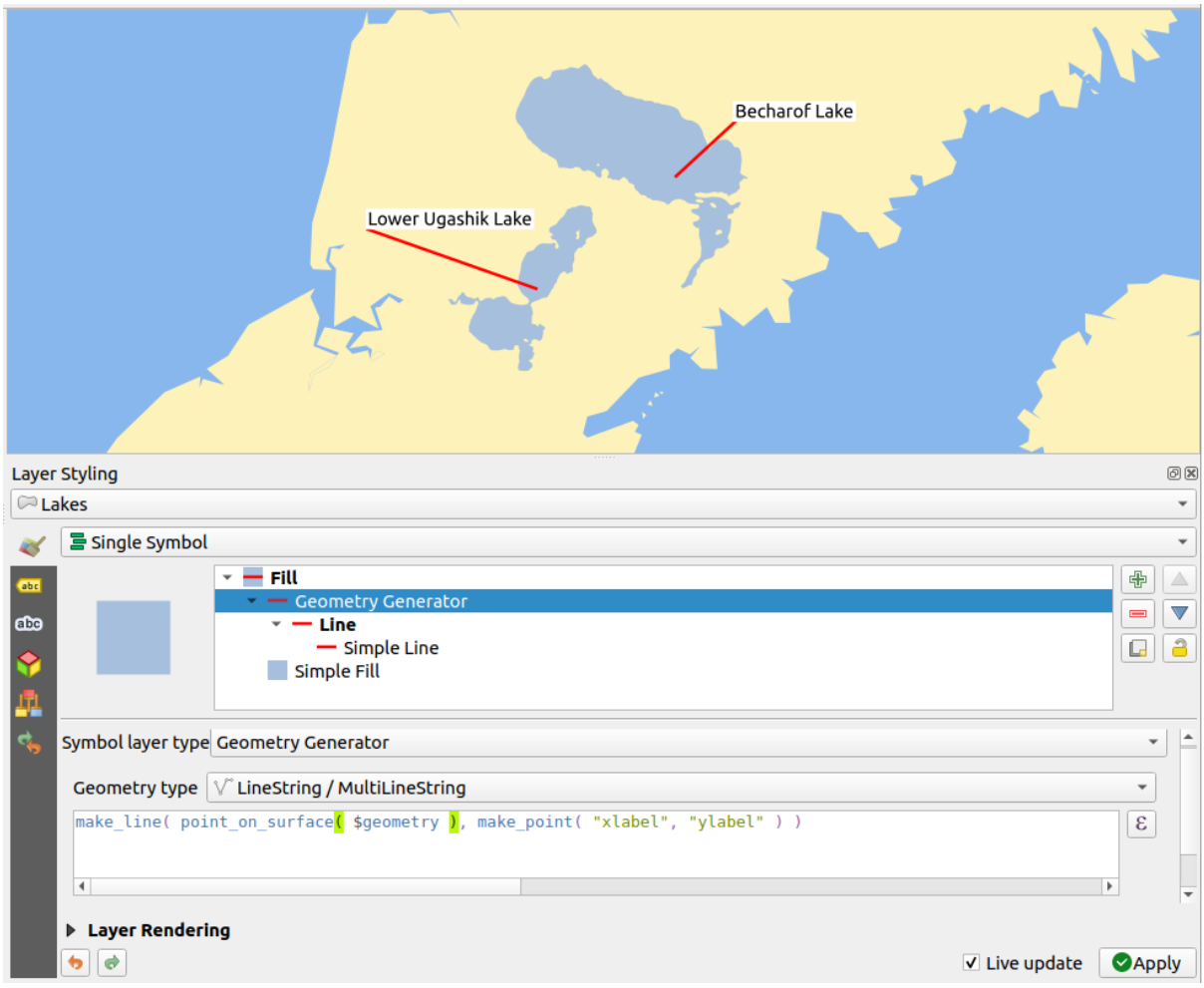


Figura 12.33: Etiquetas desplazadas

> **Nota**
>
> El mecanismo *Propiedades de almacenamiento auxiliar* puede usarse con propiedades definidas por datos sin tener una fuente de datos editable.

12.1.5 Propiedades de diagrama

La pestaña  :guilabel:`Diagramas` permite añadir una superposición gráfica a una capa vectorial (véase Figura 12.34). También se puede acceder a este diálogo desde el panel *Estilos de capa*, o utilizando el botón  Opciones de diagrama de capa de la barra de herramientas **Etiquetas**.

El núcleo de implementación de diagramas actual proporciona soporte para:

-  *No diagrams*: el valor predeterminado sin diagrama que se muestra sobre las entidades;
-  *Gráfico de tarta*, un gráfico estadístico circular dividido en porciones para ilustrar la proporción numérica. La longitud del arco de cada porción es proporcional a la cantidad que representa;
- abc *Diagrama de texto*, un círculo dividido horizontalmente que muestra los valores estadísticos en su interior;
-  *Histograma*, barras de distintos colores para cada atributo alineadas una al lado de la otra;
-  *Barras apiladas*, apila barras de distintos colores para cada atributo unas sobre otras vertical u horizontalmente;
-  *Diagrama apilado*, apila diagramas de tipos iguales o distintos, uno junto a otro, vertical u horizontalmente. Más detalles en *Diagramas apilados*.

En la esquina superior derecha de la pestaña *Diagrams* , el botón  Automated placement settings (applies to all layers) proporciona medios para controlar el diagrama *labels placement* en el mapa del lienzo.

> **Truco**
>
> **Cambiar rápidamente entre tipos de diagramas**
>
> Dado que la configuración es casi común para los diferentes tipos de diagrama, al diseñar su diagrama, puede cambiar fácilmente el tipo de diagrama y verificar cuál es más apropiado para sus datos sin ninguna pérdida.

Para cada tipo de diagrama, las propiedades se dividen en varias pestañas:

- *Attributes*
- *Rendering*
- *Size*
- *Colocación*
- *Options*
- *Legend*

Atributos

Attributes define qué variables mostrar en el diagrama. Use el botón add item para seleccionar los campos deseados en el panel “Assigned Attributes”. Atributos generados con *Expresiones* también puede ser usado.

Puede mover hacia arriba y hacia abajo cualquier fila haciendo clic y arrastrando, ordenando cómo se muestran los atributos. También puede cambiar la etiqueta en la columna “Legend” o el color del atributo haciendo doble clic en el elemento.

Esta etiqueta es el texto predeterminado que se muestra en la leyenda del diseño de impresión o del árbol de capas.

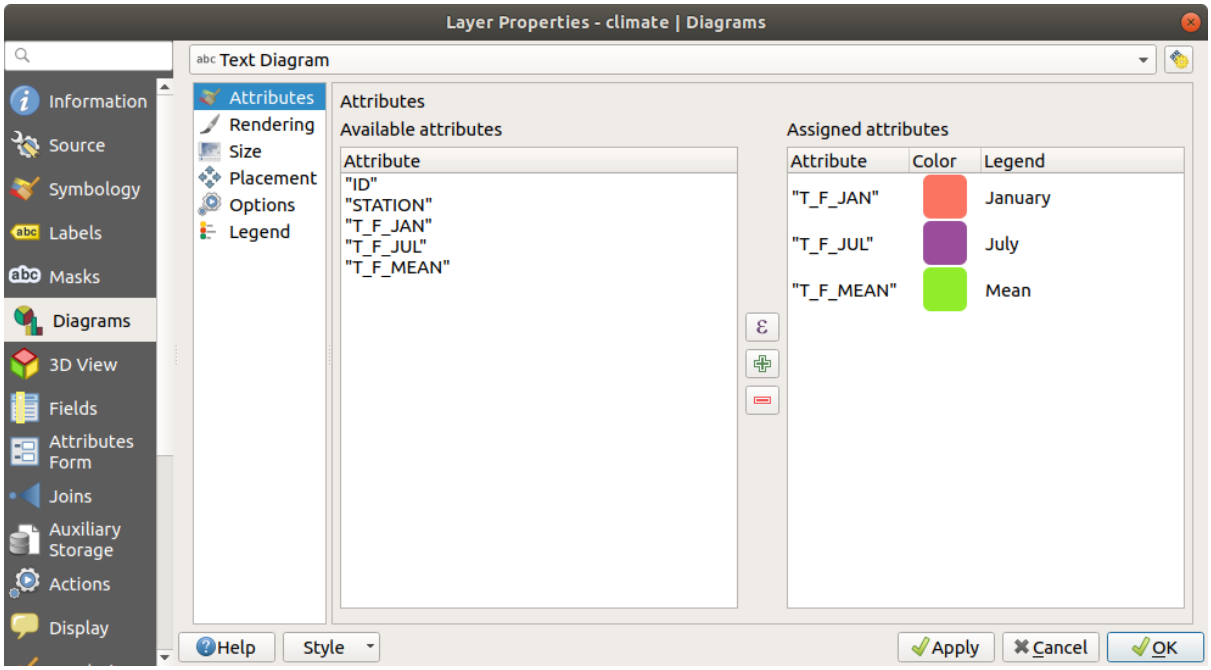


Figura 12.34: Propiedades de Diagrama- Pestaña Atributos

Representación

Rendering define cómo se ve el diagrama. Proporciona configuraciones generales que no interfieren con los valores estadísticos como:

- la opacidad de los gráficos, su borde y color;
- dependiendo del tipo de diagrama:
  - para histograma y barras apiladas, el ancho de la barra y el espacio entre las barras. Es posible que desee establecer el espaciado en `0` para las barras apiladas. Además, el *Símbolo de línea del eje* se puede hacer visible en el lienzo del mapa y personalizarlo usando *propiedades del símbolo de línea*.
  - para diagrama de texto, el color de fondo del círculo y la *fuente* usada para los textos;
  - para gráficos de tarta, el *ángulo de Inicio* de la priemra porción y su *Dirección* (sentido horario o no).
- el uso de *efectos de pintura* en los gráficos.

En esta petaña, también puede administrar y ajustar la visibilidad del diagrama con diferentes opciones:

- *Diagrama índice-z*: controla cómo se dibujan los diagramas unos sobre otros y sobre las etiquetas. Un diagrama con un índice alto se dibuja sobre otros diagramas y etiquetas;
- *Show all diagrams*: muestra todos los diagramas incluso si se superponen entre sí;
- *Show diagram*: solo permite renderizar diagramas específicos;

- *Always Show*: selecciona diagramas específicos para representar siempre, incluso cuando se superponen con otros diagramas o etiquetas de mapa;
- ajustando la *Scale dependent visibility*;

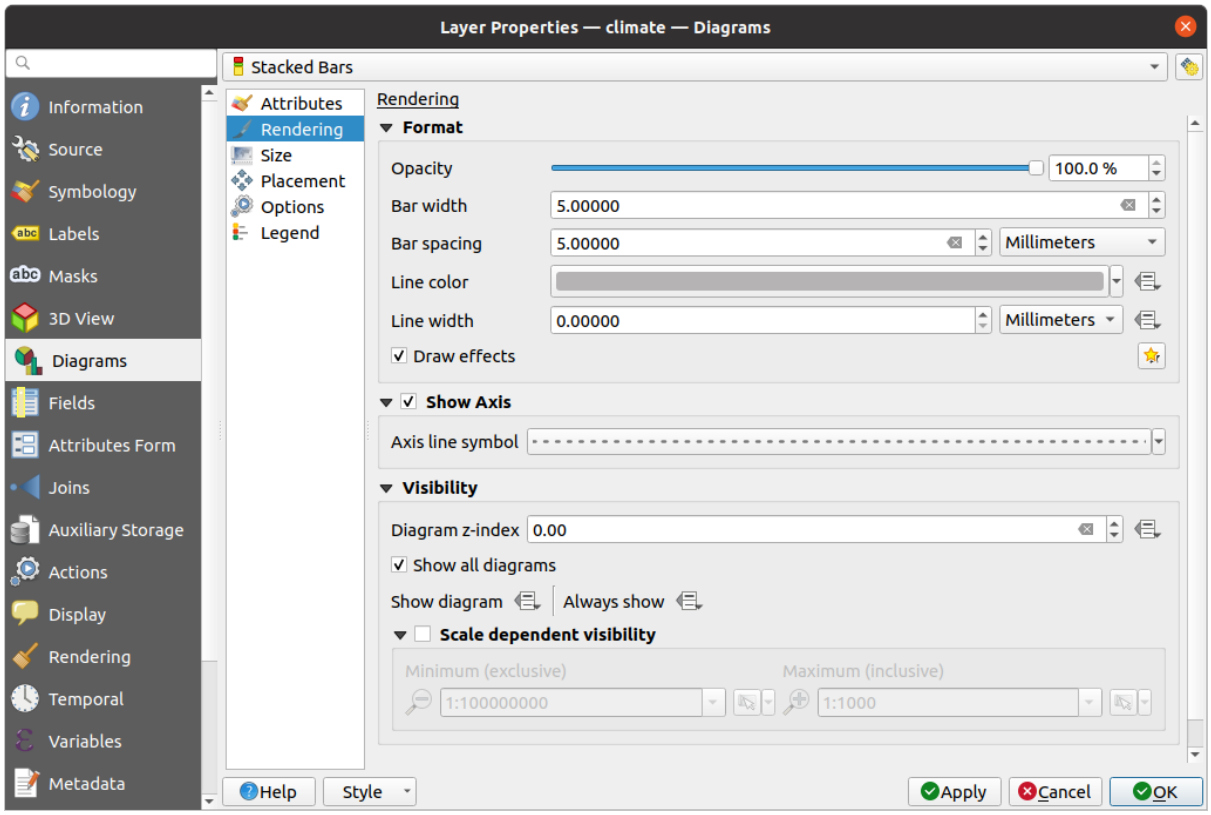


Figura 12.35: Propiedades del diagrama - pestaña Renderizado

Tamaño

Tamaño es la pestaña principal para establecer cómo se representan las estadísticas seleccionadas. Las *unidades* de tamaño del diagrama pueden ser “Milímetros, “Puntos”, “Pixels”, “Unidades de Mapa” o “Pulgadas”. Puede usar:

- *tamaño Fijo*, un tamaño único para representar el gráfico de todas las entidades (no disponible para histogramas)
- o *tamaño Escalado*, basado en una expresión usando atributos de capa:
  1. En *Atributos*, seleccione un campo o construya una expresión
  2. Presione *Encontrar* para devolver el *valor Máximo* del atributo o ingrese un valor personalizado en el control.
  3. Para el histograma y las barras apiladas, introduzca un valor *Longitud de barra*, utilizado para representar el *Valor máximo* de los atributos. Para cada característica, la longitud de la barra se escalará linealmente para mantener esta correspondencia.
  4. Para gráficos de tarta y diagramas de texto, ingrese un valor *Tamaño*, usado para representar el *Valor máximo* de los atributos. Para cada objeto, el área o diámetro del círculo se escalará linealmente para mantener esta coincidencia (desde `0`). Sin embargo, el *tamaño Mínimo* se puede establecer para diagramas pequeños.

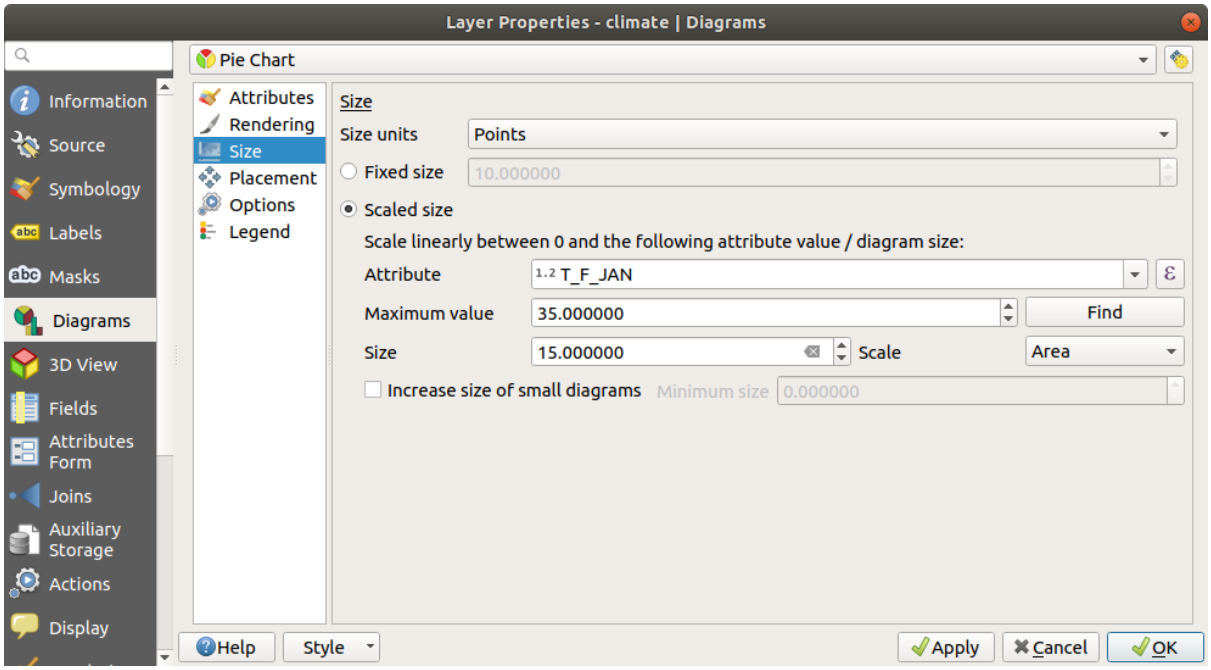


Figura 12.36: Propiedades de diagrama - pestaña de tamaño

Ubicación

Ubicación define la posición del diagrama. Dependinendo del tipo de geometría de capa, oferce diferentes opciones para la ubicación (mas detalles en *Ubicación*):

- *En torno al punto* o *Sobre el punto* para geometría de punto. La primera variable requiere un radio a mantener.
- *En torno a la línea* o *Sobre la línea* para geometría lineal. Al igual que entidad de punto, la primera variable requiere una distancia para respetar y puede especificar la ubicación del diagrama en relación con la característica (“arriba”, “en” y/o “debajo” de la línea) Es posible seleccionar varias opciones a la vez. En ese caso, QGIS buscará la posición óptima del diagrama. Recuerde que también puede utilizar la orientación de la línea para la posición del diagrama.
- *En torno al Centroide* (a una *Distancia* establecida), *Sobre le Centroide*, *Usando prímetro* y *Dentro del polígono* son las opciones para entidades poligonales.

El grupo *Coordenada* proporciona control directo sobre la ubicación del diagrama, de forma entidad-por-entidad, usando sus atributos o una expresión para establecer la coordenada *X* e *Y*. La información también se puede rellenar usando la herramienta *Mover etiquetas y diagramas*.

En la sección *Prioridad*, puede definir el rango de prioridad de colocación de cada diagrama, es decir, si hay diferentes diagramas o etiquetas candidatos para la misma ubicación, se mostrará el elemento con mayor prioridad y los demás podrían quedar fuera.

Desaconsejar que los diagramas y etiquetas cubran las características define los objetos espaciales a utilizar como *obstáculos*, i.e. QGIS intentará no colocar diagramas ni etiquetas sobre estos objetos espaciales. El rango de prioridad se utiliza entonces para evaluar si un diagrama podría omitirse debido a un objeto espacial de obstáculo de mayor peso.

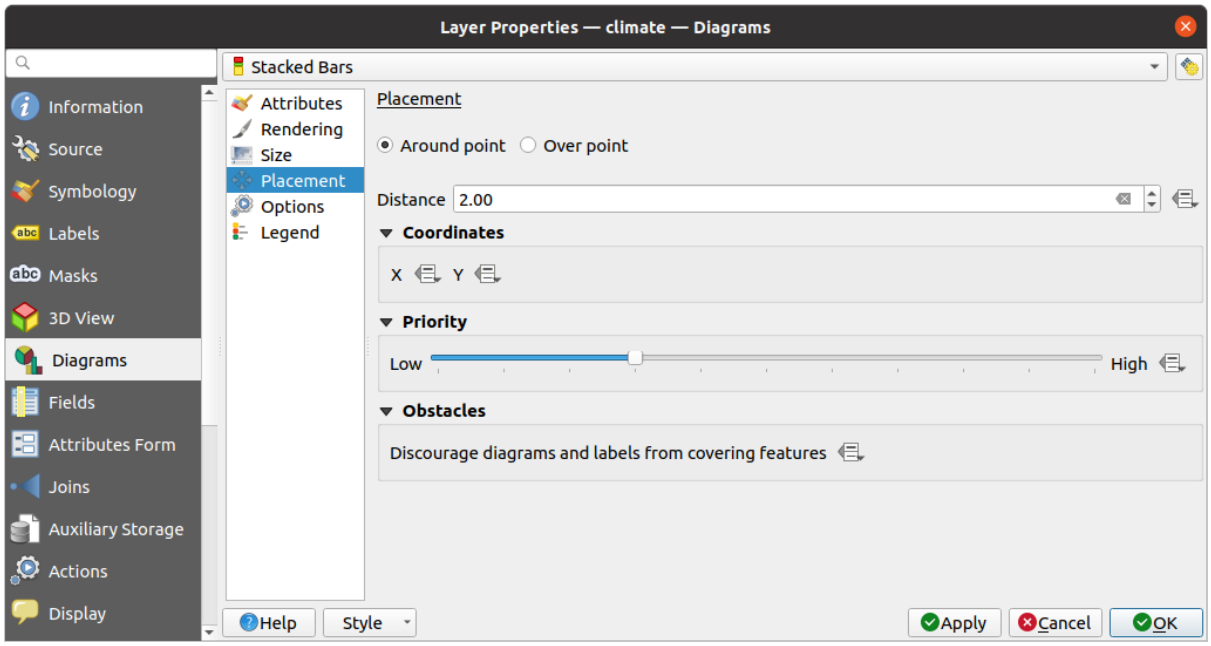


Figura 12.37: Diálogo de propiedades de vector con propiedades de diagrama, pestaña Colocación

Opciones

La pestaña *Opciones* tiene configuraciones para histogramas y barras apiladas. Puede elegir si la *Orientación de la barra* debe ser *Arriba*, *Abajo*, *Derecha* o *Izquierda*, para diagramas horizontales y verticales.

Leyenda

En la pestaña de *Leyenda*, puede elegir mostrar elementos del diagrama en el *panel de Capas*, y en la *leyenda del compositor de impresión*, al lado de simbología de capa:

- marque *Show legend entries for diagram attributes* para mostrar en la leyenda las propiedades `Color` y `Legend` , como asignó previamente en la pestaña *Attributes* ;
- y, cuando un *scaled size* es usada para los diagramas, pulse el botón *Legend Entries for Diagram Size…* para configurara el aspecto de la simbología del diagrama en la leyenda. Esto abre el diálogo *Data-defined Size Legend* cuyas opciones se describen en *Leyenda de tamaño definido por datos*.

Cuando se establece, los elementos de la leyenda del diagrama (atributos con color y tamaño del diagrama) también se muestran en la leyenda del diseño de impresión, junto a la simbología de la capa.

Diagramas apilados

Los diagramas apilados permiten a los usuarios crear diagramas complejos, como pirámides de población, en los que dos subdiagramas, en concreto histogramas, se sitúan uno al lado del otro y se muestran horizontalmente.

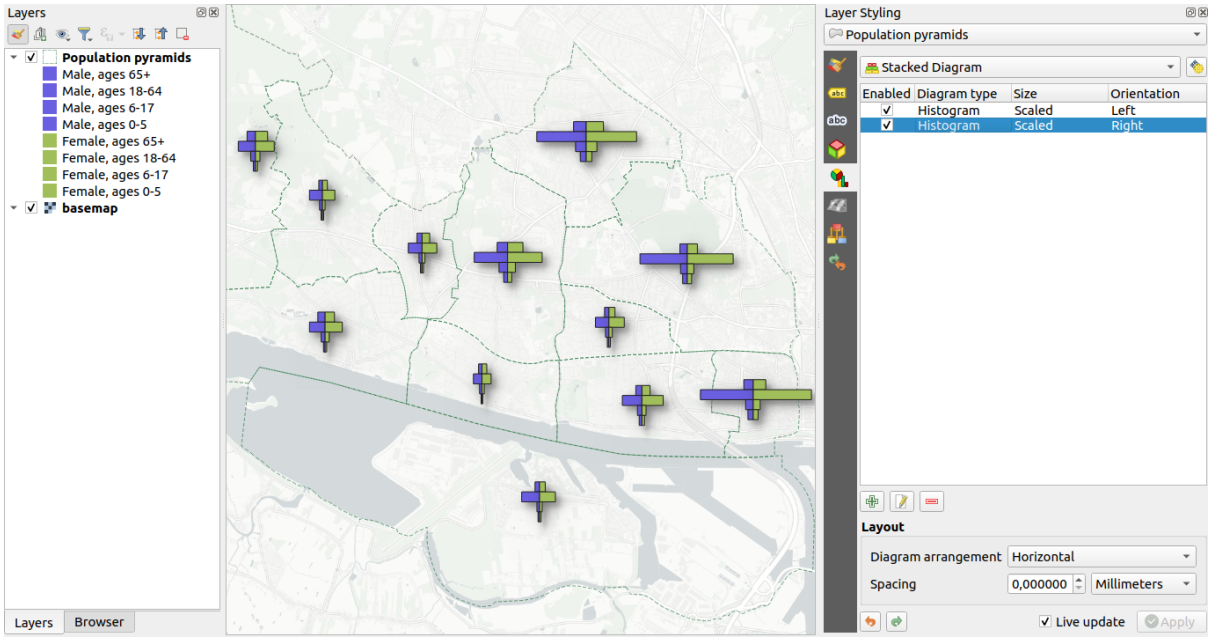


Figura 12.38: Pirámides de población construidas para cada objeto espacial de la capa

Los diagramas multitemporales también pueden construirse como diagramas apilados. Se puede configurar el número de subdiagramas, así como el espaciado entre ellos.

Además, los subdiagramas pueden ser de distintos tipos (por ejemplo, un gráfico circular junto a un histograma) y tener sus propios ajustes independientes como *Atributos*, *Renderizado*, *Tamaño*, *Opciones* y *Leyenda*.

ajustes de *Colocación* en un diagrama apilado, así como algunos ajustes de visibilidad (ubicados en la pestaña *Renderizado*), vienen determinados por los ajustes de colocación y visibilidad del primer subdiagrama de la pila.

Por último, el orden de los subdiagramas viene dado por el orden de los elementos en la lista del diagrama apilado. El primer subdiagrama aparece a la izquierda en un diagrama apilado horizontal, o en la parte superior de uno vertical.

12.1.6 Propiedades de Máscaras

La pestaña *Máscaras* le ayuda a configurar la superposición de los símbolos de la capa actual con otras capas de símbolos o etiquetas, de cualquier capa. Esto está pensado para mejorar la legibilidad de los símbolos y etiquetas cuyos colores están próximos y pueden ser difíciles de descifrar cuando se superponen; añade una máscara personalizada y transparente alrededor de los elementos para «ocultar» partes de las capas de símbolos de la capa actual.

Para aplicar máscaras en la capa activa, primero debe habilitar en el proyecto *capas de símbolo de máscara* o *etiquetas de máscara*. Luego, de la pestaña *Máscara*, marque:

- Las *Capas de símbolo enmascaradas*: enumera en una estructura de árbol todas las capas de símbolo de la capa actual. Allí puede seleccionar el elemento de la capa de símbolo que le gustaría «recortar» de forma transparente cuando se superponen a las fuentes de máscara seleccionadas
- la pestaña *Fuentes de máscara*: lista todas las etiquetas de máscara y capas de símbolo de máscara definidas en el proyecto. Seleccione los elementos que generarían la máscara sobre las capas de símbolo enmascarado seleccionadas

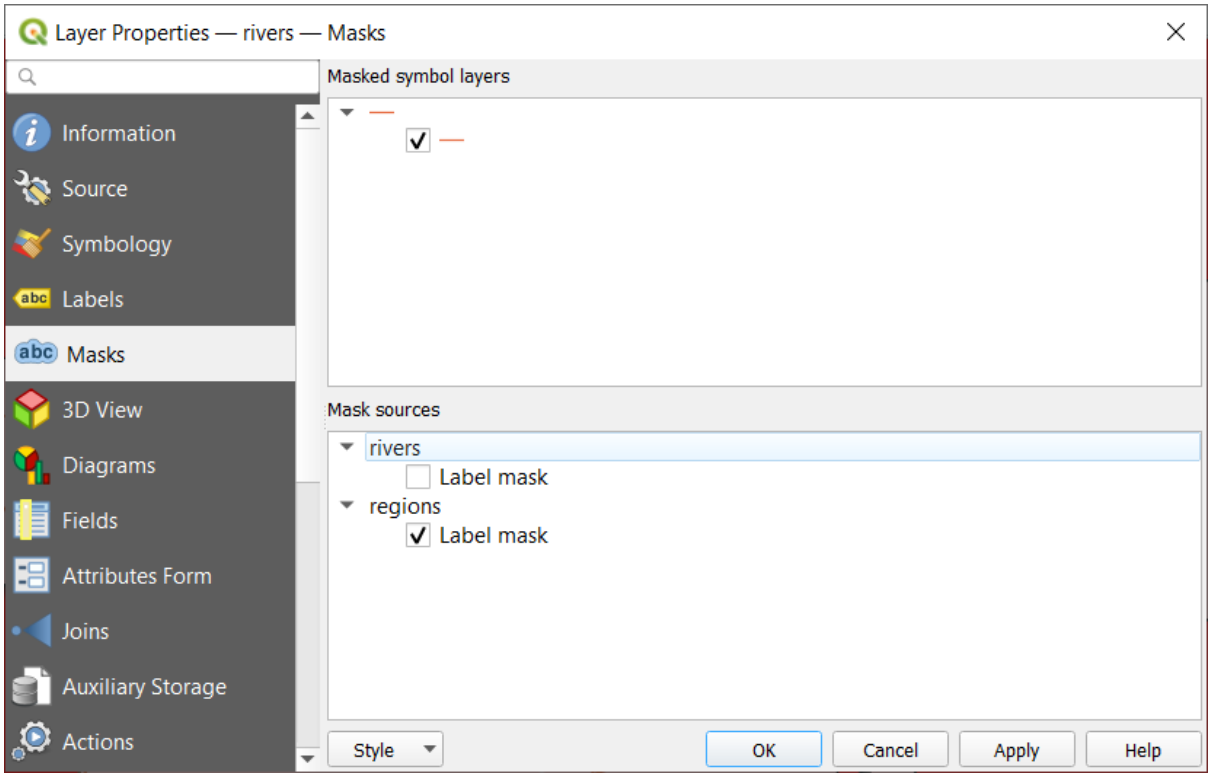


Figura 12.39: Propiedades de capa - pestaña Máscaras

12.1.7 Propiedades de visor 3D

La pestaña  *Vista 3D* proporciona la configuración de las capas vectoriales que deben representarse en la herramienta *vista Mapa 3D*.

Para mostrar una capa en 3D, seleccione del cuadro combinado en la parte superior de la pestaña, ya sea:

- *Símbolo único*: las entidades se representan utilizando un símbolo 3D común cuyas propiedades pueden ser *definido por datos* o no. Leer detalles en *configurar un símbolo 3D* para cada tipo de geometría de capa.
- *Rule-based*: Se pueden definir múltiples configuraciones de símbolos y aplicarlas selectivamente en función de los filtros de expresión y el rango de escala. Más detalles sobre cómo hacerlo en *Rule-based rendering*.

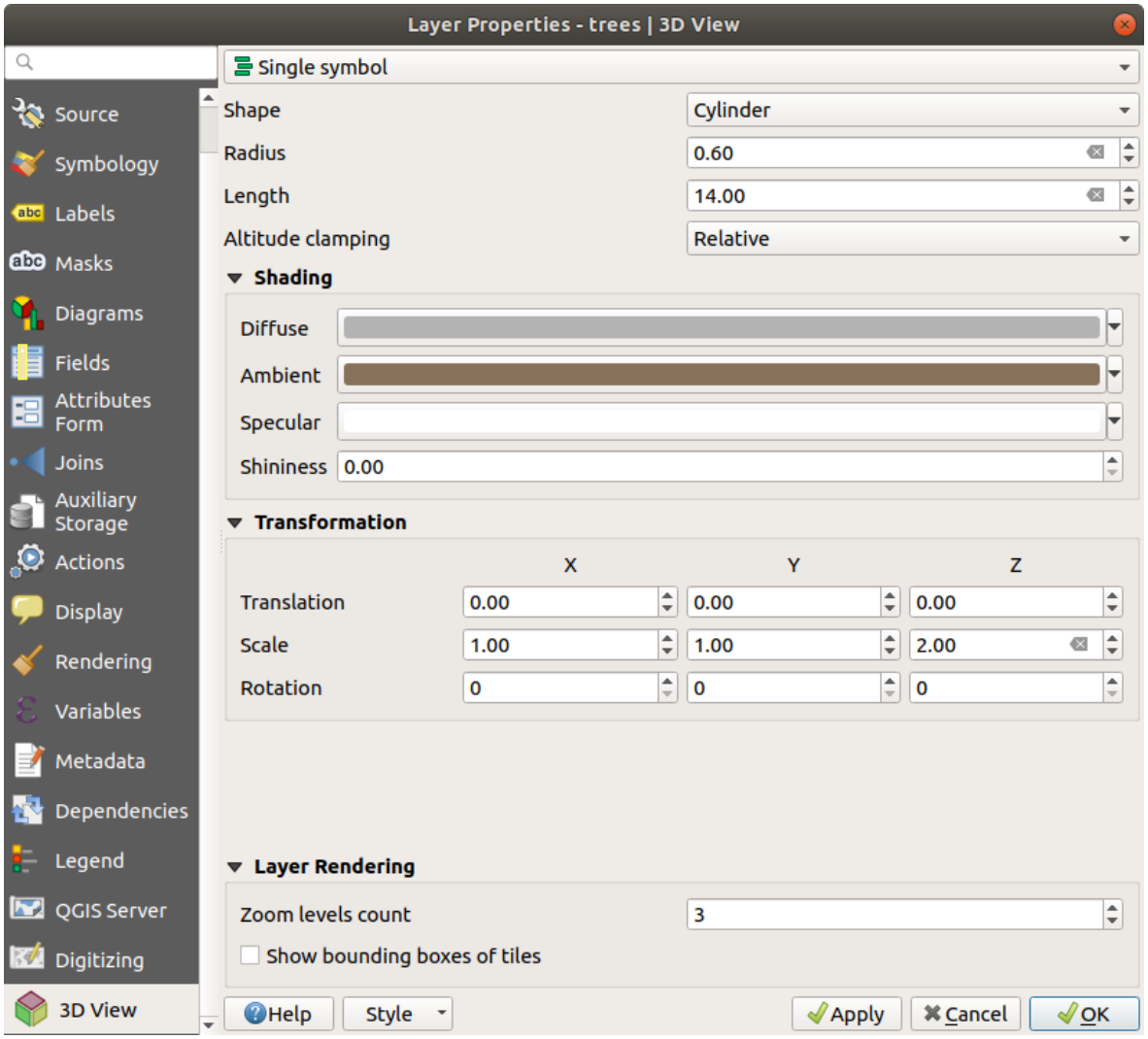


Figura 12.40: Propiedades 3D de una capa de punto

Atención

Prefiera la *Elevación* pestaña para la elevación del símbolo y la configuración del terreno

Las propiedades relacionadas con la elevación y la altitud de los objetos espaciales (*Fijación de altitud*, *Fijación de altitud*, *Extrusión* o *Altura*) en la pestaña *3D View* heredan sus valores por defecto de las propiedades *Elevation* y debe establecerse preferentemente desde la pestaña *Elevación*.

Para un mejor rendimiento, los datos de las capas vectoriales se cargan en segundo plano, utilizando subprocesos múltiples, y se renderizan en mosaicos cuyo tamaño se puede controlar desde la sección *máscaras de capa* de la pestaña:

- *Recuento de niveles de zoom*: determina la profundidad del quadtree. Por ejemplo, un nivel de zoom significa que habrá un solo mosaico para toda la capa. Tres niveles de zoom significan que habrá 16 mosaicos en el nivel de la hoja (cada nivel de zoom adicional lo multiplica por 4). El valor predeterminado es `3` y el máximo es `8`.
- *Mostrar las cajas delimitadoras de las teselas*: especialmente útil si hay problemas con baldosas que no se muestran cuando deberían.

12.1.8 Propiedades de campos

La pestaña *Campos* proporciona información sobre los campos relacionados con la capa y le ayuda a organizarlos.

La capa se puede hacer *editable* using the Toggle editing mode. En este momento, puede modificar su estructura utilizando los botones New field y Delete field.

Al crear Nuevo campo, la opción *Comentario* sólo está disponible para las fuentes de datos que permiten editar comentarios (Véase *Entradas de la base de datos* para más detalles). También puede establecer alias dentro del diálogo *Añadir campo*, para los formatos OGR admitidos (GeoPackage y ESRI File Geodatabase).

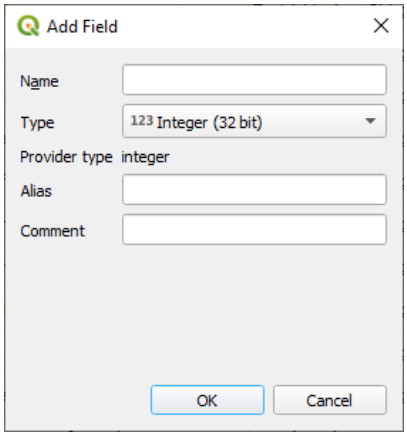


Figura 12.41: Diálogo Añadir campo

También puede cambiar el nombre de los campos haciendo doble clic en su nombre. Esto sólo es compatible con proveedores de datos como PostgreSQL, Oracle, la capa de memoria y algunas capas GDAL dependiendo de la versión de GDAL.

Si se establece en la fuente de datos subyacente o en las *forms properties*, también se muestra el alias del campo. Un alias es un nombre de campo legible por humanos que puede usar en el formulario de características o en la tabla de atributos. Los alias se guardan en el archivo del proyecto.

Aparte de los campos contenidos en el conjunto de datos, *virtual fields* y *Auxiliary Storage* incluidos, la pestaña *Fields* también enumera los campos de cualquier *joined layers*. Dependiendo del origen del campo, se le aplica un color de fondo diferente.

Para cada campo enumerado, el cuadro de diálogo también enumera características de solo lectura, como su *Type*, *Type name*, *Length* y *Precision*`.

Dependiendo del proveedor de datos, puede asociar un comentario con un campo, por ejemplo, en su creación. Esta información se recupera y se muestra en la columna *Comment* y luego se muestra al pasar el mouse sobre la etiqueta del campo en un formulario de entidades.

En la columna *Configuration*, puedes establecer cómo debe comportarse el campo en determinadas circunstancias:

- `Not searchable`: marque esta opción si no desea que este campo sea consultado por el *search locator bar*
- `Do not expose via WMS`: marque esta opción si no desea mostrar este campo si la capa se sirve como WMS desde el servidor QGIS
- `Do not expose via WFS`: marque esta opción si no desea mostrar este campo si la capa se sirve como WFS desde el servidor QGIS

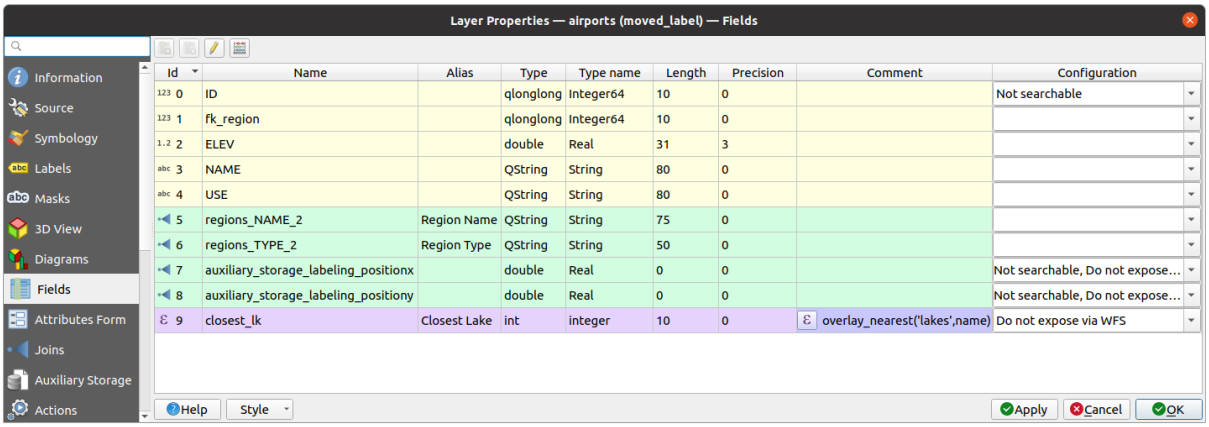


Figura 12.42: Pestaña de propiedades de campo

12.1.9 Formulario de propiedades de campo

La pestaña *Formulario de atributos* le ayuda a configurar el formulario que se mostrará al crear nuevas características, editar o consultar las existentes. Esto afecta tanto a las vistas de tabla como a las de formulario. <attribute_table_view>`. Puede definir:

- el aspecto y el comportamiento (etiqueta, visibilidad, widget, restricciones…) de cada campo en la capa actual, las *capas unidas* y las *relacionales*;
- la ubicación de las habilitadas *acciones*;
- la estructura del formulario (personalizada o generada automáticamente);
- HTML adicional, QML, widgets de texto para mejorar el formulario;
- lógica adicional en Python para manejar la interacción con los widgets de formulario o campo.

En la parte superior derecha del cuadro de diálogo, puede establecer si el formulario se abre de forma predeterminada al crear nuevas funciones. Esto se puede configurar por capa o globalmente con la opción *Suppress attribute form pop-up after feature creation* en el menú *Settings* ► *Options* ► *Digitizing*.

Personalizando un formulario para sus datos

De manera predeterminada, cuando hace clic en una función con la herramienra Identify Features o cambia la tabla de atributos al modo *form view*, QGIS muestra un formulario básico con widgets predefinidos (generalmente spinboxes y textboxes — cada campo es representado en una fila dedicada por su etiqueta al lado del widget). Si las *relations* están establecidas en la capa, los campos de las capas referenciadas se muestran en un marco embebido en la parte inferor del formulario, siguiendo la misma estructura básica.

Esta representación es el resultado del valor predeterminado `Autogenerate` de la configuración *Attribute editor layout* en la pestaña *Layer properties* ► *Attributes Form*. Esta propiedad tiene tres valores diferentes:

- `Autogenerate`: keeps the basic structure of «one row - one field» para el formulario pero permite personalizar cada widget correspondiente.
- `Drag-and-drop designer`: aparte de la personalización de widgets, la estructura del formulario se puede hacer más compleja, por ejemplo, con widgets incrustados en grupos y pestañas.
- `Provide ui file`: permite utilizar un archivo de diseño Qt, por lo tanto, una plantilla potencialmente más compleja y con todas las funciones, como formulario de funciones.

Just below the *Attribute editor layout* setting, a search box allows you to filter fields, relations, containers and other available widgets.

For items that support aliases, like fields or relations, the filter targets both names and aliases.

The Show field aliases instead of names button allows you to toggle field aliases and field names.

When showing aliases, fields without an alias set will have their names displayed in light gray to help identify them easily.

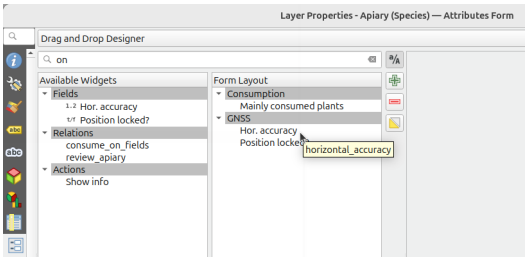


Figura 12.43: Search fields using aliases and names in Attributes Form page

El formulario autogenerado

Cuando la opción `Autogenerate` está activada, el panel *Widgets disponibles* muestra listas de campos (de la capa actual, las capas de unión y relación) y las acciones habilitadas que se mostrarían en el formulario. Seleccione un campo y podrá configurar su apariencia y comportamiento en el panel derecho:

- añadiendo *custom label and automated checks* al campo;
- ajustando a *particular widget* para usar.

En este modo de editor, solo se pueden ver las propiedades de las acciones. Además, se mostrarán a través de un menú desplegable *Acciones* en el formulario de características y, siguiendo la propiedad *Mostrar en la tabla de atributos*, en otros contextos.

El diseñador copiar y pegar

El diseñador de arrastrar y soltar le permite crear un formulario con varios contenedores (pestañas o grupos) para presentar los campos de atributos u otros widgets que no están directamente vinculados a un campo en particular (como los widgets HTML/QML o las *actions* definido para la capa), como se muestra, por ejemplo, en Figura 12.44.

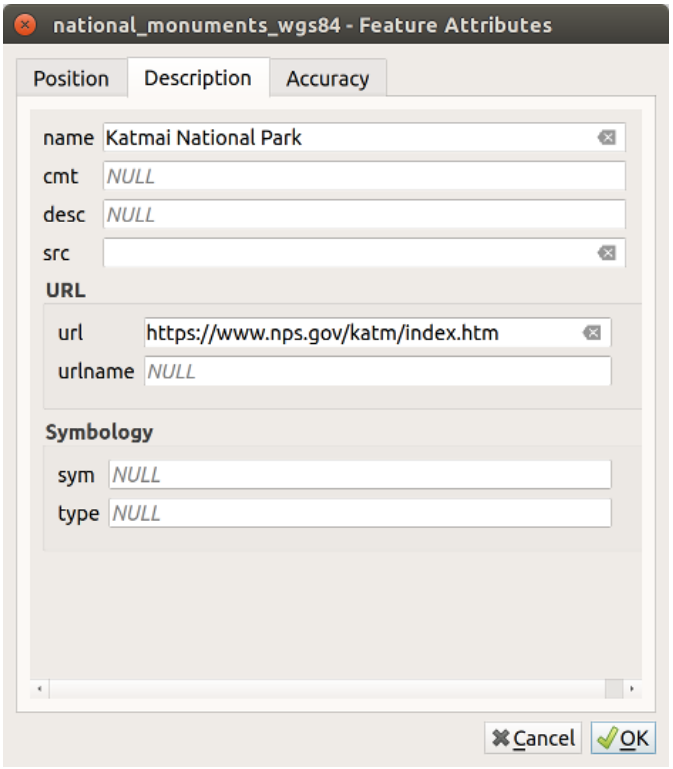


Figura 12.44: Formulario integrado resultante con pestañas y grupos con nombre

1. Elija `Arrastrar y soltar diseñador` del cuadro combinado *Seleccionar editor de diseño de atributos*. Esto habilita el panel *Diseño de formulario* al lado del panel *Controles disponibles*, lleno de campos existentes. El campo seleccionado muestra sus *propiedades* en un tercer panel.

2. Seleccione los campos que no desee utilizar en su panel :guilabel:Diseño del formulario y pulse el botón para eliminarlos. También puede alternar la selección con el botón Invertir selección.

3. Arrastre y suelte los campos del primer panel al de *Diseño de formulario* para volver a añadirlos. El mismo campo puede añadirse varias veces.

4. Arrastre y suelte campos dentro del panel *Form Layout* para reordenar sus posiciones.

5. Añada contenedores para asociar campos que pertenecen a la misma categoría y estructurar mejor el formulario.

    1. El primer paso es utilizar el icono Añadir una nueva pestaña o grupo al diseño del formulario. Los campos y otros grupos se mostrarán en ella.

    2. A continuación, establezca las propiedades del contenedor, es decir:

        - el *Etiqueta*: el título que se utilizará para el contenedor
        - el *Tipo de contenedor*: puede ser una *Tabla*, *Caja de grupo en contenedor* (una caja de grupo plegable dentro de una pestaña u otro grupo) o una *Fila* (un tipo de contenedor que permite organizar los widgets en una fila horizontal, determinando automáticamente el número de columnas en función del número de widgets),
        - el *Dentro de*: esta función opcional le permite seleccionar un contenedor existente en el que se incrustará el nuevo contenedor (*Caja de grupo en contenedor* o *Fila*).
        - y el *Número de columnas* en las que deben distribuirse los campos incrustados

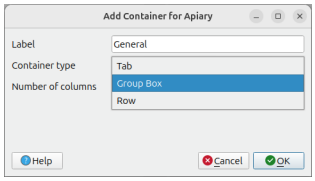


Figura 12.45: Diálogo para crear contenedores con el **Attribute editor layout**

Estas y otras propiedades se pueden actualizar más tarde seleccionando el elemento y, desde el tercer panel:

- oculte o muestre la etiqueta del contenedor
- renombra el contenedor
- ajusta el número de columnas
- ingrese una expresión para controlar la visibilidad del contenedor. La expresión se volverá a evaluar cada vez que los valores en el formulario cambien, y la pestaña o el cuadro de grupo se muestra / oculta en consecuencia
- *Mostrar como cuadro de grupo*: convierte una pestaña en un cuadro de grupo de nivel superior y viceversa.
- en el caso de un cuadro de grupo, establecer si al abrir el formulario, debe mostrarse como *Collapsed* para todos los objetos espaciales, o sólo para aquellas que coincidan con una expresión (*Control colapsado por expresión*).
- configurar el aspecto *Estilo* del contenedor, con propiedades personalizadas *Color de fondo*, color de la etiqueta y fuente.

3. Puede crear e incrustar tantos contenedores como desee pulsando de nuevo el icono Añadir una nueva pestaña o grupo al diseño del formulario.

6. El siguiente paso consiste en asignar los campos pertinentes a cada contenedor, simplemente arrastrando y soltando. Los cuadros de grupo y las pestañas también pueden moverse del mismo modo.
7. *Personalizar el control* de los campos a usar
8. En caso de que la capa esté involucrada en una relación *one or many to many relation*, arrastre y suelte el nombre de la relación desde el panel *Available Widgets* al panel *Form Layout*. El formulario de atributo de capa asociado se incrustará en el lugar elegido en el formulario de la capa actual. En cuanto a los demás elementos, seleccione la etiqueta de relación para configurar algunas propiedades:
   - oculta o muestra la etiqueta relación
   - muestra el botón de enlace
   - mestra el botón desenlazar
9. En caso de que la capa tenga una o más *actions* habilitadas para el alcance de `Layer` o `Feature`, las acciones se enumerarán en *Actions* y puede arrastrarlas y soltarlas como con los otros campos. La acción asociada se incrustará en el lugar elegido en el formulario de la capa actual.
10. Personalice aún más el formulario añadiendo uno o más widgets de *Otros Widgets* (véase *Otros Widgets*)
11. Aplica el diálogo propiedades de capa
12. Abra un formulario de atributo de característica (por ejemplo, usando la herramienta Identify features) y debería mostrar el nuevo formulario.

Otros Widgets

El diseñador de arrastrar y soltar ofrece una serie de widgets que no están conectados a un campo concreto. Pueden utilizarse para mejorar el aspecto del formulario o para mostrar valores calculados dinámicamente.

- *Widget HTML*: incrusta una página HTML, la fuente HTML puede contener el resultado de expresiones calculadas dinámicamente.

  Los widgets HTML pueden utilizarse, por ejemplo, para mostrar imágenes almacenadas como BLOB en un campo (llamémoslo `foto`):

  1. En el modo `Diseñador arrastrar y soltar`, añada un *Widget HTML* a su *Diseño de formulario*`.
  2. Haga doble clic en *HTML Widget* para configurarlo.
  3. Cambia el *Título* predeterminado u ocúltelo.
  4. Pulse el botón **|expresión|** e introduzca la siguiente expresión QGIS:

     ```
     '<img src= "' || to_base64("photo") || '">'
     ```

     Asegúrese de sustituir *foto* por su propio nombre de campo BLOB. La expresión anterior crea una cadena con etiqueta de imagen HTML en la que se codifica el archivo BLOB.

  5. Aplique el diálogo y pulse el botón  .
  6. QGIS aplica automáticamente el formato HTML y funciones para evaluar su expresión, dando como resultado el siguiente código:

     ```
     <script>document.write(expression.evaluate("'<img src=\"data:image/png;
     ↪base64,' || to_base64(\"photo\") || '\">'"));</script>
     ```

     A la derecha aparece una vista previa de la imagen.

- *Widget QML*: incrusta un documento QML de Qt, mostrando elementos gráficos en el formulario de atributos. Además del *Title* personalizado del widget añadido y de si debe mostrarse o no, puede seleccionar entre elementos *Código QML* predefinidos:
  - *Texto libre…*: permite escribir desde cero o pegar un código existente
  - *Rectángulo*: proporciona un código mínimo para mostrar un rectángulo
  - *Gráfico de barras*: proporciona un código mínimo para mostrar un gráfico de barras
  - *Gráfico circular*: proporciona un código mínimo para mostrar un gráfico circular

  Puede ampliar el código con sintaxis QML, utilizar campos de capa o expresiones QGIS que se calculan dinámicamente.

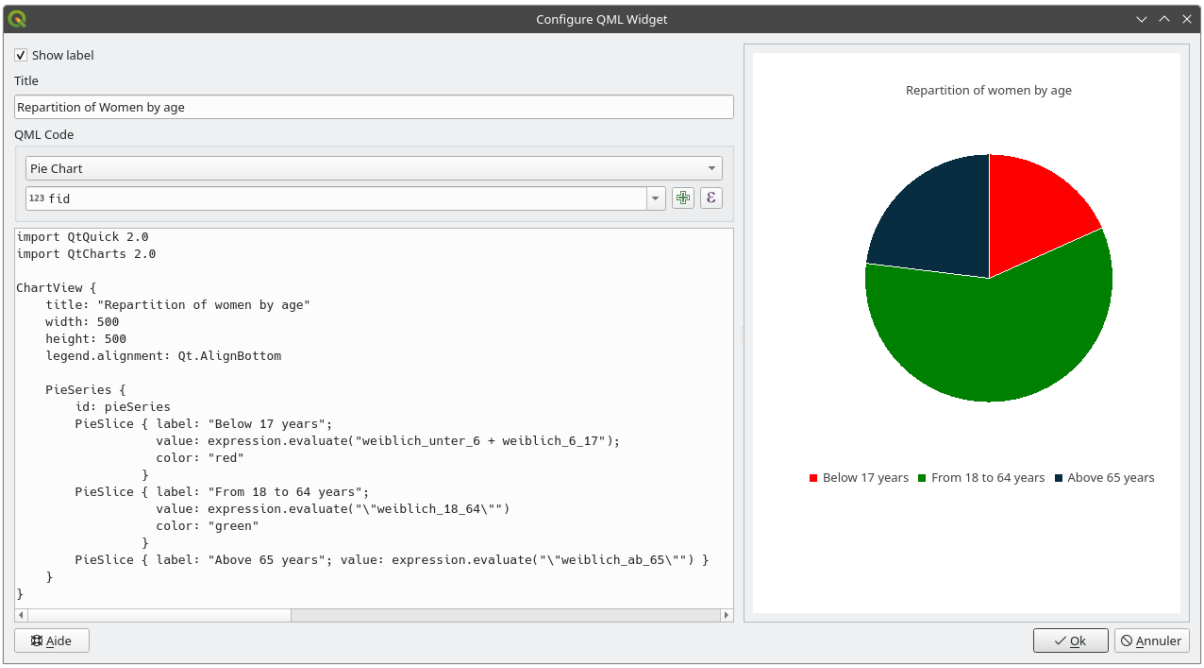


Figura 12.46: Configurar un gráfico QML para que se muestre en forma de atributo

- *Widget de texto*: muestra un widget de texto que admite marcado HTML básico y puede contener el resultado de expresiones calculadas dinámicamente.
- *Widget espaciador*: inserta un rectángulo transparente vacío, aumentando la distancia vertical entre dos widgets.

Truco

Mostrar contenido dinámico

Los widgets mencionados anteriormente (excepto el *Spacer Widget*) admiten expresiones que pueden utilizarse para mostrar contenido que cambia dinámicamente cada vez que cambia otro campo del formulario. Esto puede conseguirse utilizando la función `current_value('nombre_campo')` en la expresión para examinar el valor de otro campo.

Usando ui-archivo personalizado

La opción `Provide ui-file` le permite usar diálogos complejos hechos con Qt-Designer. El uso de un archivo UI permite una gran libertad al crear un diálogo. Tenga en cuenta que, para vincular los objetos gráficos (cuadro de texto, cuadro combinado …) a los campos de la capa, debe darles el mismo nombre.

Use la *Edit UI* para definir la ruta al archivo a utilizar.

Los archivos de IU también se pueden alojar en un servidor remoto. En este caso, proporciona la URL del formulario en lugar de la ruta del archivo en *Edit UI*.

Encontrarás algún ejemplo en la lección Crear un nuevo formulario del QGIS-training-manual-index-reference. Para obtener información más avanzada, consulte https://woostuff.wordpress.com/2011/09/05/qgis-tips-custom-feature-forms-with-python-logic/.

Mejora tu formulario con funciones personalizadas

Los formularios QGIS pueden tener una función Python que se llama cuando se abre el cuadro de diálogo. Use esta función para agregar lógica adicional a sus cuadros de diálogo. El código del formulario se puede especificar de tres maneras diferentes:

- `load from the environment`: usar una función, por ejemplo en `startup.py` o desde un complemento instalado
- `load from an external file`: un selector de archivos le permitirá seleccionar un archivo Python de su sistema de archivos o ingresar una URL para un archivo remoto.
- `provide code in this dialog`: aparecerá un editor de Python donde puede escribir directamente la función a utilizar.

En todos los casos, debe ingresar el nombre de la función que se llamará (`open` en el ejemplo siguiente).

Un ejemplo es (en el módulo MyForms.py):

```
def open(dialog,layer,feature):
    geom = feature.geometry()
    control = dialog.findChild(QWidget,"My line edit")
```

Referencia en Python Init Function como: `open`

Configurar el comportamiento del campo

La parte principal de la pestaña *Attributes Form* le ayuda a establecer el tipo de widget utilizado para completar o mostrar los valores del campo, en la tabla de atributos o en el formulario de características: puede definir cómo interactúa el usuario con cada campo y los valores o rango de valores que se pueden agregar a cada uno .

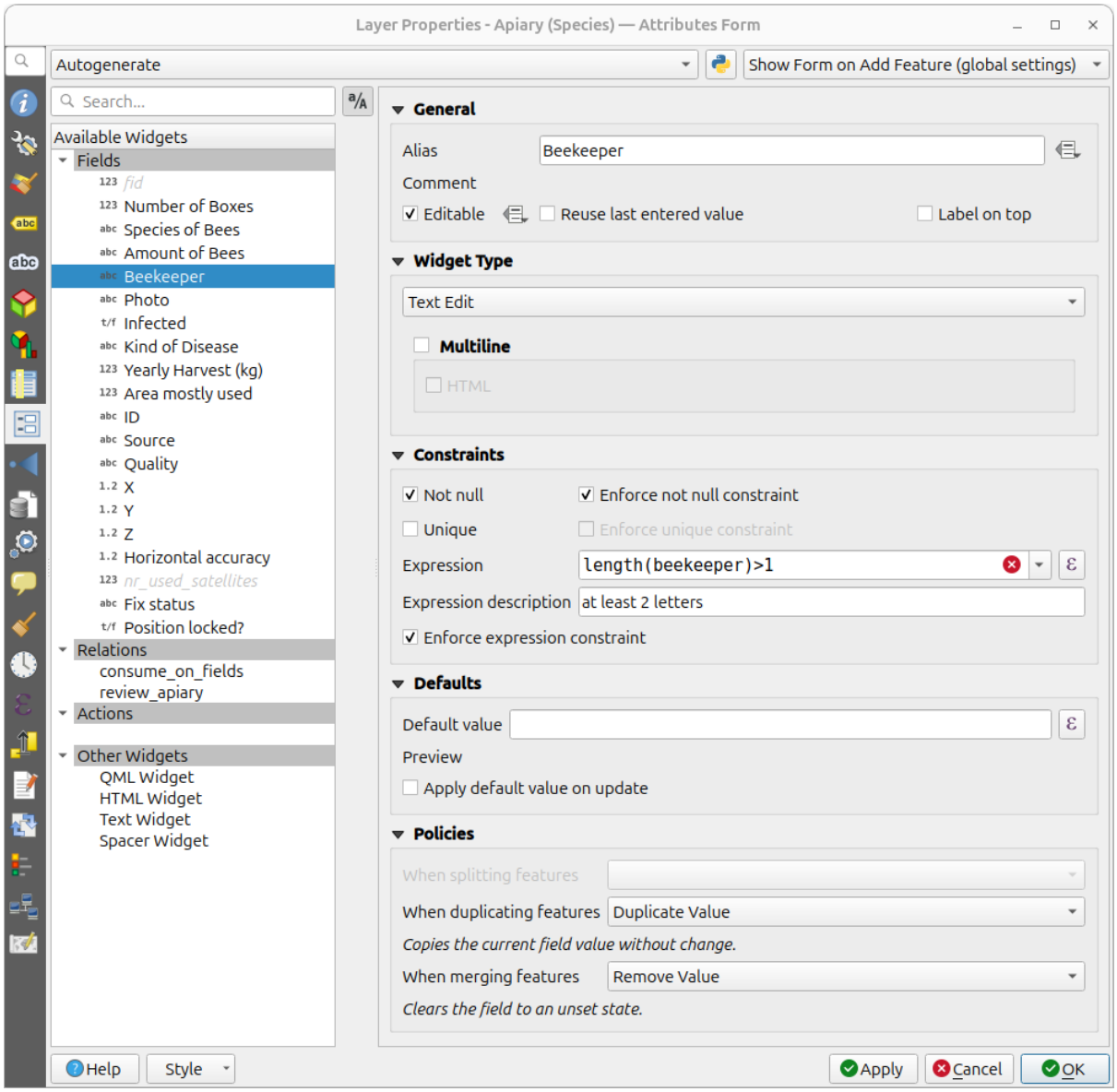


Figura 12.47: Cuadro de diálogo para seleccionar un widget de edición para una columna de atributos

Truco

Copy and paste widget configuration

The field configuration can be copied to the clipboard and then pasted to another field via right click on a field in the *Available widgets* panel.

This comes handy to save time and to make sure that common settings are correctly applied to several related fields, e.g., when configuring latitude and longitude field widgets.

Copy and paste of widget configuration can be done between fields of the same layer, between fields from different layers in the QGIS project, or between fields from layers in different QGIS instances.

Note that neither aliases nor comments are copied, since those are specific to each particular field.

Configuración común

Independientemente del tipo de widget aplicado al campo, hay algunas propiedades comunes que puede configurar para controlar si un campo puede editarse y cómo.

Widget visualizar

Sólo disponible para el modo diseñador *Arrastrar y soltar*, este grupo le ayuda a configurar el aspecto del widget asignado al campo:

- *Mostrar etiqueta*: indica si el nombre del campo debe mostrarse en el formulario
- *Sustituir el color de la etiqueta*: aplica un color específico a la etiqueta del campo
- *Sustituir fuente de etiqueta*: aplica propiedades de fuente específicas (negrita, cursiva, subrayado, tachado o familia de fuentes) a la etiqueta del campo.
- *Tamaño*: permite controlar cómo se redimensionarán relativamente los widgets al redimensionar un formulario de atributos.
  - *Estiramiento horizontal*: establece un valor horizontal más alto para los widgets que necesitan más espacio horizontal.
  - *Estiramiento vertical*: determina cómo se redimensionan verticalmente los widgets cuando se cambia el tamaño del formulario.

Opciones generales

- *Alias*: un nombre legible por humanos para usar en campos. El alias se mostrará en el formulario de características, la tabla de atributos o en el panel *Identify results*. También se puede usar como reemplazo de nombre de campo en *expression builder*, facilitando expresiones, comprensión y comentarios. Los alias se guardan en el archivo del proyecto.
- *Comment*: muestra el comentario del campo como se muestra en la pestaña *Fields*, en un estado de solo lectura. Esta información se muestra como información sobre herramientas al pasar el cursor sobre la etiqueta del campo en un formulario de características.
- *Editable*: desmarque esta opción para establecer el campo como de sólo lectura (no modificable manualmente) incluso cuando la capa esté en modo edición. Tenga en cuenta que marcar esta opción no anula ninguna limitación de edición del proveedor. Esta opción puede ser controlada por una propiedad definida por datos gracias al botón Anulación definida por datos.
- *Reuse last entered value*: recuerda el último valor ingresado en este campo y lo usa como predeterminado para la próxima entidad que se edite en la capa.
- *Label on top*: coloca el nombre del campo encima o al lado del widget en el formulario de características.

Restricciones

Puede restringir el valor para insertar en el campo. Esta restricción puede ser:

- *Not null*: Requiere que el usuario proporcione un valor;
- *Unique*: garantizar que el valor insertado sea único en todo el campo;
- basada en una *expresión* personalizada: p.ej. `not regexp_match(col0,'[^A-Za-z]')` se asegurará de que el valor del campo *col0* tenga solo letras del alfabeto. Se puede agregar una breve descripción para ayudarlo a recordar la restricción.

Cada vez que se agrega o edita un valor en un campo, se envía a las restricciones existentes y:

- si cumple con todos los requisitos, se muestra un cheque verde al lado del campo en el formulario;
- si no cumple con todos los requisitos, el campo se colorea de amarillo o naranja y se muestra la cruz correspondiente junto al widget. Puede pasar el cursor sobre la cruz para recordar qué restricciones se aplican al campo y corregir el valor:
  - Aparece una cruz amarilla cuando la restricción no cumplida es una restricción no forzada (`soft constraint`) y no impide guardar los cambios con los valores «erróneos»;
  - Una cruz naranja no puede ignorarse y no le permite guardar sus modificaciones hasta que cumplan las restricciones. Aparece cuando la opción  *Aplicar la restricción* está marcada (`hard constraint`).

Valores predeterminados

- *Default value*: para las nuevas funciones, rellena automáticamente de forma predeterminada el campo con un valor predefinido o un *expression-based one*. Por ejemplo, puede:
  - use `$x`, `$length`, `$area` rellenar automáticamente un campo con la coordenada X de la entidad, la longitud, el área o cualquier información geométrica en su creación;
  - incremente un campo en 1 para cada nueva característica usando `maximum("field")+1`;
  - guardar la fecha y hora de creación de la función usando `now()`;
  - use en expresiones *variables*, haciendo más fácil, p. inserte el nombre del operador (`@user_full_name`), la ruta del archivo del proyecto (`@ project_path`), …

  Una vista previa del valor predeterminado resultante se muestra en la parte inferior del widget.

  Nota

  La opción `` Valor predeterminado “” no conoce los valores en ningún otro campo de la función que se está creando, por lo que no será posible usar una expresión que combine ninguno de esos valores, es decir, usar una expresión como ``concat(field1,field2) `` puede no funcionar.

-  *Apply default value on update*: cada vez que se cambia el atributo de característica o la geometría, se recalcula el valor predeterminado. Esto podría ser útil para guardar valores como el último usuario que modifica los datos, la última vez que se cambió …

Políticas

Field policies determine how values are assigned to fields during various editing operations:

Split and Duplicate Policies

These policies apply when *Splitting features* or *Duplicating features*:

- *Duplicar valores*: Mantiene el valor existente del campo para las nuevas características.
- *Usar valor predeterminado*: Restablece el campo recalculando su *valor predeterminado*. Si no existe ninguna cláusula de valor predeterminado, se mantiene el valor existente para los nuevos objetos espaciales.
- *Eliminar valor*: Borra el campo a un estado no establecido.
- *Utilizar geometrías de relación*: Recalcula el valor del campo para todas las partes divididas multiplicando el valor existente por la relación de las longitudes o áreas de las partes divididas.

Merge Policies

These policies determine initial values when *Merging features*:

- *Remove Value*: Clears the field to an unset state (data provider may populate default value).
- *Use Default Value*: Uses the default field value set in QGIS. (expressions use a dummy feature with the merged geometry; references to other fields won't work).
- *Use Sum* (numeric fields only): Sums values from all merged features.
- *Use Minimum Value* (numeric fields only): Uses smallest value from the selected features.
- *Use Maximum Value* (numeric fields only): Uses largest value from the selected features.
- *Average Weighted by Geometry* (numeric fields only): New values are computed as the weighted average of the source values.
- *Use Largest Feature*: Uses value from feature with the largest geometry.
- *Set to NULL*: Sets attribute to NULL.

Modo de edición

Según el tipo de campo, QGIS determina y le asigna automáticamente un tipo de widget predeterminado. Luego puede reemplazar el widget con cualquier otro compatible con el tipo de campo. Los widgets disponibles son:

- **Binario (BLOB)**: Disponible sólo para campos binarios, muestra por omisión una etiqueta con el tamaño de los datos incrustados, si no está vacía. Un botón desplegable junto a la etiqueta permite:
  - *Incrustar archivo*, sustituyendo o rellenando el campo
  - *Borrar contenido*, eliminando cualquier dato del campo
  - *Guardar el contenido en un fichero*, exportando los datos como un fichero en disco

  También es posible previsualizar el archivo binario incrustado en el campo, si se combina de forma arrastrada con, por ejemplo, un *widget QML o HTML*.
- **Checkbox**: Displays a checkbox whose state defines the value to insert. Internally, the widget stores a value (e.g. 1 for checked, 0 for unchecked) while representing it visually as a toggled box. You can configure the stored values for the checked and unchecked states in the representation. It is also possible to display the checkbox state as a plain text using «True»/»False» labels. These settings influence how the values are shown in classification renderers.
- **Classification**: Solo disponible cuando una *categorized symbology* se aplica a la capa, muestra un cuadro combinado con los valores de las clases.
- **Color**: Muestra un *color widget* permitiendo seleccionar un color; El valor de color se almacena como una notación html en la tabla de atributos.
- **Date/Time**: Muestra un campo de línea que puede abrir un widget de calendario para ingresar una fecha, una hora o ambas. El tipo de columna debe ser texto. Puede seleccionar un formato personalizado, abrir un calendario, etc.
- **Enumeration**: Abre un cuadro combinado con valores predefinidos extraídos de la base de datos. Actualmente, esto solo es compatible con el proveedor de PostgreSQL, para campos de tipo``enum``.
- **Adjunto**: Utiliza un cuadro de diálogo «Abrir archivo» para almacenar la ruta del archivo en modo relativo o absoluto. Puede utilizarse para mostrar la ruta del documento como un hipervínculo o para mostrar el documento en un widget específico del formulario. Los tipos de documento admitidos son imagen, página web, audio y vídeo, y los formatos de archivo admitidos dependen del sistema operativo. También puede configurar un *sistema de almacenamiento externo* para buscar/almacenar recursos.

  Truco

  Relative Path in Attachment widget

Si la ruta que se selecciona con el explorador de archivos se encuentra en el mismo directorio que el archivo de proyecto o inferior `.qgs`, las rutas se convierten en rutas relativas. Esto aumenta la portabilidad de un proyecto `.qgs` con información multimedia adjunta.

- **Geometry**: Compatible with the *Geometry* field type. For example, PostgreSQL and Memory layers can contain multiple geometry fields. QGIS uses the first geometry column as the layer geometry. Applying the Geometry widget to additional geometry fields allows lossless use of the data in the attribute form:
  - Copy in the formats WKT (Well-known text representation, e.g. `POINT (30 10)`) or GeoJSON
  - Paste
  - Clear
- **Hidden**: Una columna de atributos ocultos es invisible. El usuario no puede ver su contenido.
- **Key/Value**: Muestra una tabla de dos columnas para almacenar conjuntos de pares clave / valor dentro de un solo campo. Actualmente es compatible con el proveedor de PostgreSQL, para campos de tipo `hstore`.
- **JSON View**: Muestra datos JSON en una edición de texto resaltada por sintaxis o en una vista de árbol. Este widget actualmente es de solo lectura. Hay varias opciones disponibles para cambiar la forma en que se muestran los datos. “Default view” especifica si el widget debe aparecer en modo Texto o Árbol. “Format JSON” tiene tres opciones que están relacionadas solo con la vista de árbol:
  - Con sangría: muestra los datos en un formato legible por humanos con líneas nuevas y cuatro caracteres de espacio para la sangría.
  - Compacto: muestra los datos en una cadena optimizada de tamaño de una línea sin líneas nuevas ni espacios.
  - Deshabilitado: muestra los datos tal como provienen del proveedor.
- **List**: Muestra una tabla de una sola columna para agregar diferentes valores dentro de un solo campo. Actualmente es compatible con el proveedor de PostgreSQL, para campos de tipo `array`.
- **Range**: Le permite establecer valores numéricos de un rango específico. El widget de edición puede ser un control deslizante o un cuadro giratorio.

> **Nota**
>
> Algunas capas, como GeoPackage o ESRI File Geodatabase, con **dominios de campo de rango** predefinidos serán reconocidas automáticamente por QGIS y se les asignará un widget **Rango** para los campos relevantes. El widget se rellenará previamente con los valores mínimo y máximo especificados en el dominio.

- **Relation Reference**: This is the default widget assigned to the referencing field (i.e., the foreign key in the child layer) when a *relation* is set. It provides direct access to the parent feature’s form which in turn embeds the list and form of its children. The number of entries in the widget can be limited for efficiency, and if limit is not set, all entries will be loaded. The field defined in the display expression is used as representable value in classification renderers.
- **Text Edit** (default): Esto abre un campo de edición de texto que permite utilizar texto simple o varias líneas. Si elige varias líneas, también puede elegir contenido html.
- **Unique Values**: Puede seleccionar uno de los valores ya utilizados en la tabla de atributos. Si se activa “Editable”, se muestra una edición de línea con soporte de autocompletado; de lo contrario, se utiliza un cuadro combinado.
- **Uuid Generator**: Genera un campo UUID (identificadores únicos universal) de solo lectura, si está vacío.
- **Value Map**: A combo box with predefined items. The value is stored in the attribute, the description is shown in the combo box. You can define values manually or load them from a layer or a CSV file. The description is used as the representable value, meaning it will be shown in classification renderers.

> **Nota**
>
> Algunas capas, como GeoPackage o ESRI File Geodatabase, con **dominios de campo codificados** predefinidos serán reconocidos automáticamente por QGIS y se les asignará un widget **Mapa de Valores** para los campos relevantes.

- **Value Relation**: Offers values from a related table in a combobox. You can select layer, key column and value column. The key column is the stored value in the attribute table, while the value column is the label shown in the combobox. Value column is also the representable value that will be shown in the classification renderers. You may also choose to order the list of displayed values by the *Key* column (the default sort order), the *Value* column, or a specific *Field* of the layer. Enable the *Descending order* if you wish the values to be sorted in reverse order.

  Several options are available to change the standard behavior: allow NULL value, allow multiple selections and use of auto-completer. The forms will display either a drop-down list or a line edit field when completer checkbox is enabled.

  Si una capa que está almacenada en PostgreSQL, GeoPackage o SpatiaLite está configurada para usar un widget de relación de valor, pero la capa requerida aún no está cargada en el proyecto, QGIS buscará automáticamente la capa en la misma base de datos/conexión.

12.1.10 Propiedades de unión

La pestaña *Uniones* permite asociar características de la capa actual a características de otra capa vectorial cargada (o tabla). La unión se basa en un atributo compartido por las capas, en una relación de uno a uno. Para más detalles sobre las uniones, lea *Unir objetos espaciales entre dos capas*.

12.1.11 Propiedades de almacenamiento auxiliar

La forma habitual de personalizar el estilo y el etiquetado es usar propiedades definidas por datos como se describe en *Configuración de anulación definida por datos*. Sin embargo, puede que no sea posible si los datos subyacentes son de solo lectura. ¡Además, la configuración de estas propiedades definidas por datos puede llevar mucho tiempo o no ser deseable! Por ejemplo, si desea utilizar completamente las herramientas de mapas que vienen con *La Barra de Herramientas Etiqueta*, entonces necesita agregar y configurar más de 20 campos en su fuente de datos original (posiciones X e Y, ángulo de rotación, estilo de fuente, color, etc.).

El mecanismo de almacenamiento auxiliar proporciona la solución a estas limitaciones y configuraciones incómodas. Los campos auxiliares son una forma indirecta de administrar y almacenar automáticamente estas propiedades definidas por datos (etiquetas, diagrama, simbología …) en una base de datos SQLite gracias a las combinaciones editables. Esto le permite almacenar propiedades para capas que no son editables.

Hay una pestaña disponible en el cuadro de diálogo de propiedades de capa vectorial para administrar el almacenamiento auxiliar:

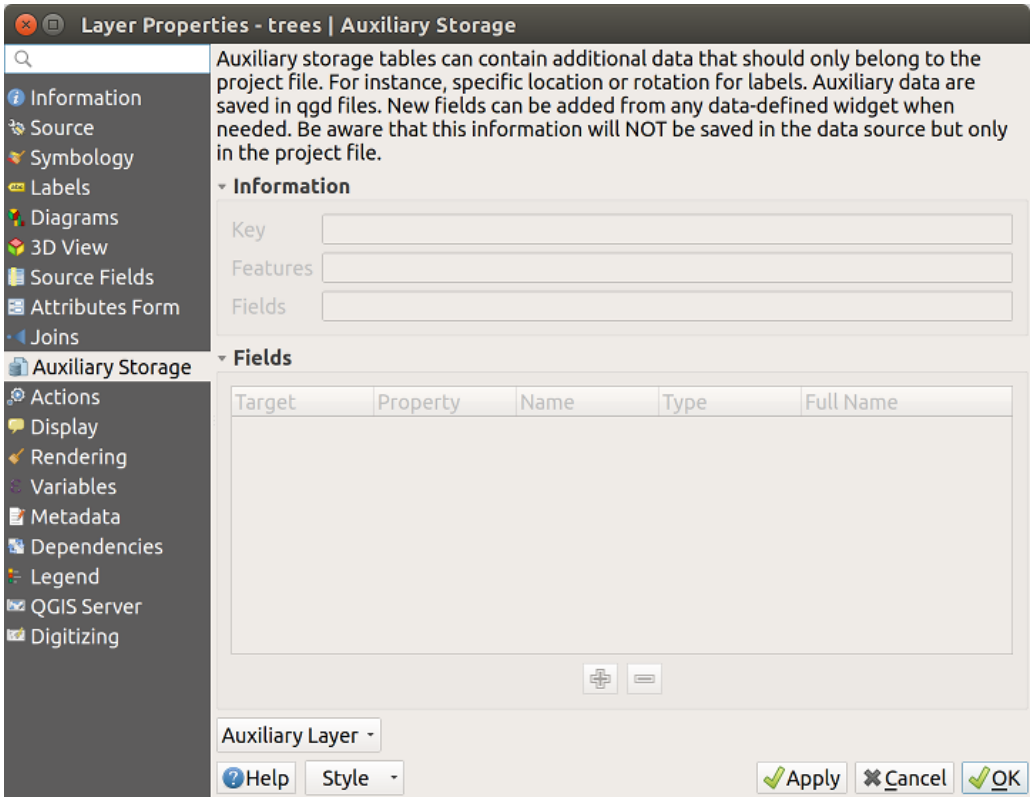


Figura 12.48: Pestaña de almacenamiento auxiliar

Etiquetando

Teniendo en cuenta que la fuente de datos puede personalizarse gracias a las propiedades definidas por los datos sin ser editables, las herramientas de mapa de etiquetado descritas en *La Barra de Herramientas Etiqueta* siempre están disponibles tan pronto como se activa el etiquetado.

En realidad, el sistema de almacenamiento auxiliar necesita una capa auxiliar para almacenar estas propiedades en una base de datos SQLite (ver *Base de datos de almacenamiento auxiliar*). Su proceso de creación se ejecuta la primera vez que hace clic en el mapa mientras se activa una herramienta de mapa de etiquetado. Luego, se muestra una ventana que le permite seleccionar la clave principal que se usará para unirse (para garantizar que las entidades se identifiquen de forma exclusiva):

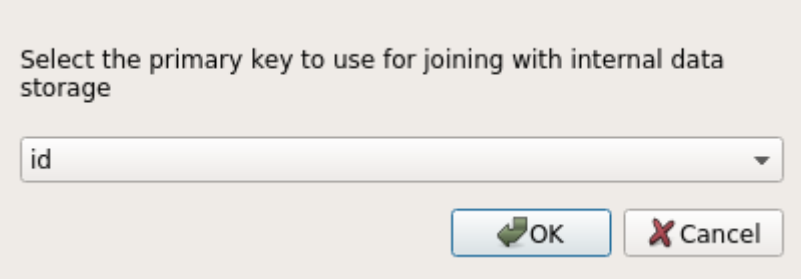


Figura 12.49: Diálogo de creación de capa auxiliar

Tan pronto como se configura una capa auxiliar para la fuente de datos actual, puede recuperar su información en la pestaña:

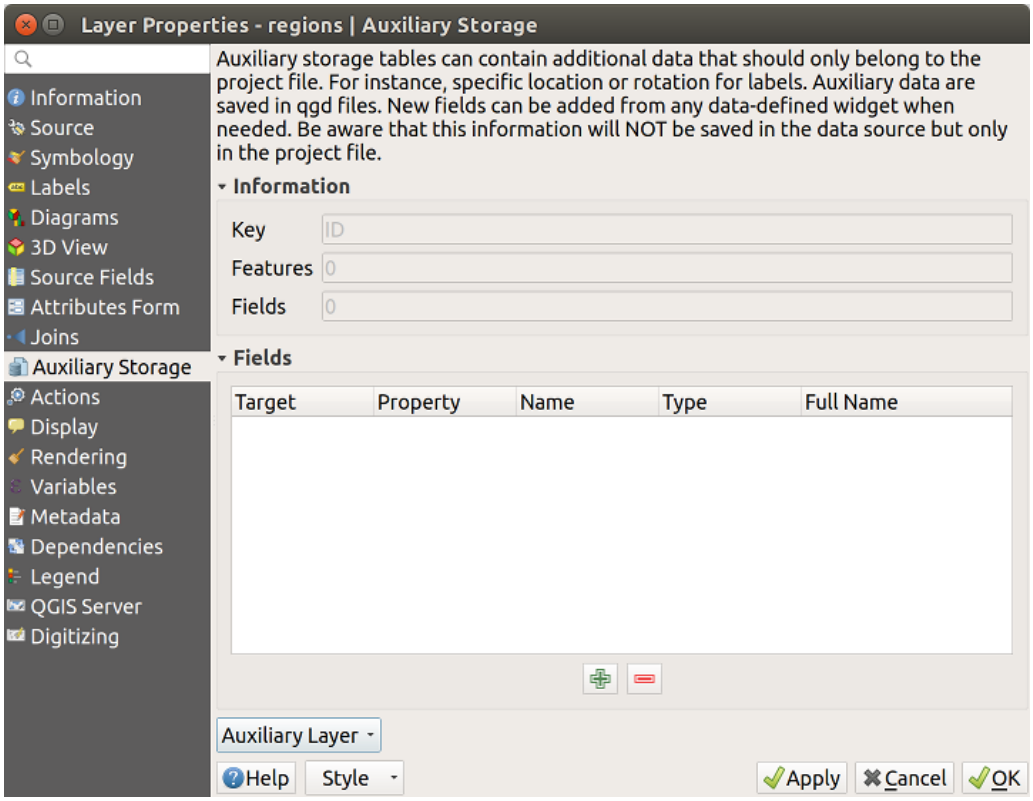


Figura 12.50: Clave de capa auxiliar

La capa auxiliar ahora tiene estas características:

- la clave principal es `ID`,
- hay `0` entidades que usan un campo auxiliar,
- hay `0` campos auxiliares.

Ahora que se creó la capa auxiliar, puede editar las etiquetas de capa. Haga clic en una etiqueta mientras la herramienta de mapa Change Label está activada, entonces puede actualizar las propiedades de estilo como tamaños, colores, etc. Se crean las propiedades definidas por datos correspondientes y se pueden recuperar:

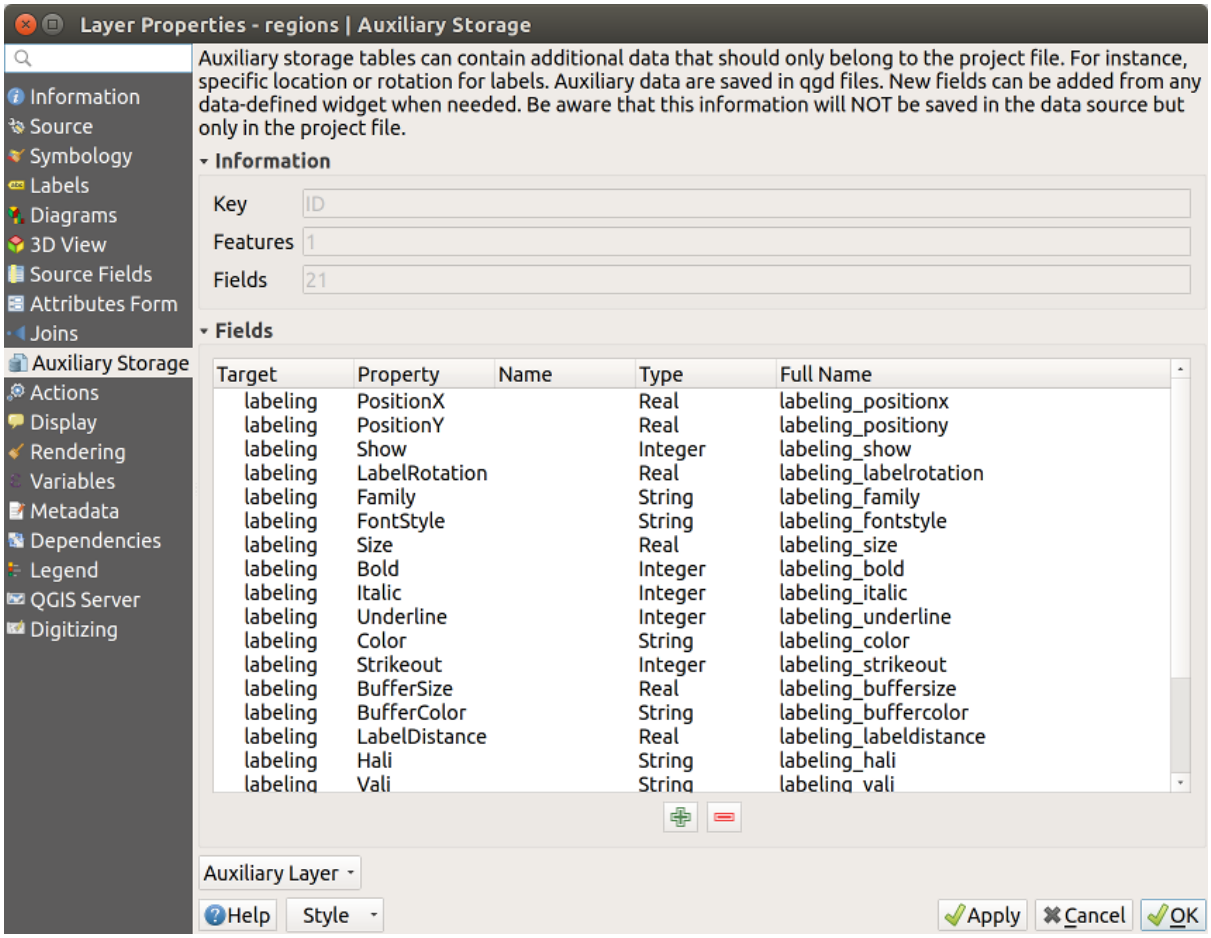


Figura 12.51: Campos Auxiliares

Como puedes ver en la figura de arriba, `21` los campos se crean y configuran automáticamente para el etiquetado. Por ejemplo, el campo auxiliar `FontStyle` es del tipos `String` y es llamado `labeling_fontstyle``en la base de datos SQLite subyacente. También hay una función ``1` que actualmente utiliza estos campos auxiliares.

Note que el icono  se muestra en la pestaña propiedades *Labels* indicando que las opciones de anulación definidas por datos están configuradas correctamente:

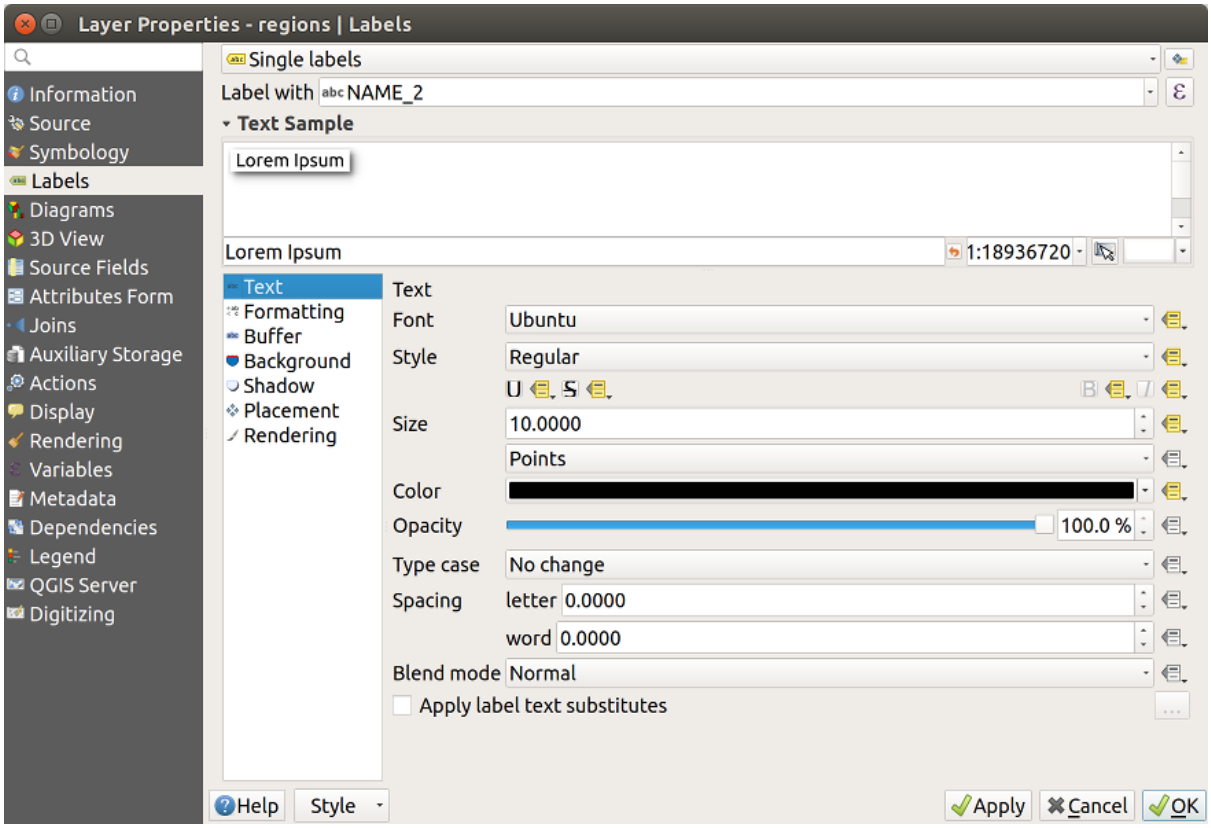


Figura 12.52: Propiedades definidas por datos creadas automáticamente

Para obtener información más avanzada, consulte De lo contrario, hay otra forma de crear un campo auxiliar para una propiedad específica gracias al botón anulación definida por datos. Al hacer clic en *Almacenar datos en el proyecto*, se crea automáticamente un campo auxiliar para el campo *Opacidad*. Si hace clic en este botón y la capa auxiliar aún no se ha creado, primero se muestra una ventana (Figura 12.49) para seleccionar la clave principal que se utilizará para la unión.

Simbología

Al igual que el método descrito anteriormente para personalizar etiquetas, los campos auxiliares también se pueden usar para estilizar símbolos y diagramas. Para hacer esto, haga clic en Data-defined override y seleccione *Store data in the project* para una propiedad específica. Por ejemplo, el campo *Fill color* :

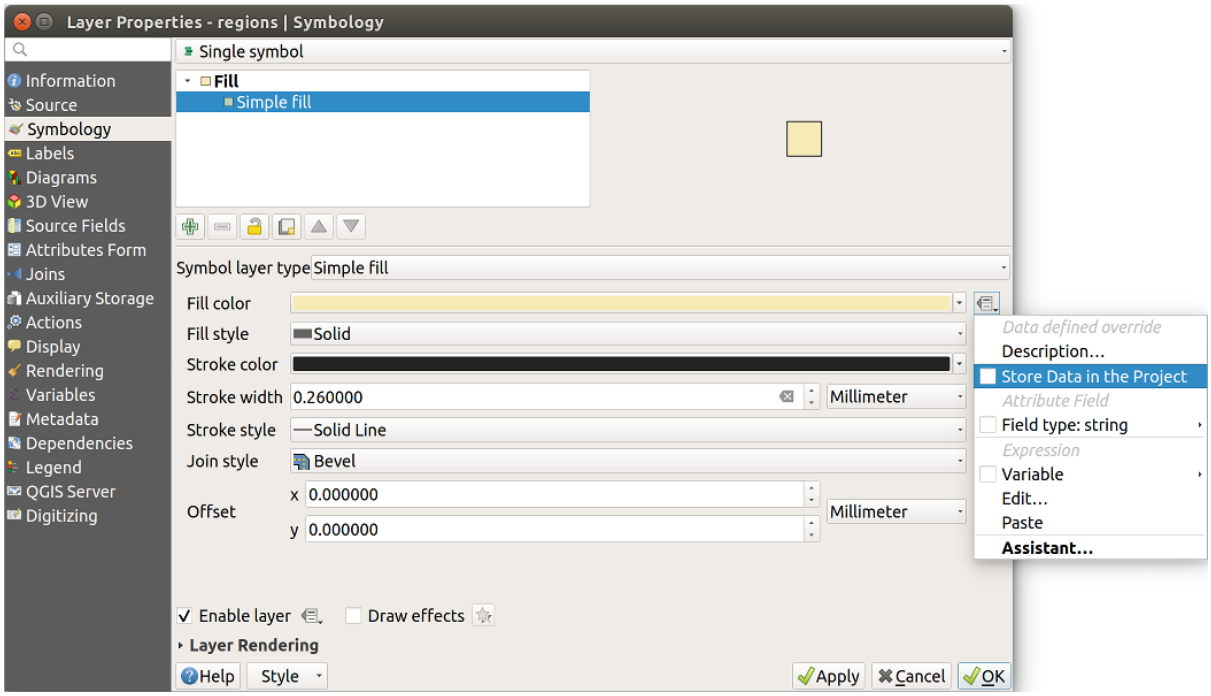


Figura 12.53: Menú de propiedades definido por datos para símbolo

Hay diferentes atributos para cada símbolo (por ejemplo, estilo de relleno, color de relleno, color de trazo, etc.), por lo que cada campo auxiliar que representa un atributo requiere un nombre único para evitar conflictos. Después de seleccionar *Store data in the project*, se abre una ventana y muestra el *Type`del campo y le solicita que ingrese un nombre único para el campo auxiliar. Por ejemplo, al crear un campo auxiliar :guilabel:`Fill color* se abre la siguiente ventana:

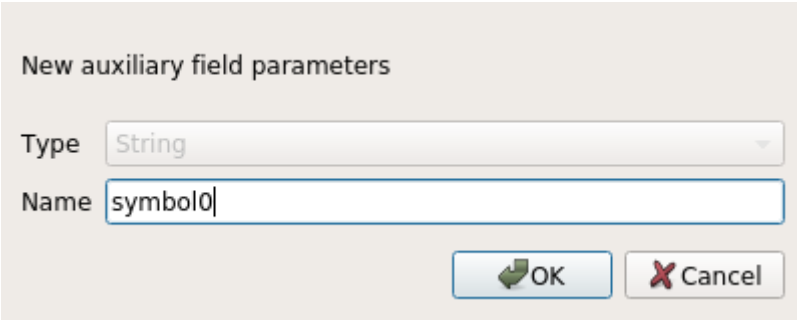


Figura 12.54: Nombre del campo auxiliar para un símbolo

Una vez creado, el campo auxiliar puede ser recuperado en la pestaña de almacenamiento auxiliar:

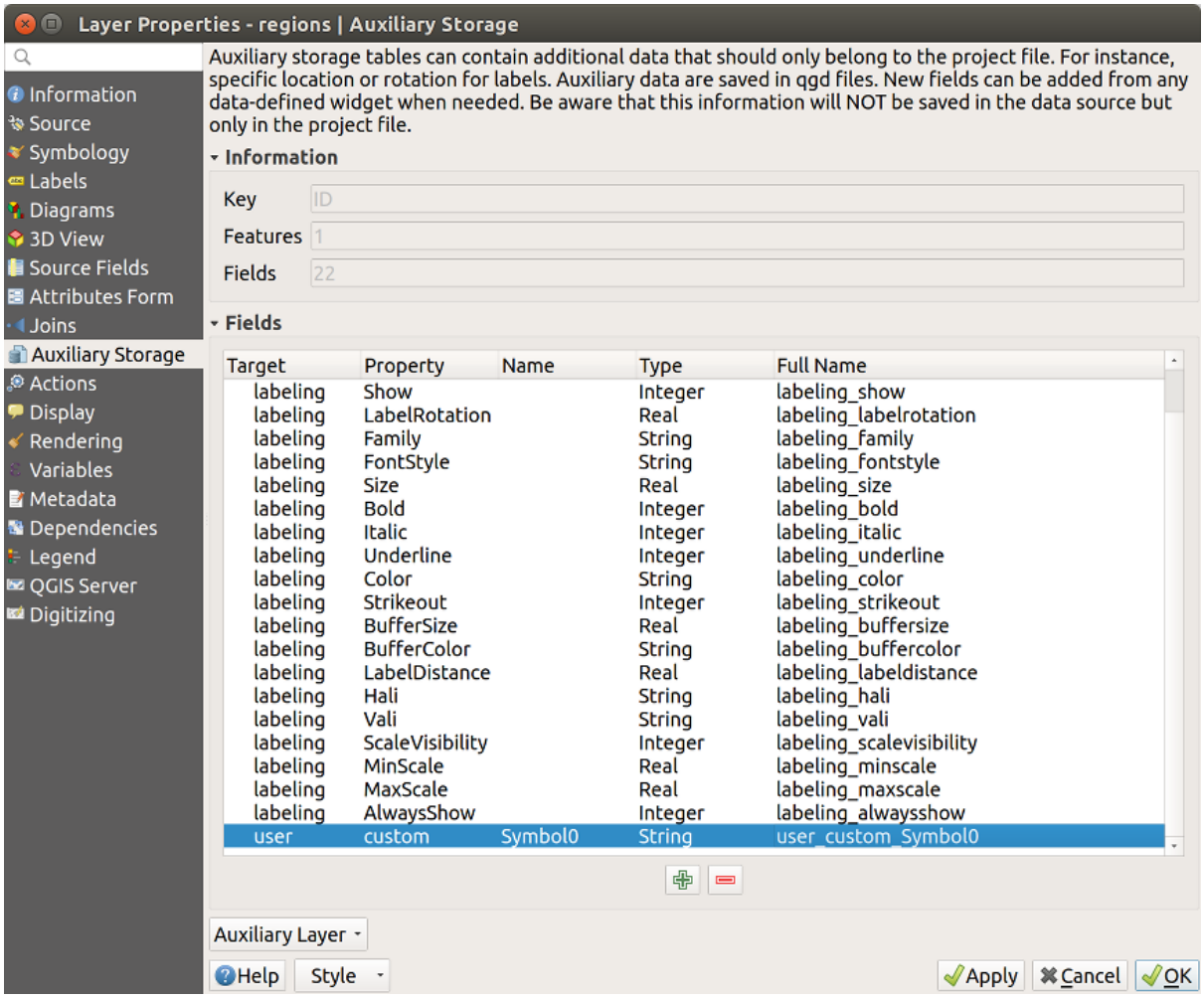


Figura 12.55: Símbolo de campo auxiliar

Tabla de atributos y widgets

Los campos auxiliares se pueden editar usando el *attribute table*. Sin embargo, no todos los campos auxiliares son inicialmente visibles en la tabla de atributos.

Los campos auxiliares que representan los atributos de la simbología, el etiquetado, la apariencia o los diagramas de una capa aparecerán automáticamente en la tabla de atributos. La excepción son los atributos que se pueden modificar utilizando *Label Toolbar* los cuales se ocultan por defecto. Los campos auxiliares representado a un `Color` tienen un widget **Color** preestablecido por defecto, de lo contrario, los campos auxiliares predeterminados para el widget **Text Edit**.

Campos auxiliares que representan atributos que pueden modificarse utilizando la *Label toolbar* son **Hidden** en la tabla de atributos por defecto. Para hacer visible un campo, abra el *Attribute Form properties tab* y cambiar el valor de un campo auxiliar *Widget Type* de **Hidden** a otro valor relevante. Por ejemplo, cambie el **auxiliary_storage_labeling_size** a **Text Edit** o cambie **auxiliary_storage_labeling_color** al wiget **Color** . Esos campos ahora serán visibles en la tabla de atributos.

Los campos auxiliares en la tabla de atributos aparecerán como la siguiente imagen:

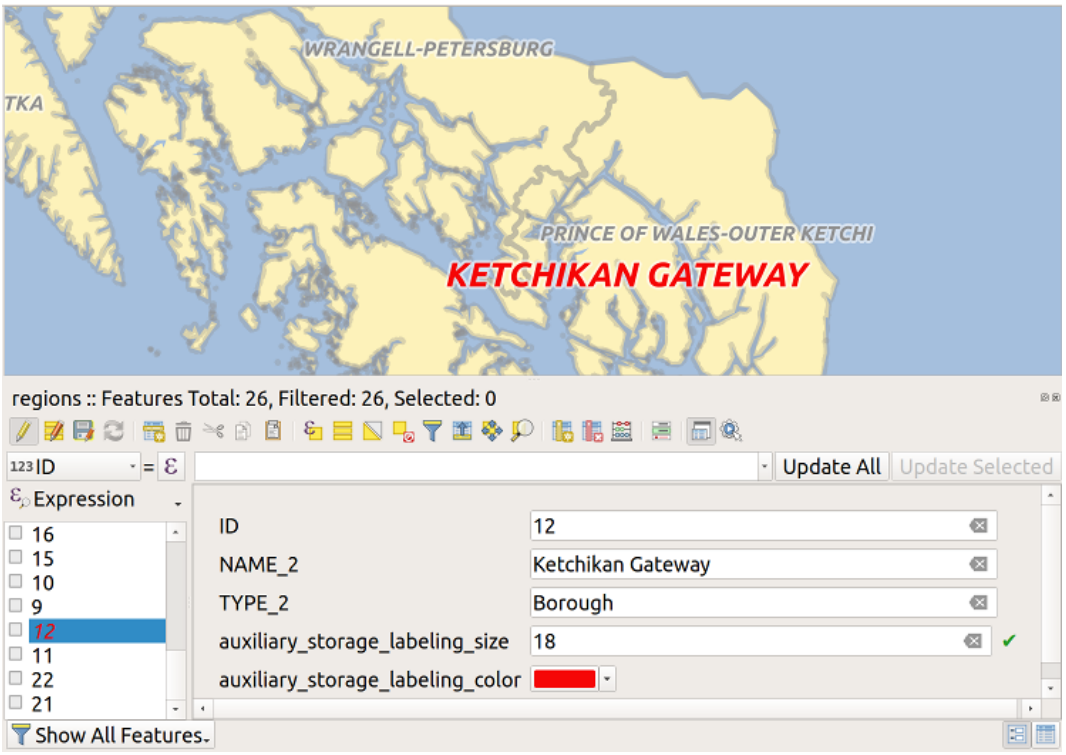


Figura 12.56: Formulario con campos auxiliares

Administración

El menú *Auxiliary Layer* le permite administrar los campos auxiliares:

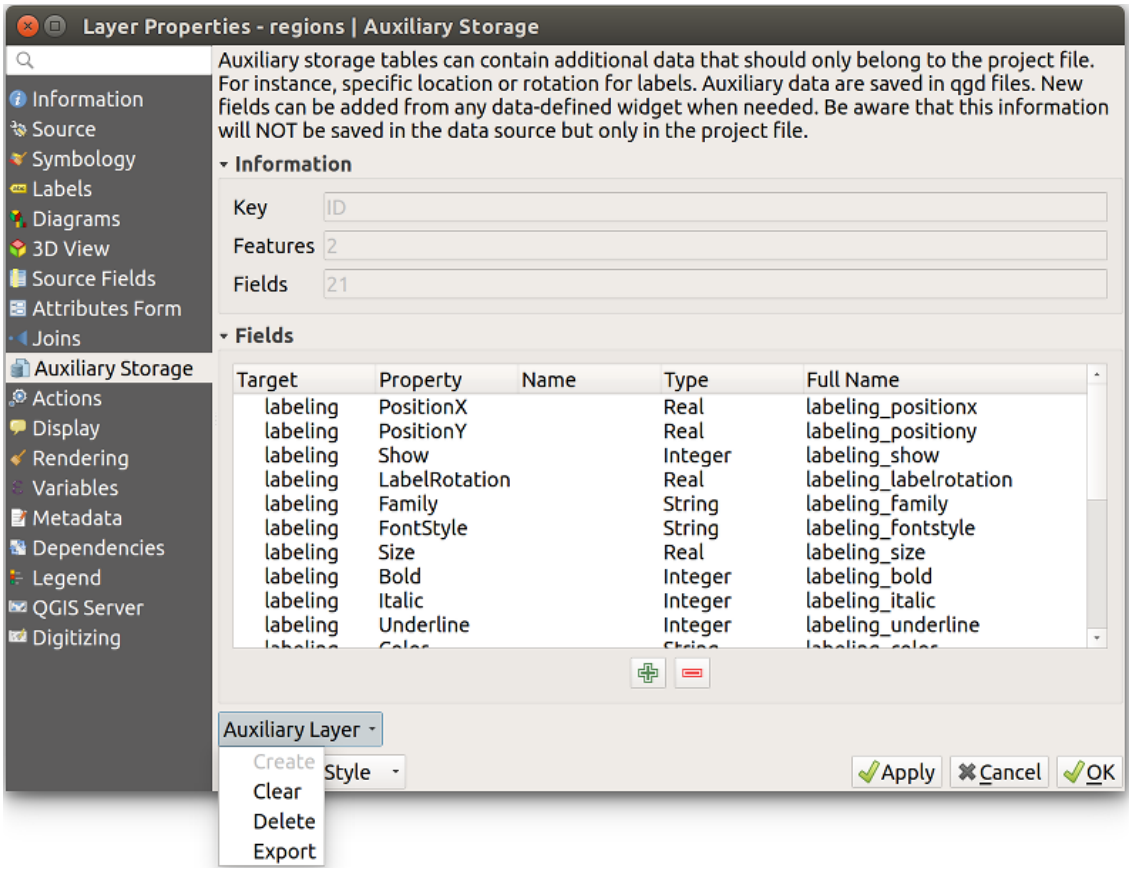


Figura 12.57: Administración de capas auxiliares

El primer elemento *Create* está deshabilitado en este caso porque la capa auxiliar ya está creada. Pero en caso de un nuevo trabajo, puede usar esta acción para crear una capa auxiliar. Como se explica en *Etiquetando*, entonces se necesitará una clave primaria.

La acción *Clear* permite mantener todos los campos auxiliares, pero eliminar sus contenidos. De esta manera, la cantidad de características que usan estos campos caerá a `0`.

La acción *Delete* elimina completamente la capa auxiliar. En otras palabras, la tabla correspondiente se elimina de la base de datos SQLite subyacente y se pierde la personalización de propiedades.

Finalmente, la acción *Export* permite grabar la capa auxiliar como una *new vector layer*. Tenga en cuenta que las geometrías no se almacenan en el almacenamiento auxiliar. Sin embargo, en este caso, las geometrías también se exportan desde la fuente de datos original.

Base de datos de almacenamiento auxiliar

Cuando guarda su proyecto con el formato `.qgs`, la base de datos SQLite utilizada para el almacenamiento auxiliar se guarda en el mismo lugar pero con la extensión `.qgd`.

Para mayor comodidad, se puede usar un archivo en su lugar gracias al formato `.qgz`. En este caso, los archivos `.qgd` y `.qgs` están incrustados en el archivo.

12.1.12 Propiedades de acciones

La pestaña  *Acciones* ofrece la posibilidad de realizar una acción basada en los atributos de un objeto espacial. Esto se puede utilizar para realizar cualquier número de acciones, por ejemplo, ejecutar un programa con argumentos construidos a partir de los atributos de un objeto espacial o pasar parámetros a una herramienta de informes web.

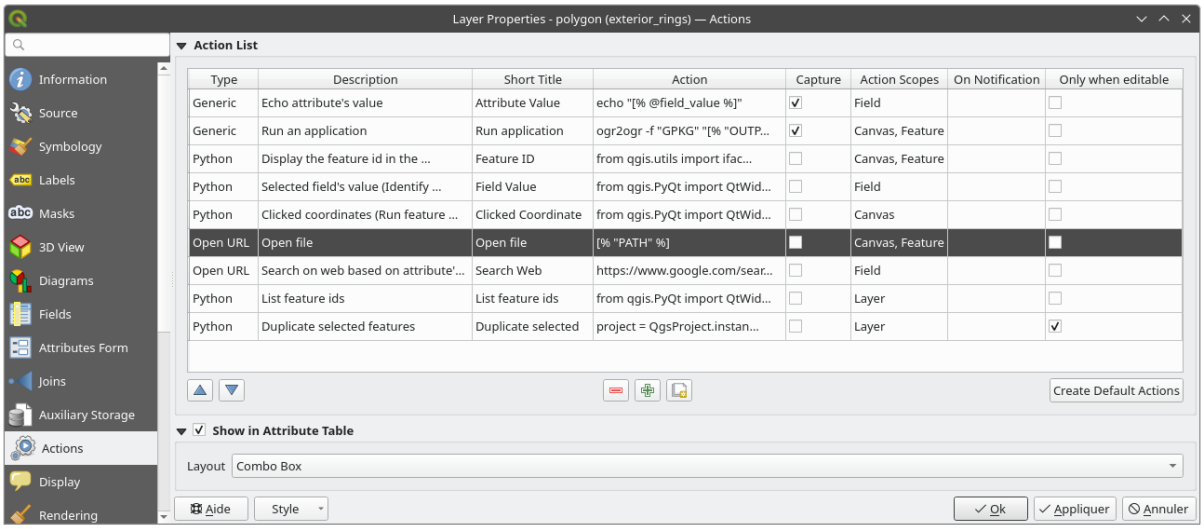


Figura 12.58: Revisión del diálogo de acción con algunas acciones de muestra

Las acciones son útiles cuando con frecuencia desea ejecutar una aplicación externa o ver una página web basándose en uno o más valores de su capa vectorial. Las hay de diferentes tipos y pueden utilizarse así:

- Las acciones *Genérico*, *macOS*, *Windows* y *Unix* inician un proceso externo.
- *Python* las acciones ejecutan una expresión Python.
- Las acciones *Genérico* y *Python* son visibles en todas partes.
- Las acciones *macOS*, *Windows* y *Unix* sólo son visibles en la plataforma respectiva (es decir, puede definir tres acciones «Editar» para abrir un editor y los usuarios sólo podrán ver y ejecutar la acción «Editar» correspondiente a su plataforma para ejecutar el editor).
- *Abrir URL*: Utiliza una petición HTTP GET para abrir una URL proporcionada.
- *Enviar URL (urlencoded o JSON)*: Igual que la acción *Open URL* pero utilizando una petición HTTP POST. Los datos se envían a una URL, utilizando «application/x-www-form-urlencoded» o «application/json» si el cuerpo es un JSON válido.

  Un ejemplo de llamada a la acción podría ser:

  ```
  http://localhost:8000?/[% url_encode(map('file', 'index.php')) %]
  ```

- *Enviar URL (multiparte)*: Igual que la acción *Abrir URL* pero utilizando una petición HTTP POST. Los datos se envían a una URL, utilizando «multipart/form-data».

Hay varios ejemplos incluidos en el diálogo. Puede cargarlos haciendo clic en *Create Default Actions*. Para editar cualquiera de los ejemplos, haga doble clic en su fila. Un ejemplo es realizar una búsqueda basada en un valor de atributo. Este concepto se utiliza en la siguiente discusión.

El *Show in Attribute Table* le permite mostrar en el cuadro de diálogo de la tabla de atributos las acciones marcadas de ámbito de función, ya sea como *Combo Box* o como *Separate Buttons* (ver *Configurando las columnas*).

Definir Acciones

Para definir una acción de atributo, abra el diálogo de vectorial *Layer Properties* y haga click en la pestaña *Actions*.

En la pestaña *Actions*, haga click en Add a new action para abrir el diálogo *Edit Action*.

Seleccione la acción *Tipo* y proporcione un nombre descriptivo para la acción. La propia acción debe contener el nombre de la aplicación que se ejecutará cuando se invoque la acción. Puede añadir uno o más valores de campo de atributo como argumentos de la aplicación. Cuando se invoque la acción, cualquier conjunto de caracteres que empiece por `%` seguido del nombre de un campo será sustituido por el valor de dicho campo. Los caracteres especiales `%%` se sustituirán por el valor del campo que se haya seleccionado en la tabla de resultados o atributos de la identificación (véase *Empleando Acciones*). Las comillas dobles se pueden utilizar para agrupar texto en un único argumento para el programa, script o comando. Las comillas dobles se ignorarán si van precedidas de una barra invertida.

Las acciones pueden invocar un único proceso, con argumentos, por lo que los operadores booleanos (como `&`, `&&`, `;`, `|`) no funcionarán. En sistemas operativos tipo UNIX se pueden ejecutar múltiples órdenes mediante `bash -c`.

La etiqueta *Ámbitos de acción* le permite definir dónde debe estar disponible la acción. Tiene las siguientes opciones:

1. *Campo*: la acción está disponible al hacer clic con el botón derecho del ratón en la celda de la tabla de atributos, en el formulario de características y en el botón de acción por defecto de la barra de herramientas principal.
2. *Objeto espacial*: la acción está disponible al hacer clic con el botón derecho del ratón en la celda de la tabla de atributos.
3. *Canvas*: La acción está disponible en el botón de acción principal en la barra de herramientas.
4. *Formulario*: la acción sólo está disponible en un formulario de características diseñado mediante modo *drag-and-drop*.
5. *Capa*: la acción está disponible en el botón de acción de la barra de herramientas de la tabla de atributos. Tenga en cuenta que este tipo de acción afecta a toda la capa y no a las características individuales.

Si tiene nombres de campo que son subcadenas de otros nombres de campo (p. Ej., `col1` y `col10`), debe indicar que al rodear el nombre del campo (y el carácter %) con corchetes (p. ej., `[%col10]`). Esto evitará que el nombre de campo `%col10` sea confundido con el nombre de campo `%col1` con un `0``al final. QGIS eliminará los corchetes cuando sustituya el valor del campo. Si desea que el campo sustituido esté rodeado de corchetes, use un segundo conjunto como este:``[[%col10]]`.

Usando la herramienta *Identify Features*, puede abrir el diálogo *Identify Results*. Incluye un elemento *(Derived)* que contiene información relevante para el tipo de capa. Se puede acceder a los valores de este elemento de manera similar a los otros campos al proceder con el nombre del campo derivado con `(Derived).`. Por ejemplo, una capa de puntos tiene un campo `X` e `Y`,y los valores de estos campos se pueden usar en la acción con `%(Derived).X` y `%(Derived).Y`. Los atributos derivados solo están disponibles en la caja de diálogo *Identify Results*, no la caja de diálogo *Attribute Table*.

A continuación se muestran dos acciones de ejemplo:

- `konqueror https://www.google.com/search?q=%nam`
- `konqueror https://www.google.com/search?q=%%`

En el primer ejemplo, se invoca el navegador web konqueror y se le pasa una URL para abrir. La URL realiza una búsqueda en Google sobre el valor del campo `nam` de nuestra capa vectorial. Tenga en cuenta que la aplicación o script invocado por la acción debe estar en la ruta o debe proporcionar la ruta completa. Para estar seguros, podríamos reescribir el primer ejemplo como: `/opt/kde3/bin/konqueror https://www.google.com/search?q=%nam`. Esto asegurará que la aplicación konqueror se ejecutará cuando se invoque la acción.

El segundo ejemplo usa la notación %%, que no se basa en un campo particular para su valor. Cuando se invoca la acción, el %% será reemplazado por el valor del campo seleccionado in los resultados de identificación o tabla de atributos.

Duplicar acciones

QGIS permite duplicar acciones existentes. Para duplicar una acción de atributo, abra el cuadro de diálogo del vector *Propiedades de capa* y haga clic en la pestaña *Acciones*. En la pestaña *Acciones*, haga clic en el botón **|duplicarDisposición|** Duplicar una acción para abrir el cuadro de diálogo *Duplicar acción*. Debe haber seleccionado al menos una acción existente para poder crear un duplicado.

En el cuadro de diálogo que aparece, realice los cambios que sean necesarios. Consulte añadir_acciones para más información. Cuando haya terminado, pulse *Aceptar* para crear un duplicado de la acción con los cambios que haya realizado. Si no ha editado la descripción, o si la ha cambiado para que sea idéntica a la descripción de cualquier otra acción existente, se añadirá «_1» al final de la misma.

Empleando Acciones

QGIS ofrece muchas formas de ejecutar acciones que habilitó en una capa. Dependiendo de su configuración, pueden estar disponibles:

- En el botón del menú desplegable Run Feature Action de la *Attributes toolbar* o diálogo *Attribute table*;
- al hacer click derecho en una entidad con la herramienta Identify Features (ver *Identificando entidades* para mas información);
- del panel *Identify Results* , bajo la sección *Actions*;
- como elementos de una columna *Actions* en el diálogo *Attribute Table*.

Si se está invocando una acción que utilice la notación ``%% `` haga clic derecho en el valor del campo en el cuadro de diálogo *Identify Results* o en el diálogo *Attribute Table* que desee pasar a la aplicación o script.

Aquí hay otro ejemplo que extrae datos de una capa vectorial y los inserta en un archivo usando bash y el comando `echo` (así que solo funcionará en o quizás). La capa en cuestión tiene campos para el nombre de una especie `taxon_name`, latitud `lat` y longitud `long`. Nos gustaría poder hacer una selección espacial de localidades y exportar estos valores de campo a un archivo de texto para el registro seleccionado (se muestra en amarillo en el área del mapa QGIS). Aquí está la acción para lograr esto:

```
bash -c "echo \"%taxon_name %lat %long\" >> /tmp/species_localities.txt"
```

Después de seleccionar algunas localidades y ejecutar la acción en cada una, al abrir el archivo de salida mostrará algo como esto:

```
Acacia mearnsii -34.0800000000 150.0800000000
Acacia mearnsii -34.9000000000 150.1200000000
Acacia mearnsii -35.2200000000 149.9300000000
Acacia mearnsii -32.2700000000 150.4100000000
```

Como ejercicio, podemos crear una acción que haga una búsqueda en Google en la capa `` lagos “”. Primero, necesitamos determinar la URL requerida para realizar una búsqueda de una palabra clave. Esto se hace fácilmente con solo ir a Google y hacer una búsqueda simple, luego tomar la URL de la barra de direcciones en su navegador. De este pequeño esfuerzo, vemos que el formato es https://www.google.com/search?q=QGIS, donde `QGIS` es el término de búsqueda. Armados con esta información, podemos proceder:

1. Asegúrese de que la capa de `lakes` esté cargada.
2. Abriendo el diálogo *Layer Properties* mediante doble-click en la capa en la leyenda, o haga clic derecho y elija *Properties* del menú emergente.
3. Haga clic en la pestaña *Acciones*

4. Click  Add a new action.
5. Elija el tipo de acción *Abrir URL*,
6. Ingrese un nombre para la acción , por ejemplo `Búsqueda de Google`.
7. Adicionálmente puede añadir un *Short Name* o incluso un *Icon*.
8. Elija la *Ámbitos de acción*. Véase *Definir Acciones* para más información. Deje la configuración por defecto para este ejemplo.
9. Para la acción, añada la URL utilizada para realizar una búsqueda en Google, hasta el término de búsqueda, pero sin incluirlo: `https://www.google.com/search?q=`.
10. El texto del campo *Acción* debería tener ahora este aspecto:

```
https://www.google.com/search?q=
```

11. Haga clic en el cuadro desplegable que contiene los nombres de campo para la capa `lakes`. Está ubicada justo a la izquierda del botón *Insert*.
12. En el cuadro desplegable, seleccione *NAMES* y haga clic en *Insertar*.
13. Su texto de acción ahora se ve así:

```
https://www.google.com/search?q=[%NAMES%]
```

14. Para finalizar y añadir la acción, click en el botón *OK*.

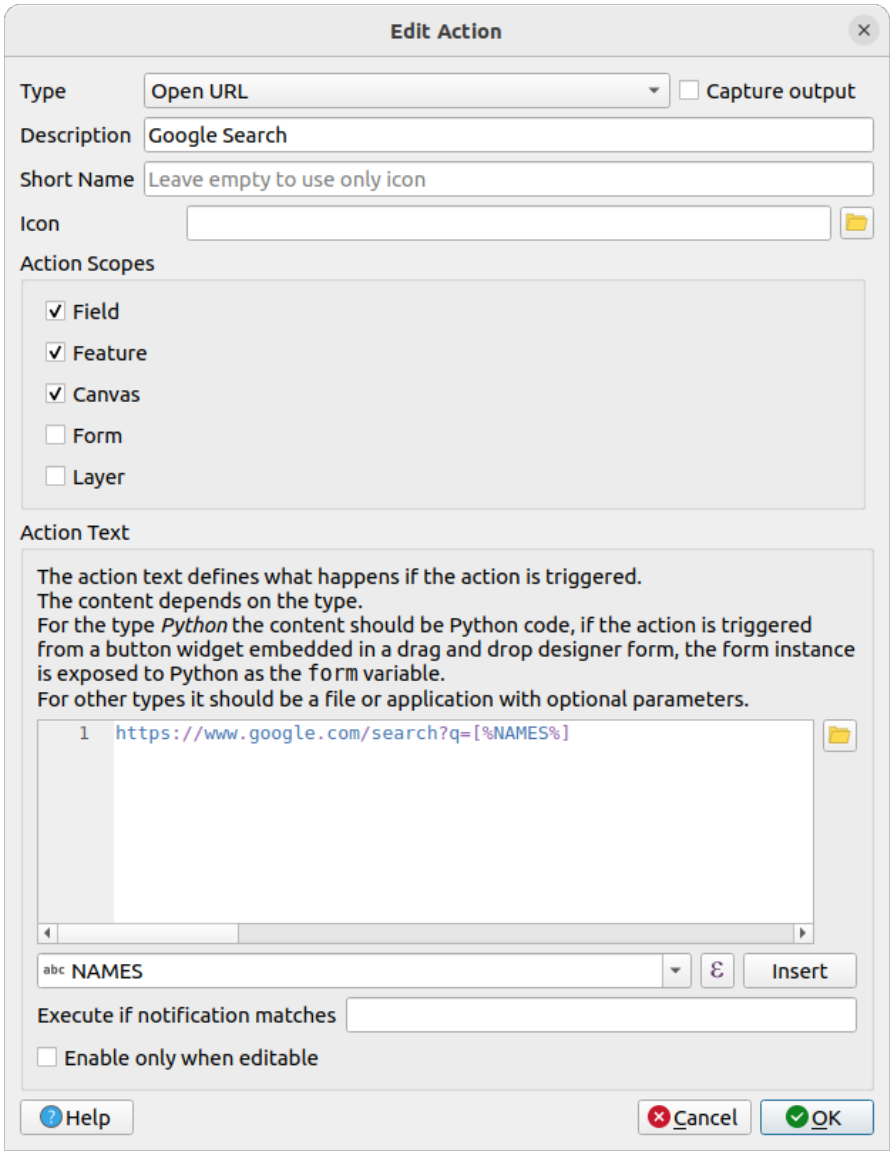


Figura 12.59: Diálogo de edición de acción configurado con el ejemplo

Esto completa la acción, y está listo para usar.

Cierre el cuadro de diálogo *Propiedades de capa* y acerque el zoom a un área de interés. Asegúrese de que la capa `lakes` está activa e identifique un lago. En el cuadro de resultados verá ahora que nuestra acción es visible:

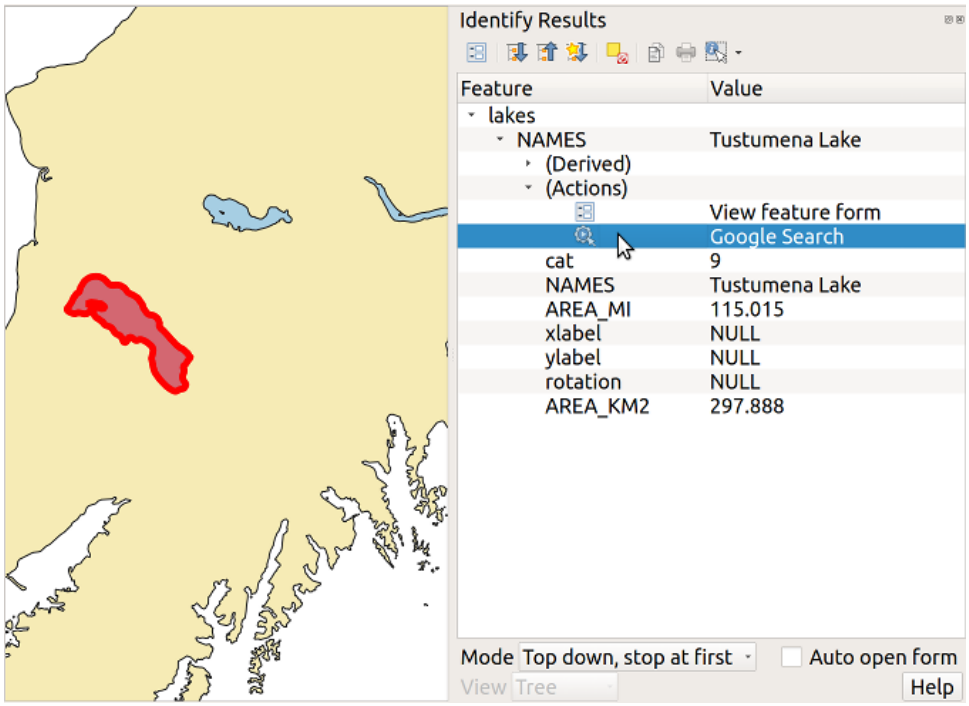


Figura 12.60: Seleccionar objetos espaciales y elegir una acción

Cuando hacemos clic en la acción, se abre nuestro navegador por defecto y se navega a la URL https://www.google.com/search?q=Tustumena. También es posible añadir más campos de atributo a la acción. Por lo tanto, puede añadir un + al final del texto de la acción, seleccionar otro campo y hacer clic en *Insertar campo*. En este ejemplo, no hay ningún otro campo disponible que tenga sentido buscar.

Puede definir varias acciones para una capa, y cada una aparecerá en el cuadro de diálogo. *Identify Results*.

También puede invocar acciones desde la tabla de atributos seleccionando una fila y haciendo clic derecho, luego eligiendo la acción del menú emergente.

Hay todo tipo de usos para las acciones. Por ejemplo, si tiene una capa de puntos que contiene ubicaciones de imágenes o fotos junto con un nombre de archivo, puede crear una acción para iniciar un visor para mostrar la imagen. También puede usar acciones para lanzar informes basados en la web para un campo de atributo o combinación de campos, especificándolos de la misma manera que lo hicimos en nuestro ejemplo de búsqueda de Google.

También podemos hacer ejemplos más complejos, por ejemplo, usando acciones **Python**.

Por lo general, cuando creamos una acción para abrir un archivo con una aplicación externa, podemos usar rutas absolutas, o eventualmente rutas relativas. En el segundo caso, la ruta es relativa a la ubicación del archivo ejecutable del programa externo. Pero, ¿qué pasa si necesitamos usar rutas relativas, relativas a la capa seleccionada (una basada en archivos, como Shapefile o SpatiaLite)? El siguiente código hará el truco:

```
command = "firefox"
imagerelpath = "images_test/test_image.jpg"
layer = qgis.utils.iface.activeLayer()
import os.path
layerpath = layer.source() if layer.providerType() == 'ogr'
  else (qgis.core.QgsDataSourceURI(layer.source()).database()
    if layer.providerType() == 'spatialite' else None)
path = os.path.dirname(str(layerpath))
image = os.path.join(path,imagerelpath)
import subprocess
subprocess.Popen( [command, image ] )
```

Solo tenemos que recordar que la acción es de tipo *Python* y el *command* y las variables *imagerelpath* deben ser cambiadas para satisfacer nuestras necesidades.

Pero, ¿qué pasa si la ruta relativa debe ser relativa al archivo de proyecto (guardado)? El código de la acción de Python sería:

```
command = "firefox"
imagerelpath = "images_test/test_image.jpg"
projectpath = qgis.core.QgsProject.instance().fileName()
import os.path
path = os.path.dirname(str(projectpath)) if projectpath != '' else None
image = os.path.join(path, imagerelpath)
import subprocess
subprocess.Popen( [command, image ] )
```

Otro ejemplo de acción de Python es el que nos permite agregar nuevas capas al proyecto. Por ejemplo, los siguientes ejemplos agregarán al proyecto, respectivamente, un vector y un ráster. Los nombres de los archivos que se agregarán al proyecto y los nombres que se darán a las capas están controlados por datos (*filename* y *layername* son nombres de columna de la tabla de atributos del vector donde se creó la acción):

```
qgis.utils.iface.addVectorLayer('/yourpath/[% "filename" %].shp',
  '[% "layername" %]', 'ogr')
```

Para añadir un ráster (una imagen TIF en este ejemplo), se vuelve:

```
qgis.utils.iface.addRasterLayer('/yourpath/[% "filename" %].tif',
  '[% "layername" %]')
```

12.1.13 Propiedades a mostrar

La pestaña *Mostrar* le ayuda a configurar los campos que se utilizarán para la identificación de objetos espaciales:

- La *Nombre de visualización*: basada en un campo o en una *expresión*. De forma predeterminada, se establece en el primer campo de la capa si no hay ningún campo con componente `<name>` existe. Se utiliza como:
  - la etiqueta mostrada sobre la información de la entidad en la *herramienta identificar* resulta
  - el campo usado en la *barra de localización* al buscar entidades en todas las capas
  - el identificador de entidad en la tabla de atributos *vista de formulario*
  - el identificador del objeto espacial cuando el mapa o el esquema se exportan a un formato de salida por capas como el PDF geoespacial
  - la información de consejos sobre el mapa, es decir, el mensaje que aparece en el lienzo del mapa cuando se pasa el ratón por encima de un objeto espacial de la capa activa con el icono Mostrar consejos de mapa pulsado. Aplicable cuando *Habilitar consejos de mapa* está activo y no se ha definido *Consejo de mapa HTML*.
- *Habilitar consejos de mapa* controla si se muestran consejos de mapa para la capa
- El *Consejo de mapa HTML* proporciona un complejo y completo editor de texto HTML para consejos de mapa, mezclando expresiones QGIS y estilos y etiquetas html (multilínea, fuentes, imágenes, hipervínculo, tablas, …). Puede comprobar el resultado de su muestra de código en el marco *Preview* (también conveniente para previsualizar la salida de *Nombre de visualización*). Además, puede seleccionar y editar expresiones existentes utilizando el botón *Insertar/Editar expresión*.

> **Nota**
>
> Comprender el comportamiento del botón :guilabel:`Insertar/Editar expresión

Si selecciona algún texto dentro de una expresión (entre «[%» y «%]»), o si no hay ningún texto seleccionado pero el cursor está dentro de una expresión, se seleccionará automáticamente toda la expresión para editarla. Si el cursor o un texto seleccionado está fuera de una expresión, el diálogo se abre con la selección.

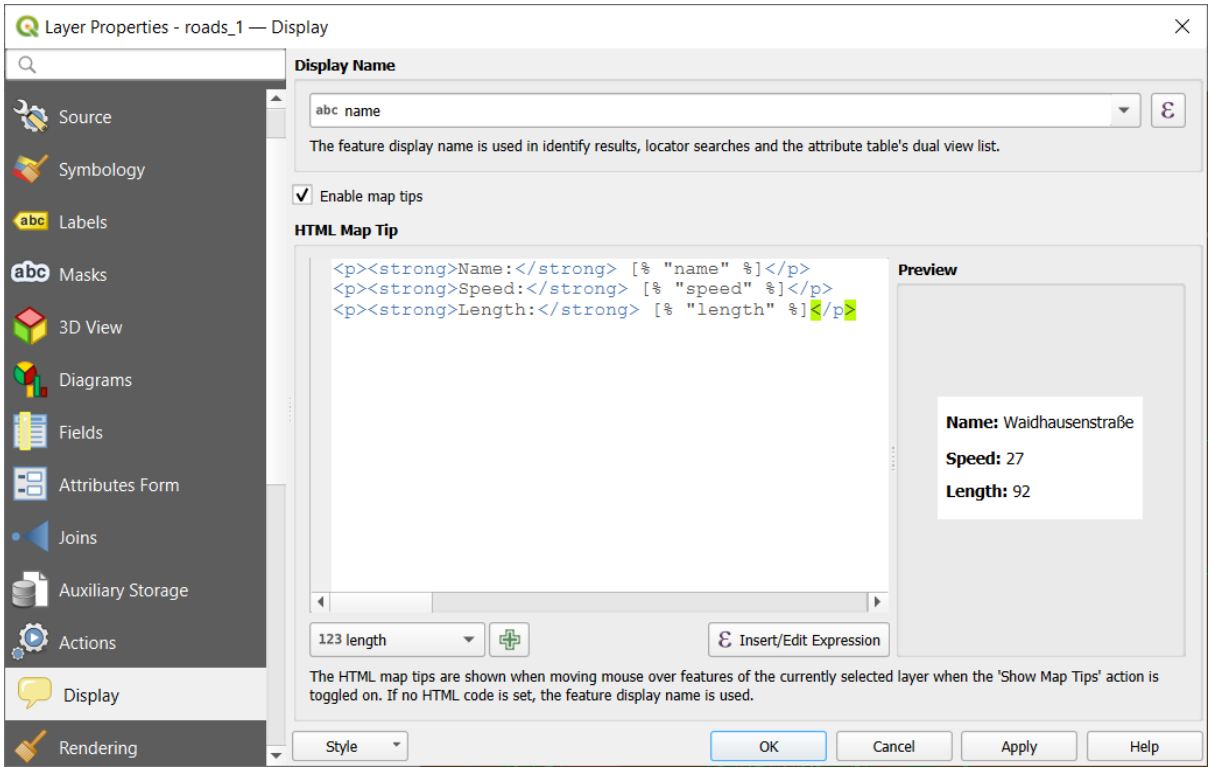


Figura 12.61: Código HTML para textos de aviso del mapa

Para mostrar consejos sobre el mapa:

1. Seleccione la opción de menú *Ver* ► *Mostrar consejos sobre el mapa* o haga clic en el icono  Mostrar consejos sobre el mapa de la barra de herramientas *Atributos*.
2. Asegúrese de que la capa de destino está activa y tiene marcada la propiedad  *Habilitar consejos de mapa*.
3. Desplácese sobre un objeto espacial y se mostrará la información correspondiente.

Un consejo de mapa es una prestación entre capas, lo que significa que una vez activada, permanece activada y se aplica a cualquier capa habilitada para consejo de mapa en el proyecto hasta que se desactive.



Figura 12.62: Textos de aviso del mapa con código HTML

12.1.14 Propiedades de representación

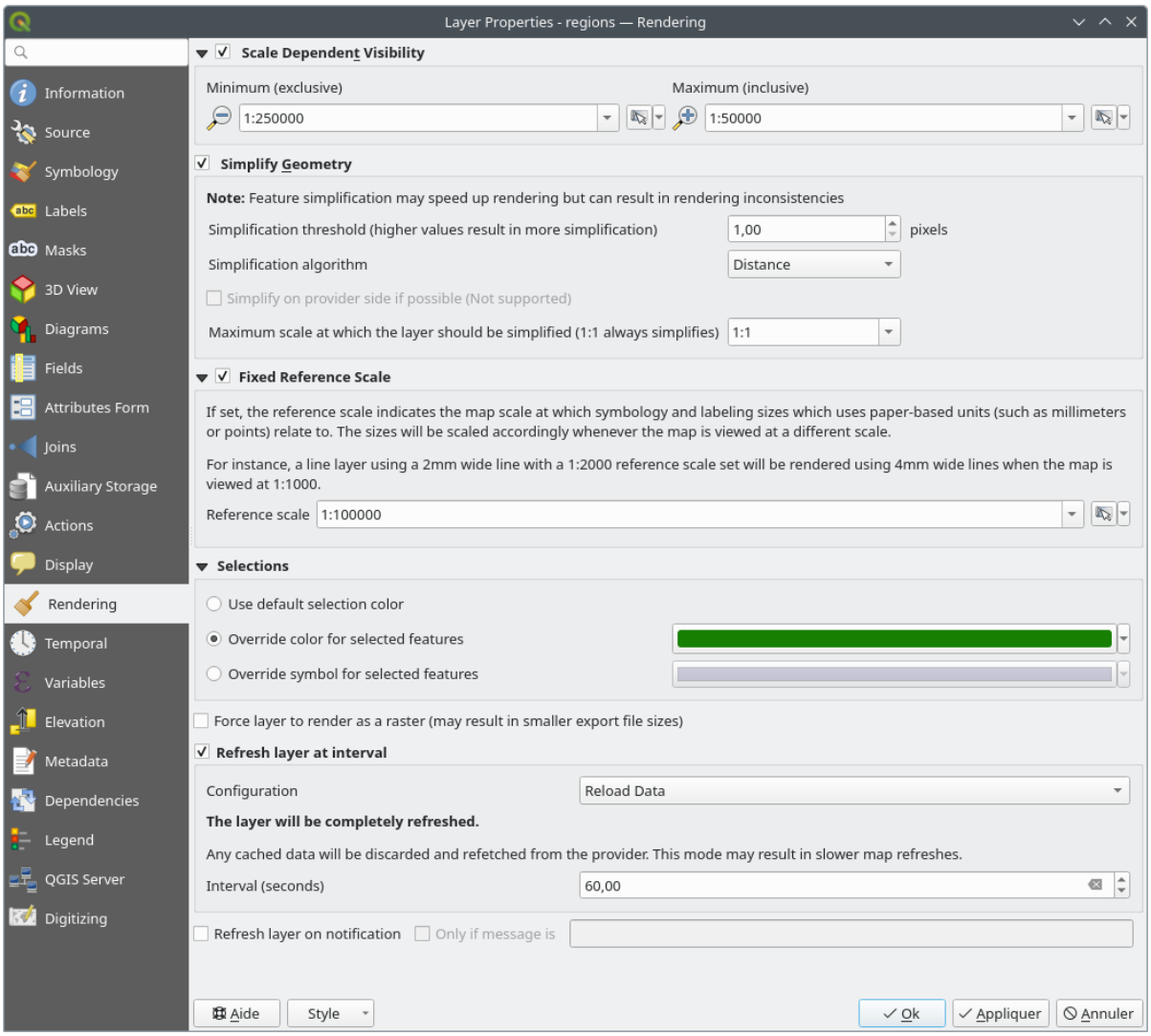


Figura 12.63: Cuadro de diálogo Propiedades de renderizado de capas

La pestaña *Renderización* ofrece las siguientes propiedades:

- En *Visibilidad dependiente de la escala*, puede establecer las escalas *Máxima (inclusive)* y *Mínima (exclusiva)*, definiendo un rango de escalas en el que las características serán visibles. Fuera de este rango, se ocultan. El botón Establecer en la escala actual del lienzo le ayuda a utilizar la escala actual del lienzo del mapa como límite del rango de visibilidad. Véase *Selector de escala de visibilidad* para más información.

  > **Nota**
  >
  > También puede activar la visibilidad dependiente de la escala en una capa desde el panel *Layers*: haga clic con el botón derecho del ratón en la capa y, en el menú contextual, seleccione *Definir la visibilidad de la escala de capas*.

- QGIS ofrece soporte para la generalización de funciones sobre la marcha. Esto puede mejorar los tiempos de renderizado al dibujar muchas características complejas a escalas pequeñas. Esta característica se puede habilitar o deshabilitar en la configuración de la capa utilizando la opción *Simplify geometry* . También

hay una configuración global que permite la generalización de forma predeterminada para las capas recién agregadas (consulte *global simplification* para mas información).

> **Nota**
>
> La generalización de entidades puede introducir artefactos en su salida renderizada en algunos casos. Estos pueden incluir astillas entre polígonos y representación imprecisa cuando se usan capas de símbolos basadas en desplazamiento.

- La etiqueta *Escala de referencia fija* indica la escala del mapa a la que se refieren los tamaños de simbología y etiquetado que utilizan unidades basadas en papel (como milímetros o puntos). Los tamaños se escalarán en consecuencia siempre que el mapa se visualice a una escala diferente.

  Por ejemplo, una capa de líneas que utilice una línea de 2 mm de ancho con una escala 1:2000 *Escala de referencia* establecida se representará utilizando líneas de 4 mm de ancho cuando el mapa se vea a 1:1000.

- El grupo *Selecciones* permite controlar si se debe utilizar un color o símbolo específico en lugar de los predeterminados (*Propiedades del proyecto* ► *General* ► *Color de selección*) para una capa concreta. Esto es útil para mejorar la visibilidad de los objetos espaciales seleccionados con cierta simbología:

  - *Utilizar color de selección predeterminado*.
  - *Sustituir el color de los objetos espaciales seleccionados*: por ejemplo, si la capa utiliza un color amarillo por el predeterminado y la selección amarilla estándar no es visible.
  - *Sustituir símbolo para objetos espaciels seleccionados*: por ejemplo, si una capa de líneas utiliza un símbolo fino, y colorear las líneas puede no hacerlas suficientemente visibles, sustituir el símbolo por una línea más gruesa puede ayudar. Además, si la capa utiliza simbología de trama o rellenos de degradado/líneas/formas con simbología de rampa de color, el color de selección predeterminado no se aplica en absoluto; puede ser útil poder establecer un símbolo específico más sencillo para los objetos espaciales seleccionados en la capa.

- El renderizado de capas extremadamente detalladas (por ejemplo, capas poligonales con un gran número de nodos), puede hacer que las exportaciones de diseño en formato PDF/SVG sean enormes, ya que todos los nodos se incluyen en el archivo exportado. Esto también puede hacer que el archivo resultante sea muy lento de trabajar/abrir en otros programas.

  Chequeando ☑ *Force layer to render as raster* obliga a que estas capas se rastericen para que los archivos exportados no tengan que incluir todos los nodos contenidos en estas capas y, por lo tanto, la representación se acelera.

  También puede hacerlo forzando la exportación del diseño como trama, pero es una solución de todo o nada, dado que la rasterización se aplica a todas las capas. Alternativamente, puede confiar en la simplificación de la geometría en *configuración de exportación del diseño*.

- ☐ *Refrescar capa a intervalos*: controla si se puede refrescar una capa y con qué periodicidad. Las opciones *Configuración* disponibles son:

  - *Recargar datos*: la capa se actualizará por completo. Cualquier dato almacenado en caché será descartado y recuperado del proveedor. Este modo puede ralentizar la actualización del mapa.
  - *Redibujar sólo capa*: este modo es útil para animaciones o cuando el estilo de la capa se actualizará a intervalos regulares. Las actualizaciones del lienzo se aplazan para evitar que se actualice varias veces si más de una capa tiene establecido un intervalo de actualización automática.

  También es posible establecer el *Intervalo (segundos)* entre refrescos consecutivos.

- Dependiendo del proveedor de datos (p. Ej. PostgreSQL), Se pueden enviar notificaciones a QGIS cuando se aplican cambios a la fuente de datos, fuera de QGIS. Utilice la casilla de verificación | :guilabel:opción *Actualizar capa en notificación* para activar una actualización. También puede limitar la actualización de la capa a un conjunto de mensajes específico en la caja de texto ☑ *Only if message is.*

12.1.15 Propiedades Temporales

El *Temporal* proporciona opciones para controlar el renderizado de la capa a lo largo del tiempo. Esta representación dinámica requiere que la *temporal navigation* esté habilitada en el lienzo del mapa.

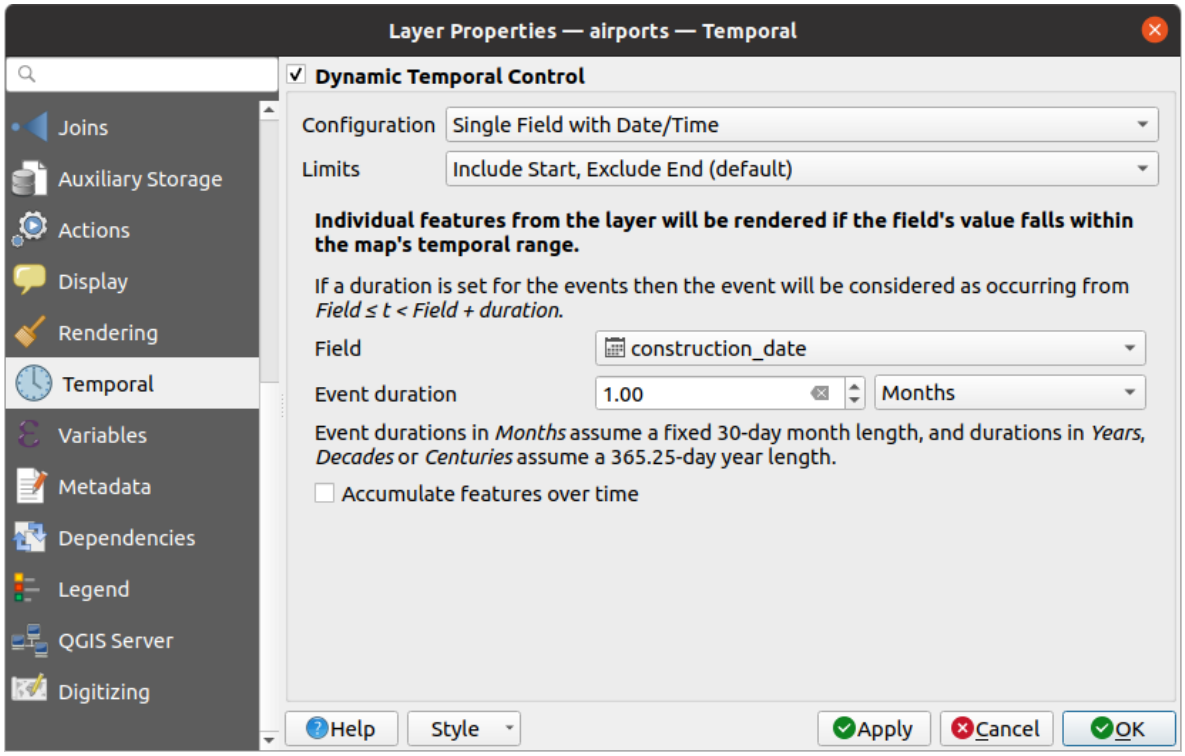


Figura 12.64: Diálogo de propiedades temporales de la capa vectorial

Marque la |casilla| *Dynamic Temporal Control* para configurar el renderizado temporal de la capa vectorial. Dependiendo de la estructura de su conjunto de datos, es posible que desee utilizar una de las opciones de *Configuration* provistas:

- *Fixed time range*: todas las características se representan si el marco temporal del lienzo del mapa se superpone al rango dado de *Start date* y *End date*.
- *Single field with date/time*: las características se representan si su valor de *Field* se encuentra dentro del marco temporal del lienzo del mapa. Se puede establecer una *Event duration*. Al marcar la opción *Accumulate features over time*, todas las entidades que ocurran antes o dentro del rango temporal del mapa continuarán mostrándose. Por lo tanto, se ignora la duración del evento.
- *Separate fields for start and end date/time*: las características se representan si el rango especificado por sus valores de *Start field* y *End field* se superpone al lienzo del mapa temporal.
- *Separate fields for start and event duration*: las características se representan si el rango definido por sus valores de *Start field* y *Event duration field* se superpone al lienzo del mapa temporal.
- *Fecha/hora de inicio y fin de las expresiones*: las características se representan si el rango de tiempo especificado por los campos *Expresión de inicio* y *Expresión de fin* se superpone al temporal del lienzo del mapa.
- *Redraw layer only*: la capa se vuelve a dibujar en cada nuevo cuadro de animación, pero no se aplica ningún filtrado basado en el tiempo a las entidades. Es útil cuando la capa usa valores de expresión basados en el tiempo para la configuración del renderizador (por ejemplo, simbología definida por datos).

También es posible establecer los *Limits* del rango de tiempo de las características como:

- *Incluye inicio, excluye final*
- *Incluye inicio, incluye final*

12.1.16 Propiedades de variables

La pestaña *Variables* lista todas las variables disponibles a nivel de capa (que incluye todas las variables globales y de proyecto).

También permite al usuario administrar variables de nivel de capa. Haga clic en el botón para agregar una nueva variable de nivel de capa personalizada. Del mismo modo, seleccione una variable de nivel de capa personalizada de la lista y haga clic en el botón para borrarla.

Más información sobre el uso de variables en la sección de herramientas Generales *Almacenando valores en variables*

12.1.17 Propiedades de elevación

La pestaña *Elevación* proporciona opciones para controlar las propiedades de elevación de la capa dentro de una vista de mapa *3D* y su apariencia en los *gráficos de la herramienta de perfil*. En concreto, puede establecer:

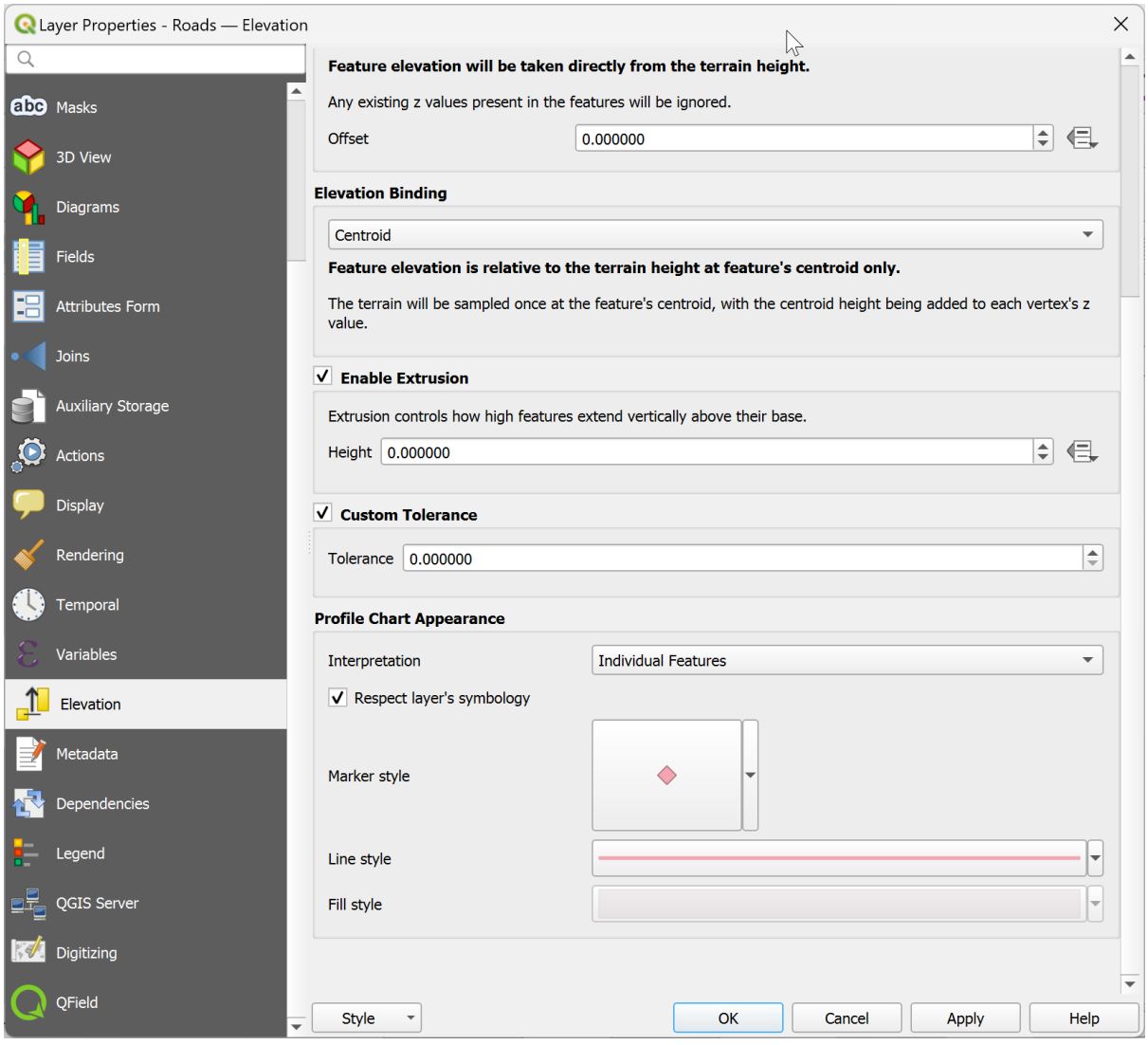


Figura 12.65: Diálogo de propiedades de elevación de la capa vectorial

- *Sistema de Referencia Vertical*: Si el SRC de su capa vectorial es compuesto (incluyendo una dimensión Z), entonces el SRC vertical utilizado para la capa será el componente vertical del SRC de la capa. En este caso, no

puede establecer manualmente un CRS vertical diferente. Si su SRC de capa es horizontal (2D), entonces puede seleccionar un SRC vertical específico haciendo clic en el botón  Seleccionar SRC. Los sistemas de referencia verticales son compatibles con las capas vectoriales mediante:

  - *Perfiles de elevación*
  - *Identificar resultados de la herramienta*
  - *Vistas de mapas 3D*

- *Sujeción por elevación*: define cómo y si debe ser la altitud de los objetos espaciales:
  - *Sujeto al terreno*: toma la elevación directamente de la altura del terreno e ignora cualquier valor Z existente en las características. También se puede rellenar un valor *Desplazamiento* del terreno definido por los datos.
  - *Relativo al terreno*: cualquier valor Z existente en los objetos espaciales se añade a la altura del terreno. Puede utilizarse un factor *Escala* seguido de un *Desplazamiento* definido por los datos para ajustar la elevación. Esta opción no está disponible para capas de geometría 2D.
  - *Absoluto*: ignora la altura del terreno y toma directamente los valores Z de los objetos espaciales para la elevación. Puede utilizarse un factor *Escala* seguido de un valor *Desplazamiento* definido por los datos para ajustar la elevación. En el caso de capas geométricas 2D (sin valores Z), puede definirse en su lugar una *Altura base* definida por datos.
-  *Activar extrusión*: puede establecer una *Altura* para controlar la altura a la que los objetos espaciales se extienden verticalmente por encima de su base. Esto es conveniente para indicar que una capa de geometría 2D, por ejemplo una capa de polígonos de huellas de edificios, representa en realidad objetos 3D.
-  *Custom tolerance*: allows you to override the global sampling tolerance defined in the elevation profile settings, specifically for this vector layer. When enabled, you can set *Tolerance* that determines how the layer is sampled along the profile line. This is useful when you need finer or coarser sampling than the global default, for example, to improve performance or accuracy for complex geometries.
- *Fijación por elevación*: sólo relevante cuando se combina una *Sujeción por elevación* que depende del terreno con una capa de líneas o polígonos, esta opción controla cómo se establece la elevación de la característica en relación con la altura del terreno. El terreno puede ser muestreado:
  - en el *Centroide* del objeto espacial, añadiendo la altura del centroide al valor z de cada vértice
  - en cada *Vértice* individual antes de ser añadido al valor z del vértice
- *Apariencia del gráfico de perfil*: controla cómo se representan los objetos espaciales al dibujar un gráfico de perfil. Existen dos modos principales de *Interpretación*:
  - as *Objetos espaciales individuales*: muestrea posiciones discretas en las que la línea del perfil de la sección transversal interseca las características del vector. Esa intersección puede representarse como punto, línea o superficie en función del tipo de capa y de si se aplica una extrusión.

    Al marcar  *Respetar simbología de capa*, los objetos espaciales se representarán en el gráfico de perfil con su correspondiente *estilo de capa* (permitiendo, por ejemplo, que las clases categorizadas sean visibles en el gráfico de perfil). Si el tipo de símbolo del perfil no coincide con los tipos de símbolo del renderizador de la capa, sólo se aplica al símbolo del perfil el color del símbolo del renderizador.

    Dependiendo de la configuración de la capa, los símbolos de perfil pueden representarse con un estilo personalizado, utilizando:

    * *Estilo de marcador*: para los elementos puntuales y lineales no extruidos, y para los elementos poligonales no extruidos tocados por la línea de perfil
    * *Estilo de línea*: para los elementos puntuales y lineales extruidos, y para los elementos poligonales no extruidos intersecados por la línea de perfil
    * *Estilo de relleno*: para objetos espaciales polígonos extruidos

- as *Superficie continua (por ejemplo, curvas de nivel)*: el gráfico de elevación se representará como una superficie en lugar de como objetos espaciales separados, uniendo los resultados de elevación muestreados en una línea continua. Esto puede mejorar la visualización y está diseñado para capas vectoriales que representan una superficie de elevación continua, por ejemplo, curvas de nivel o puntos de elevación topográfica. El perfil *Estilo* puede establecerse como:
  - una *Línea* con un estilo *Línea*
  - una superficie de elevación representada utilizando un símbolo de relleno por encima (*Rellenar por encima*) o por debajo (*Rellenar por debajo*) de la línea curva de elevación. La simbología de la superficie se representa utilizando:
    - un *Estilo de relleno*
    - y una :guilabel:"Límite": la altitud máxima (o mínima) que determina la altura de la superficie de relleno.

  Además, puede marcar *Mostrar marcadores en los puntos muestreados* para hacerlos visibles sobre la línea de interpretación y asignarles un *Estilo de marcador*.

12.1.18 Propiedades de metadatos

El *Metadata* le brinda opciones para crear y editar un informe de metadatos en su capa. Consulte *Metadatos* para obtener más información.

12.1.19 Propiedades de dependencias

La pestaña **|dependencias|** *Dependencias* permite declarar dependencias de datos entre capas. Una dependencia de datos se produce cuando una modificación de datos en una capa, no por manipulación directa del usuario, puede modificar datos de otras capas. Este es el caso, por ejemplo, cuando la geometría de una capa es actualizada por un disparador de base de datos o un script personalizado de PyQGIS después de la modificación de la geometría de otra capa.

En la pestaña *Dependencies* , puede seleccionar cualquier capa que pueda alterar externamente los datos en la capa actual. Especificar correctamente las capas dependientes permite a QGIS invalidar las memorias caché para esta capa cuando las capas dependientes se alteran.

12.1.20 Propiedades de la leyenda

La pestaña de propiedades :guilabel:` Leyenda` le proporciona una configuración avanzada para el *panel Capas* y/o la leyenda *diseño de impresión*. Estas opciones incluyen:

- Dependiendo de la simbología aplicada a la capa, puede terminar con varias entradas en la leyenda, no necesariamente legibles/útiles para mostrar. La *Legend placeholder image* le ayuda a *select an image* para reemplazarla, que se muestra tanto en el panel *Layers* como en la leyenda del diseño de impresión.
- *Show label legend*: Muestra vistas generales de las diferentes configuraciones de etiquetas como entradas en las leyendas. El *label style* se muestra junto con la descripción.
- *Text on symbols*: En algunos casos puede ser útil agregar información adicional a los símbolos en la leyenda. Con este marco, puede afectar a cualquiera de los símbolos utilizados en la simbología de capa un texto que se muestra sobre el símbolo, en ambos:guilabel:panel *Capas* y leyenda de diseño de impresión. Esta asignación se realiza escribiendo cada texto al lado del símbolo en el widget de la tabla o llenando la tabla usando el botón *Set Labels from Expression*. La apariencia del texto se maneja a través de los widgets del selector de fuente y color en el botón *Text Format*.

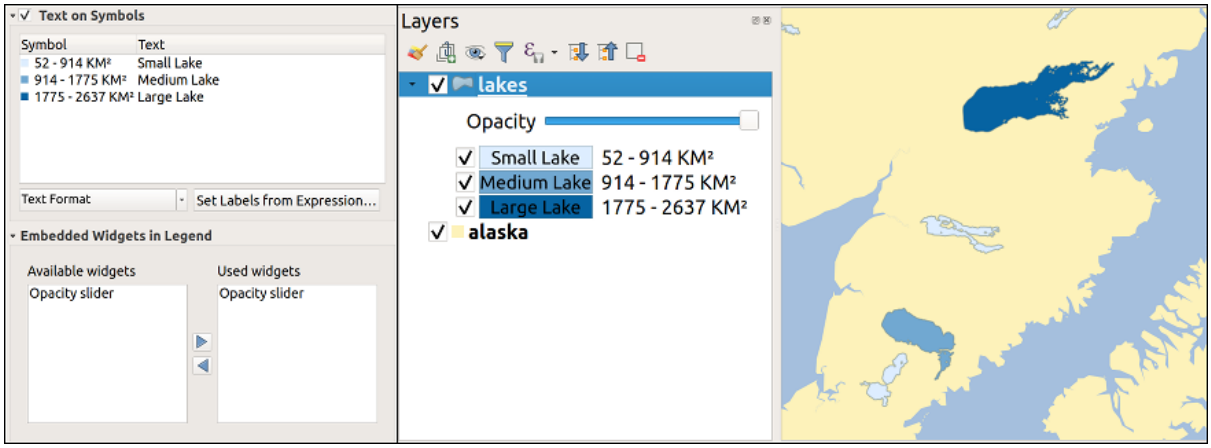


Figura 12.66: Establecer texto en símbolos (izquierda) y su representación en el panel *Layers* (derecha)

- una lista de widgets que puede incrustar dentro del árbol de capas en el panel Capas. La idea es tener una forma de acceder rápidamente a algunas acciones que a menudo se usan con la capa (configuración de transparencia, filtrado, selección, estilo u otras cosas…).

  Por defecto, QGIS proporciona un widget de transparencia, pero esto puede ampliarse mediante complementos que registren sus propios widgets y asignen acciones personalizadas a las capas que administran.

12.1.21 Propiedades de servidor QGIS

La pestaña *QGIS Server* consta de las secciones *Descripción*, *Atribución*, *URL de metadatos* y *URL de leyenda*.

Desde la sección *Descripción*, puede cambiar el *Nombre corto* utilizado para referenciar la capa en las peticiones (para saber más sobre nombres cortos, lea services_basics_short_name). También puede añadir o editar un *Título*, un *Título WFS* alternativo y un *Abstracto* para la capa, o definir una *Lista de palabras clave* aquí. Estas listas de palabras clave pueden utilizarse en un catálogo de metadatos. Si desea utilizar un título de un archivo de metadatos XML, debe rellenar un enlace en el campo *URL de datos*.

Use *Attribution* para obtener datos de atributos de un catálogo de metadatos XML.

En *Metadata URL* se pueden añadir las rutas generales al catálogo de metadatos XML. Esta información se guardará en el archivo de proyecto de QGIS para sesiones posteriores y se utilizará para QGIS Server.

En la sección *Leyenda URL*, puede proporcionar la url de una imagen de leyenda en el campo url. Puede usar la opción desplegable Formato para aplicar el formato apropiado de la imagen. Actualmente se admiten los formatos de imagen png, jpg y jpeg.

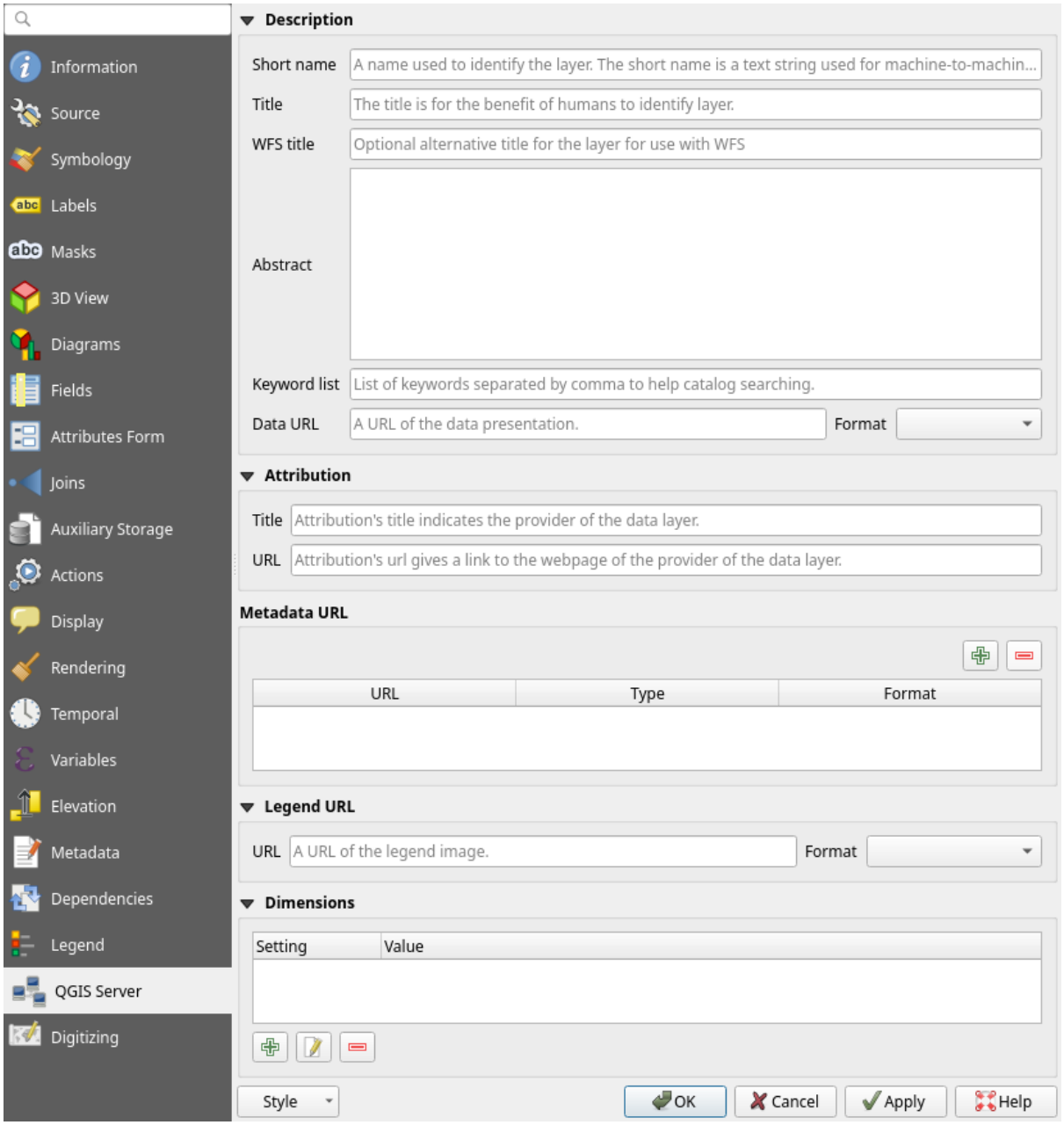


Figura 12.67: La pestaña QGIS Server en el diálogo propiedades de capas vectoriales

Para obtener más información sobre QGIS Server, lea QGIS-Server-manual.

12.1.22 Propiedades de Digitalizado

La pestaña |digitalización| *Digitalización* da acceso a opciones que ayudan a garantizar la calidad de las geometrías digitalizadas.

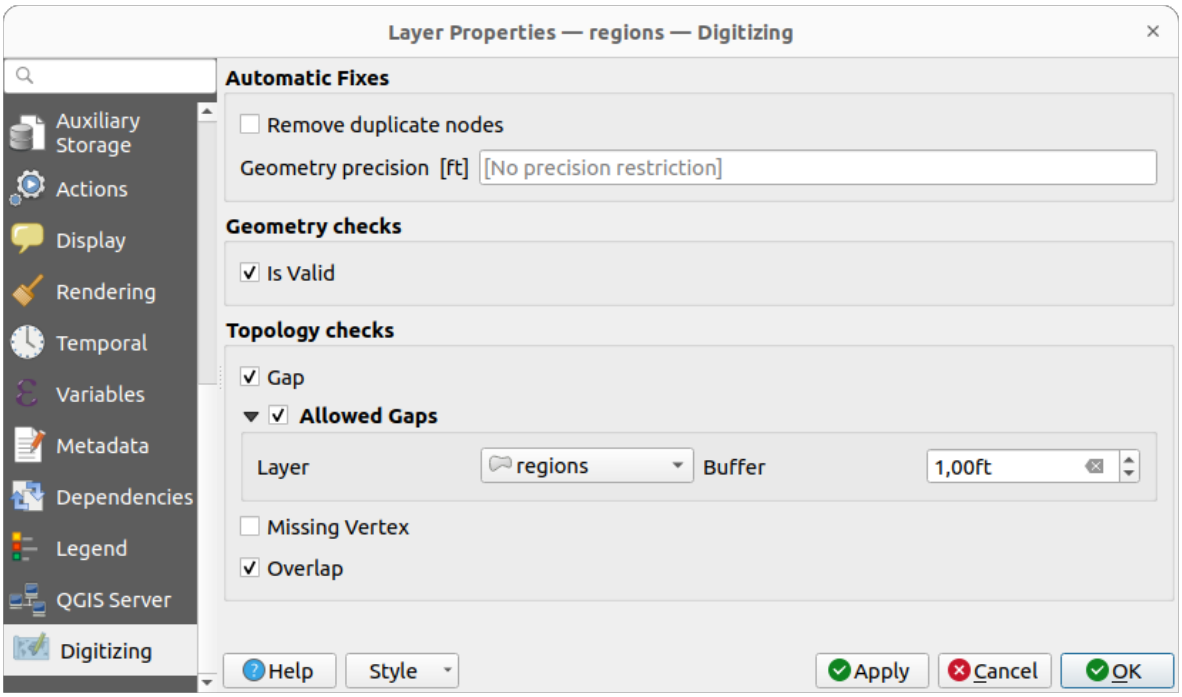


Figura 12.68: La pestaña QGIS Digitizing en el diálogo de propiedades de capas vectoriales

Correcciones automáticas

Opciones en la sección *Automatic Fixes* afectará directamente los vértices de cualquier geometría que se agregue o modifique. Si la opción *Remove duplicate nodes* está marcada, dos vértices subsiguientes cualesquiera con exactamente las mismas coordenadas serán borrados. Si está fijada la *Geometry precision* , todos los vértices se redondearán al múltiplo más cercano de la precisión geométrica configurada. El redondeo ocurrirá en el sistema de referencia de coordenadas de capa. Los valores Z y M no son redondeados. Con muchas herramientas de mapa, se muestra una cuadrícula en el lienzo mientras se digitaliza.

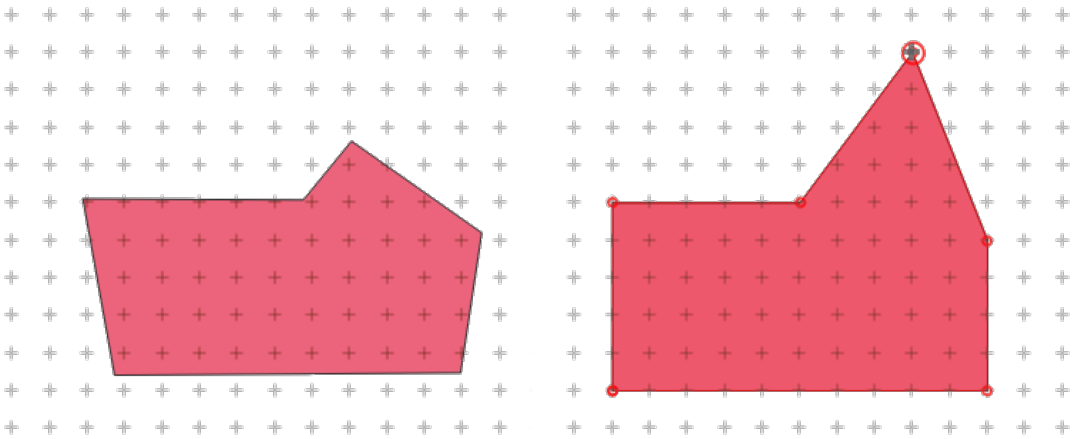


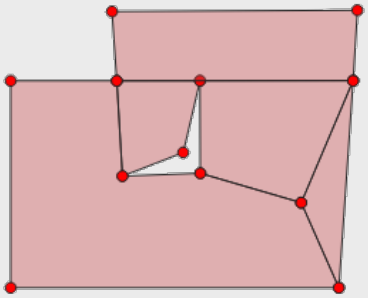
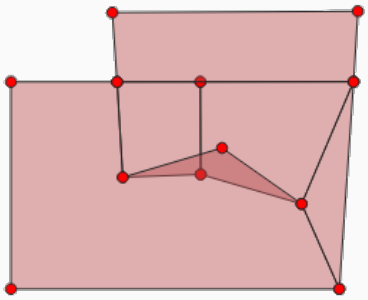
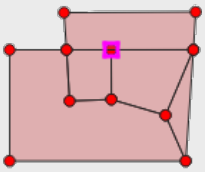
Figura 12.69: Mover el vértice superior ajusta todos los vértices a la cuadrícula

Controles de geometría

En la sección *Geometry checks* se pueden activar validaciones adicionales por geometría. Inmediatamente después de cualquier modificación de la geometría, los fallos en estas comprobaciones se informan al usuario en el panel de *Geometry validation*. Mientras una verificación falle, no es posible guardar la capa. La *Is valid* ejecutará comprobaciones de validez básicas como la autointersección en geometrías.

Controles de Topología

En la sección *Topology checks* se pueden activar comprobaciones adicionales de validación de topología. Las comprobaciones de topología se ejecutarán cuando el usuario guarde la capa. Los errores de comprobación se informarán en el panel *Geometry validation*. Mientras haya errores de validación, la capa no se puede guardar. Las comprobaciones de topología se ejecutan en el área del cuadro delimitador de las entidades modificadas. Dado que otras características pueden estar presentes en la misma área, se informan los errores topológicos relacionados con estas características, así como los errores introducidos en la sesión de edición actual.

Opción de control de topología	Ilustración
El *Gap* chequeo verificará si hay espacios entre los polígonos vecinos.	
El *Overlap* chequeo verificará las superposiciones entre polígonos vecinos.	
El *Missing vertex* chequeo verificrá los límites compartidos de polígonos vecinos donde un borde pierde un vértice que está presente en el otro.	

Excepciones de verificación de brechas

A veces es deseable mantener huecos dentro de un área en una capa de polígono que de otro modo estaría completamente cubierta por polígonos. Por ejemplo, una capa de uso de la tierra puede tener agujeros aceptables para los lagos. Es posible definir áreas que se ignoran en la verificación de huecos. Dado que se permiten huecos dentro de estas áreas, nos referiremos a ellos como áreas *Allowed Gaps*.

En las opciones para las comprobaciones de brecha en *Allowed Gaps*, una *Allowed Gaps layer* puede ser configurada.

Cada vez que se ejecuta la verificación de huecos, los huecos que están cubiertos por uno o más polígonos en el *Allowed Gaps Layer* no son reportados como errores topológicos.

También es posible configurar un adicional *Buffer*. Este búfer se aplica a cada polígono en el *Allowed Gaps Layer*. Esto hace posible que las pruebas sean menos susceptibles a pequeños cambios en los contornos en los bordes de las brechas.

Cuando se activan las *Allowed Gaps* , unbotón adicional (*Add Allowed Gap*) los errores de hueco detectados están disponibles en el muelle de validación de geometría, donde se informan huecos durante la digitalización. Si el botón *Add Allowed Gap* es pulsado, se inserta un nuevo polígono con la geometría del espacio detectado en el *Allowed Gaps Layer*.Esto hace posible marcar rápidamente las brechas según lo permitido.

Panel de validación de geometría

El panel *Geometry Validation* se activa cuando cualquiera de las comprobaciones de digitalización mencionadas anteriormente encuentra un error. El cuadro de diálogo le proporciona la lista de errores y su descripción, y puede navegar por la lista usando las flechas del teclado o las flechas dedicadas.

Deberá solucionar todos los problemas antes de poder guardar las ediciones en la capa. Para hacerlo:

1. Seleccione un error y es posible:
   -  *Zoom al objeto(s)*
   -  *Zoom al problema*
2. Pick the usual *digitizing tools* to fix the issue.

12.2 Trabajar con la tabla de atributos

La tabla de atributos muestra información sobre los elementos de una capa seleccionada. Cada fila en la tabla representa un elemento (con o sin geometría), y cada columna contiene una pieza de información particular acerca de ese elemento. Puede buscar, seleccionar, mover o incluso editar elementos en la tabla.

12.2.1 Prefacio: Tablas espaciales y no espaciales

QGIS permite cargar capas espaciales y no espaciales. Esto incluye actualmente tablas soportadas por GDAL y texto delimitado, así como los proveedores PostgreSQL, MS SQL Server, SpatiaLite y Oracle. Todas las capas cargadas aparecen en el panel *Capas*. El hecho de que una capa esté habilitada espacialmente o no determina si se puede interactuar con ella en el mapa.

Las tablas no espaciales se pueden examinar y editar utilizando la vista de tabla de atributos. Además, se pueden utilizar para búsquedas de campo. Por ejemplo, puede usar columnas de una tabla no espacial para definir valores de atributo, o un rango de valores que están permitidos, para agregarlos a una capa vectorial específica durante la digitalización. Eche un vistazo más de cerca al widget de edición en la sección *Formulario de propiedades de campo* para obtener más información.

12.2.2 Introducción a la interfaz de la tabla de atributos

Para abrir la tabla de atributos de una capa vectorial, active la capa haciendo click en ella en *Panel de capas*. Luego, desde el menú principal:menuselection:*Layer*, elija *Abrir tabla de atributos*. También es posible hacer click con el botón derecho en la capa y elegir *Abrir tabla de atributos* del menú desplegable, o hacer click en el botón *Abrir tabla de atributos* en la barra de herramientas Atributos. Si prefiere los atajos, `F6` abrirá la tabla de atributos. `Shift + F6` abrirá la tabla de atributos filtrada a las entidades seleccionadas y `Ctrl + F6` abrirá la tabla de atributos filtrada a las entidades visibles.

Esto abrirá una nueva ventana que muestra los atributos de las entidades de la capa (*figure_attributes_table*). Según la configuración en *Configuración -> Opciones -> Menú de fuentes de datos*, la tabla de atributos se abrirá en una ventana acoplada o en una ventana normal. El número total de entidades en la capa y la cantidad de entidades actualmente seleccionadas / filtradas se muestran en el título de la tabla de atributos, así como si la capa está limitada espacialmente.

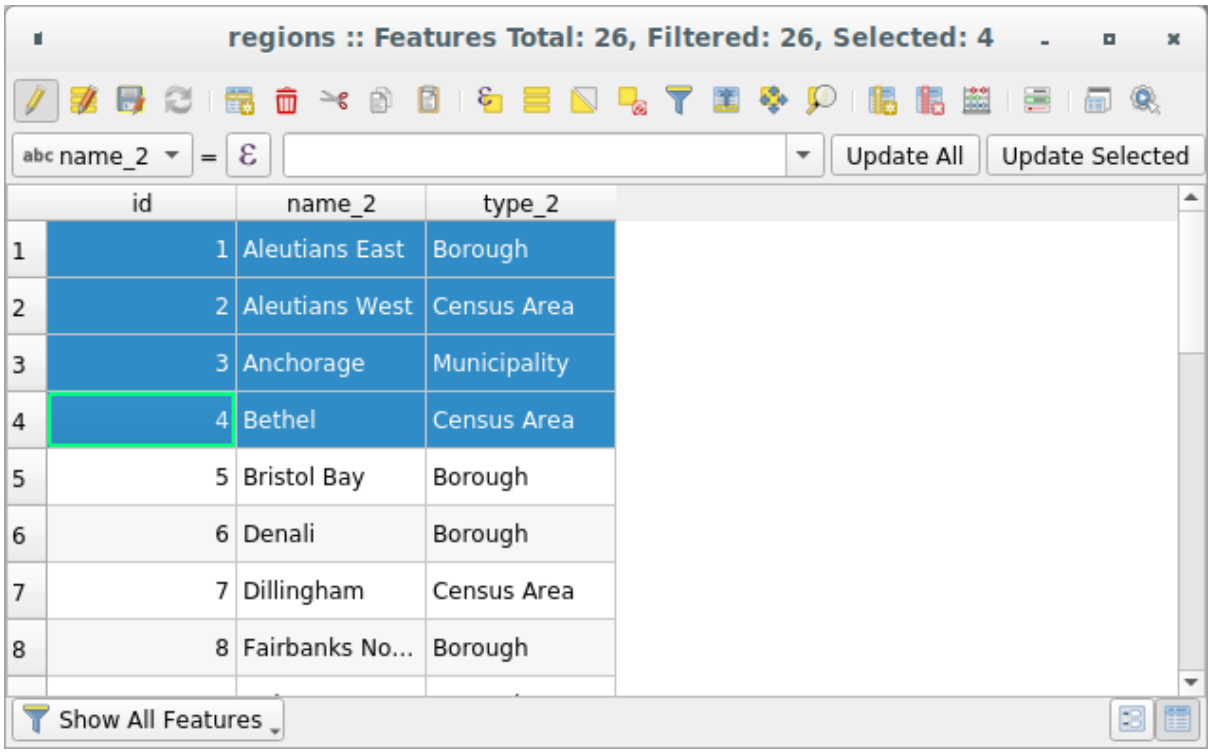


Figura 12.70: Tabla de atributos de la capa regiones

Los botones de la parte superior de la ventana de la tabla de atributos proporcionan la siguiente funcionalidad:

Tabla 12.1: Herramientas disponibles

Icono	Etiqueta	Propósito	Atajo de tecla
	Conmutar el modo de edición	Habilitar funciones de edición	`Ctrl+E`
	Conmutar el modo multiedición	Actualizar múltiples campos de muchos objetos	
	Guardar ediciones	Guardar las modificaciones actuales	
	Recargar la tabla		
	Añadir objeto espacial	Añadir nuevo objeto sin geometría	
	Borrar objetos seleccionados	Eliminar los objetos seleccionados de la capa	
	Cortar elementos seleccionados al portapapeles		`Ctrl+X`

continúe e

Tabla 12.1 – proviene de la página anterior

Icono	Etiqueta	Propósito	Atajo de tecla
	Copiar elementos seleccionados al portapapeles		`Ctrl+C`
	Pegar entidades desde el portapapeles	Insertar nuevas entidades a partir de unas copiadas	`Ctrl+V`
	Seleccionar objetos usando una expresión		
	Seleccionar todo	Seleccionar todos los objetos de la capa	`Ctrl+A`
	Invertir selección	Invertir la selección actual de la capa	`Ctrl+R`
	Deseleccionar todo	Deseleccionar todos los objetos de la capa actual	`Ctrl+Shift`
	Filtrar/Seleccionar entidades usando formulario		`Ctrl+F`
	Mover la selección arriba del todo	Mover filas seleccionadas a la parte superior de la tabla	
	Desplazar el mapa a las filas seleccionadas		`Ctrl+P`
	Ampliar mapa a las filas seleccionadas		`Ctrl+J`
	Nuevo campo	Añadir un nuevo campo a la fuente de datos	`Ctrl+W`
	Borrar campo	Borrar un campo de la fuente de datos	
	Organizar Columnas	Mostrar/ocultar campos de la tabla de atributos	
	Abrir la calculadora de campos	Actualizar campo para muchas entidades a la vez	`Ctrl+I`
	Formato condicional	Habilitar formato de tabla	
	Acoplar tabla de atributos	Permite acoplar / desacoplar la tabla de atributos	
	Actions	Enumera las acciones relacionadas con la capa.	

Atención

Dependiendo del formato de los datos y de la biblioteca GDAL incorporada con su versión de QGIS, es posible que algunas herramientas no estén disponibles.

Debajo de estos botones se encuentra la barra de Cálculo de campo rápido (habilitada solo en *modo de edición*), que permite aplicar cálculos rápidamente a todas o parte de las entidades de la capa. Esta barra usa lo mismo *expresiones* que el  Calculadora de campo (ver *Editar valores de atributo*).

Vista tabla vs Vista forma

QGIS ofrece dos modos de vista para manipular fácilmente los datos en la tabla de atributos:

- The  Table view, muestra valores de múltiples funciones en un modo tabular, cada fila representa una función y cada columna un campo. Un clic derecho en el encabezado de la columna le permite *configure the table display* mientras que un clic derecho en una celda proporciona *interaction with the feature*.

  La tabla de atributos admite el desplazamiento `Shift+Rueda del ratón` en el modo de vista de tabla para cambiar entre los movimientos de desplazamiento vertical y horizontal. Esto también se puede conseguir sustituyendo el ratón por el trackpad en macOS.

- The  Form view shows feature identifiers <maptips>`en un primer panel y muestra solo los atributos del identificador en el que se hizo clic en el segundo. Hay un menú desplegable en la parte superior del primer panel donde se puede especificar el «identificador» mediante un atributo (:guilabel:`Vista previa de columna) o una *Expresión*. El menú desplegable también incluye las últimas 10 expresiones para su reutilización. La vista de formulario utiliza la configuración de campos de capa (ver *Formulario de propiedades de campo*).

Puede navegar a través de los identificadores de características con las flechas en la parte inferior del primer panel. Los atributos de las características se actualizan en el segundo panel a medida que avanza. También es posible identificar o pasar a la función activa en el lienzo del mapa presionando cualquiera de los botones en la parte inferior:

-  Highlight current feature si está visible en el lienzo del mapa
-  Desplazarse automáticamente a la característica actual
-  Zoom a la función actual

Puede cambiar de un modo a otro haciendo click en el icono correspondiente en la parte inferior derecha del cuadro de diálogo.

También puede especificar el modo *Vista Preestablecida* al abrir la tabla de atributos en el menú *Configuración -> Opciones -> Fuentes de datos*. Puede ser «Recordar la última vista», «Vista de tabla» o «Vista de formulario».

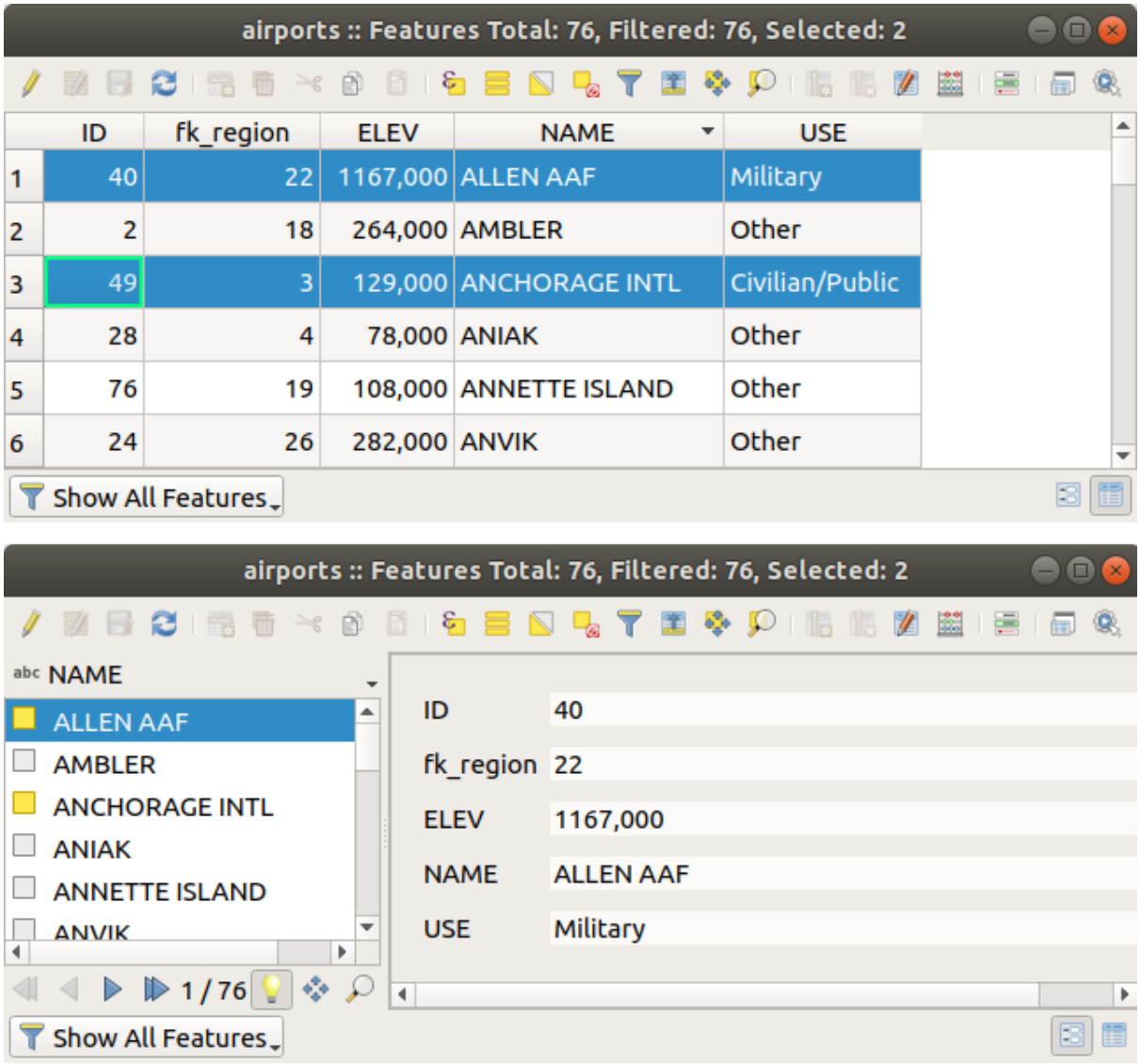


Figura 12.71: Tabla de atributos en la vista de tabla (arriba) vs vista de formulario (abajo)

Configurando las columnas

Haga clic con el botón derecho en el encabezado de una columna cuando esté en la vista de tabla para tener acceso a las herramientas que lo ayudan a controlar:

- the *column(s) size*
- the *column(s) visibility and order*
- the *sort order of the data*

Redimensionar el ancho de columnas

El ancho de las columnas puede especificarse via un clic derecho en la cabecera de la columna y seleccionar:

- *Ajustar anchura…* para ingresar el valor deseado. De forma predeterminada, el valor actual se muestra en el widget
- *Establecer todos los anchos de columna…* al mismo valor
- *Autodimensionado* para cambiar el tamaño del mejor ajuste a la columna.
- *Autodimensionar todas las columnas*

El tamaño de una columna también se puede cambiar arrastrando el límite a la derecha de su encabezado. El nuevo tamaño de la columna se mantiene para la capa y se restaura en la próxima apertura de la tabla de atributos.

En la *Configuración de fuentes de datos*, puede seleccionar la casilla de verificación *Ajustar automáticamente el tamaño de todas las columnas al abrir la tabla de atributos*, lo que hará que «Ajustar automáticamente el tamaño de todas las columnas» sea la vista predeterminada cada vez que se abran las tablas de atributos en QGIS.

Ocultar y organizar columnas y habilitar acciones

Al hacer clic con el botón derecho del ratón en el encabezado de una columna, puede seleccionar *Ocultar columna* en la tabla de atributos (en el modo «vista de tabla»). Para controles más avanzados, pulse el botón  Organizar columnas… de la barra de herramientas del cuadro de diálogo o seleccione *Organizar columnas…* en el menú contextual del encabezado de la columna.

En el nuevo cuadro de diálogo, puede:

- marque/desmarque las columnas que desea mostrar u ocultar: una columna oculta desaparecerá de todas las instancias del cuadro de diálogo de la tabla de atributos hasta que se restaure activamente. También es posible:
  - seleccione *Mostrar todo* para mostrar todos los campos (columnas) y acciones de la tabla.
  - seleccione *Ocultar todo* para ocultar todos los campos (columnas) y acciones de la tabla.
  - utilice el *Alternar selección* para invertir la visibilidad de la selección actual de columnas. Puede utilizar *combinación de teclas* para seleccionar varias columnas.
- elementos de arrastrar y soltar para reordenar las columnas en la tabla de atributos. Tenga en cuenta que este cambio es para la representación de la tabla y no altera el orden de los campos en la fuente de datos de la capa.
- agregue una nueva columna virtual *Actions* que muestre en cada fila un cuadro desplegable o una lista de botones de acciones habilitadas. Consulte *Propiedades de acciones* para obtener más información sobre las acciones.

Ordenar filas

Las filas pueden ordenarse por cualquier columna, haciendo clic en la cabecera de la columna. Una pequeña flecha indica el orden de clasificación (hacia abajo significa valores descendentes desde la fila superior hacia abajo, hacia arriba significa valores ascendentes desde la fila superior hacia abajo). También puede optar por ordenar las filas con la opción *Ordenar…* del menú contextual de la cabecera de columna y escribir una expresión. Por ejemplo, para ordenar las filas utilizando varias columnas puede escribir `concat(col0, col1)`.

En la vista de formulario, el identificador de características se puede ordenar mediante la opción  *Ordenar por expresión de vista previa*.

Tenga en cuenta que la ordenación de las filas sólo afecta a la representación de la tabla y no altera el orden de los objetos espaciales en la fuente de datos de la capa.

Truco

** Clasificación basada en columnas de diferentes tipos **

Intentar ordenar una tabla de atributos basada en columnas de tipo cadena y numérico puede dar lugar a resultados inesperados debido a que la expresión `concat("USE", "ID")` devuelve valores de cadena (es decir, `'Borough105' < 'Borough6'`). Puede evitarlo utilizando, por ejemplo, `concat("USE", lpad("ID", 3, 0))` que devuelve `'Borough105' > 'Borough006'`.

Formateo de celdas de tabla usando condiciones

La configuración de formato condicional se puede usar para resaltar en la tabla de atributos las entidades en las que es posible que desee poner un enfoque particular, utilizando condiciones personalizadas en las entidades:

- geometría (por ejemplo, identificación de entidades de múltiples partes, áreas pequeñas o en una extensión de mapa definida …);
- o valor de campo (por ejemplo, comparando valores con un umbral, identificando celdas vacías, duplicados, …).

Puede activar el panel de formato condicional pulsando el botón  Formateo condicional situado en la parte superior derecha de la ventana de atributos en la vista tabla (no se activa en la vista formulario).

El nuevo panel permite al usuario agregar nuevas reglas para formatear la representación de  *Field* o  *Full row*. Agregar nueva regla abre un formulario para definir:

- el nombre de la regla;
- una condición que utiliza cualquiera de las funciones *constructor de expresiones*;
- el formato: se puede elegir de una lista de formatos predefinidos o crearse en función de propiedades como:
  - colores de fondo y texto;
  - uso de icono;
  - negrita, cursiva, subrayado o tachado;
  - tipo de letra.

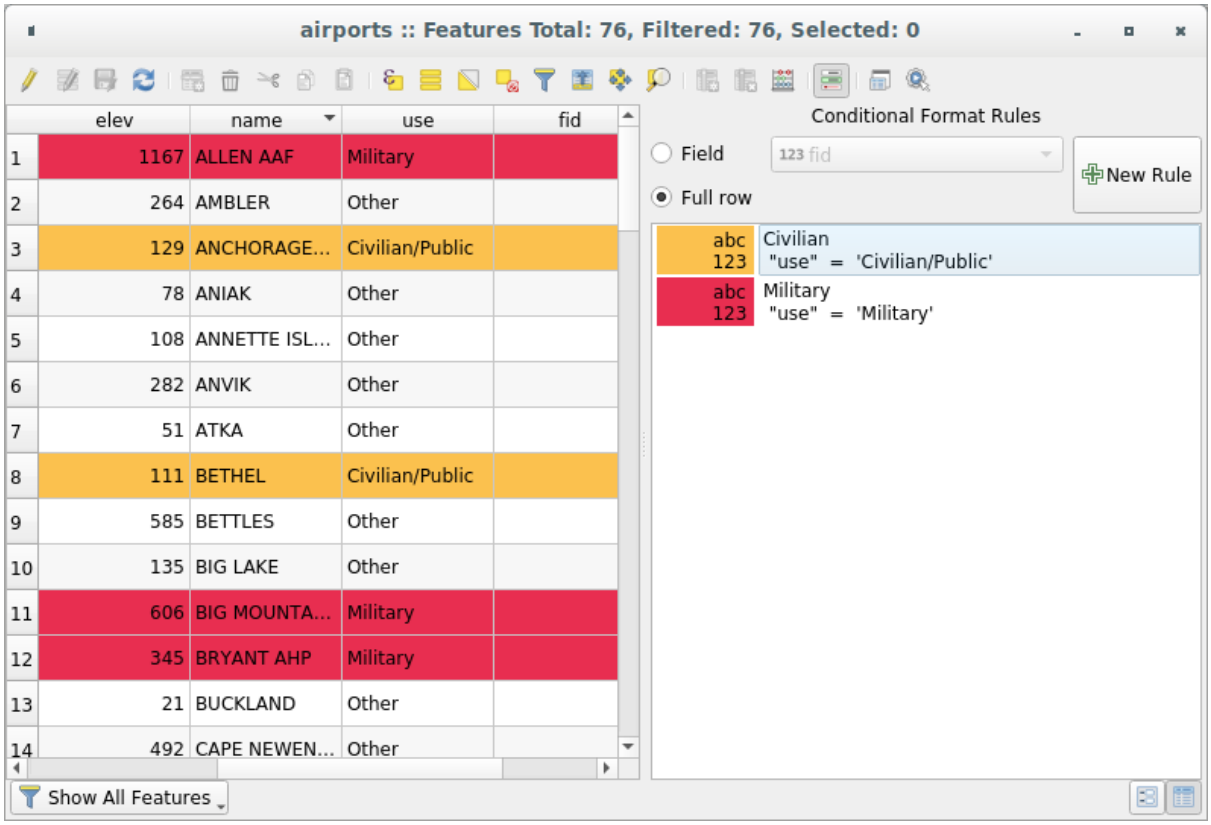


Figura 12.72: Formateo Condicional de una tabla de atributos

12.2.3 Interactuar con entidades en una tabla de atributos

Seleccionando objetos espaciales

En la vista de tabla, cada fila de la tabla de atributos muestra los atributos de una entidad única en la capa. Al seleccionar una fila, se selecciona la entidad e igualmente, al seleccionar una entidad en el lienzo del mapa (en el caso de una capa con geometría habilitada) se selecciona la fila en la tabla de atributos. Si se cambia el conjunto de entidades seleccionadas en el lienzo del mapa (o tabla de atributos), la selección también se actualiza en la tabla de atributos (o lienzo del mapa) en consecuencia.

Los registros se pueden seleccionar haciendo clic en el número de registros del lado izquierdo del registro. **Múltiples registros** se pueden marcar al mantener la tecla `Ctrl`. Una **selección continua** se puede hacer al mantener la tecla `Shift` y haciendo clic en varios encabezados de registros del lado izquierdo del registro. Todos los registros entre la posición actual del cursos y el registro donde se hace clic son seleccionados. Mover la posición del cursos en la tabla de atributos, al hacer clic en una celda de la tabla, no cambia el registro seleccionado. Cambiar la selección en el lienzo principal no mueve la posición del cursos en la tabla de atributos.

En la vista de formulario de la tabla de atributos, las características se identifican por defecto en el panel izquierdo por el valor de su campo mostrado (ver *Propiedades a mostrar*). Este identificador se puede reemplazar usando la lista desplegable en la parte superior del panel, ya sea seleccionando un campo existente o usando una expresión personalizada. También puede optar por ordenar la lista de funciones en el menú desplegable.

Haga click en un valor en el panel de la izquierda para mostrar los atributos de la entidad en el de la derecha. Para seleccionar una entidad, debe hacer click dentro del símbolo cuadrado a la izquierda del identificador. Por defecto, el símbolo se vuelve amarillo. Al igual que en la vista de tabla, puede realizar la selección de múltiples funciones utilizando las combinaciones de teclado expuestas anteriormente.

Además de seleccionar objetos espaciales con el ratón, puede realizar una selección automática basada en el atributo de la característica utilizando las herramientas disponibles en la barra de herramientas de la tabla de atributos, como

(consulte la sección *Selección automática* y siguientes para obtener más información y casos de uso):

-  *Seleccionar por expresión…*
-  *Seleccionar entidades por valor…*
-  *Deseleccionar todas las entidades de la capa*
-  *Seleccionar todas las entidades*
-  *Invertir selección de entidades.*

También es posible *seleccionar objetos espaciales mediante formularios*.

Filtrando entidades

Una vez que haya seleccionado entidades en la tabla de atributos, es posible que desee mostrar solo estos registros en la tabla. Esto se puede hacer fácilmente usando el elemento *Mostrar entidades seleccionadas* de la lista desplegable en la parte inferior izquierda del cuadro de diálogo de la tabla de atributos. Esta lista ofrece los siguientes filtros:

-  *Mostrar todas las características*
-  *Show Selected Features* - same as using :guilabel:``Mostrar características seleccionadas` - igual que usar *Abrir tabla de atributos (características seleccionadas)* desde el menú *Capa* o la *Barra de herramientas de atributos* o presionando `Shift+ F6`
-  *Mostrar características visibles en el mapa* - lo mismo que usar *Abrir tabla de atributos (Características visibles)* desde el menú *Capa* o la *Barra de herramientas de atributos* o presionando `Ctrl+F6`
-  *Mostrar objetos espaciales con Restricciones Fallidas* - las prestaciones se filtrarán para mostrar sólo las que tengan *restricciones* fallidas. Dependiendo de si la restricción incumplida es dura o blanda, los valores de los campos que fallan se muestran en celdas de color naranja oscuro o claro, respectivamente.
-  *Mostrar características editadas y nuevas* - igual que usar *Abrir tabla de atributos (características editadas y nuevas)* desde el menú *Capa* o la *Barra de herramientas de atributos*
- *Fieltro de campo* - permite al usuario filtrar según el valor de un campo: elija una columna de una lista, escriba o seleccione un valor y presione `Enter` para filtrar. Luego, solo las entidades que coinciden con `num_field = value` o `string_field como la expresión '%value%'` se muestran en la tabla de atributos. Puede marcar **|casilla|** *Sensible a mayúsculas y minúsculas* para ser menos permisivo con cadenas.
-  *Filtro avanzado (Expresión)* - Abre el cuadro de diálogo del constructor de expresiones. En él, puede crear *expresiones complejas* para hacer coincidir las filas de la tabla. Por ejemplo, puede filtrar la tabla utilizando más de un campo. Cuando se aplique, la expresión del filtro aparecerá en la parte inferior del formulario.
-  *Expresiones de filtro guardadas* ►: un atajo a *expresiones guardadas* utilizado frecuentemente para filtrar su tabla de atributos.

También es posible *filtrar entidades usando formularios*.

Nota

El filtrado de registros fuera de la tabla de atributos no filtra entidades fuera de la capa; simplemente se ocultan momentáneamente de la tabla y se puede acceder a ellos desde el lienzo del mapa o quitando el filtro. Para los filtros que ocultan entidades de la capa, use *Constructor de consultas*.

> **Truco**
>
> ** Actualizar el filtrado de la fuente de datos con ** `` Mostrar entidades visibles en el mapa``
>
> Cuando, por razones de rendimiento, las entidades que se muestran en la tabla de atributos están limitadas espacialmente a la extensión del lienzo en su apertura (ver *Opciones de origen de datos* para obtener instrucciones), seleccionando *Mostrar entidades visibles en el mapa* en un La nueva extensión de lienzo actualiza la restricción espacial.

Almacenamiento de expresiones de filtro

Las expresiones que utiliza para el filtrado de tablas de atributos se pueden guardar para futuras llamadas. Al utilizar las entradas *Filtrar Campos* o *Filtro avanzado (expresión)*, la expresión utilizada se muestra en un widget de texto en la parte inferior del cuadro de diálogo de la tabla de atributos. Pulse el botón Save expression with text as name al lado del cuadro para guardar la expresión en el proyecto. Pulsando el menú desplegable junto al botón permite guardar la expresión con un nombre personalizado (*Guardar expresión como…*). Una vez que se muestra una expresión guardada, se activa y su menú desplegable le permite *Editar la expresión* y el nombre si lo hay, o *Eliminar la expresión almacenada*.

Las expresiones de filtro guardadas se guardan en el proyecto y están disponibles a través del menú *Expresiones de filtro almacenadas* de la tabla de atributos. Son diferentes a los *user expressions*, compartida por todos los proyectos del perfil de usuario activo.

Filtrar y seleccionar entidades mediante formularios

Haciendo clic en Filtrar / Seleccionar entidades usando el formulario o presionando :kbd:` Ctrl + F` hará que el cuadro de diálogo de la tabla de atributos cambie a la vista del formulario y reemplace cada widget con su variante de búsqueda.

A partir de este punto, la funcionalidad de esta herramienta es similar a la descrita en *Seleccionar objetos por valor*, donde se pueden encontrar descripciones de todos los operadores y modos de selección.

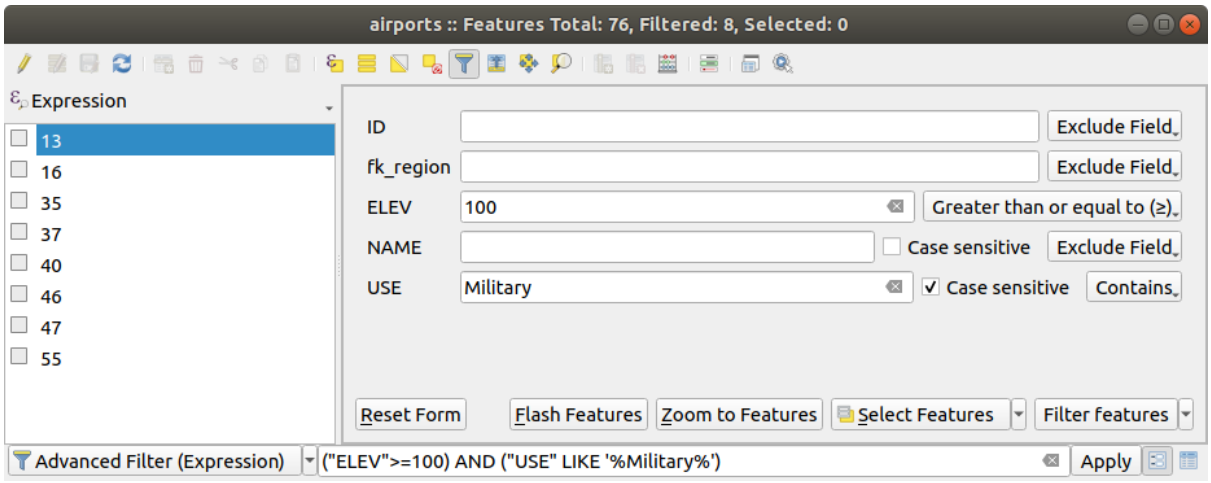


Figura 12.73: Tabla de atributos filtrada por el formulario de filtro

Al seleccionar / filtrar entidades de la tabla de atributos, hay un botón *Filtrar entidades* que permite definir y refinar filtros. Su uso activa la opción *Filtro avanzado (Expresión)* y muestra la expresión de filtro correspondiente en un widget de texto editable en la parte inferior del formulario.

Si ya hay entidades filtradas, puede refinar el filtro usando la lista desplegable junto al botón *Filtrar entidades*. Las opciones son:

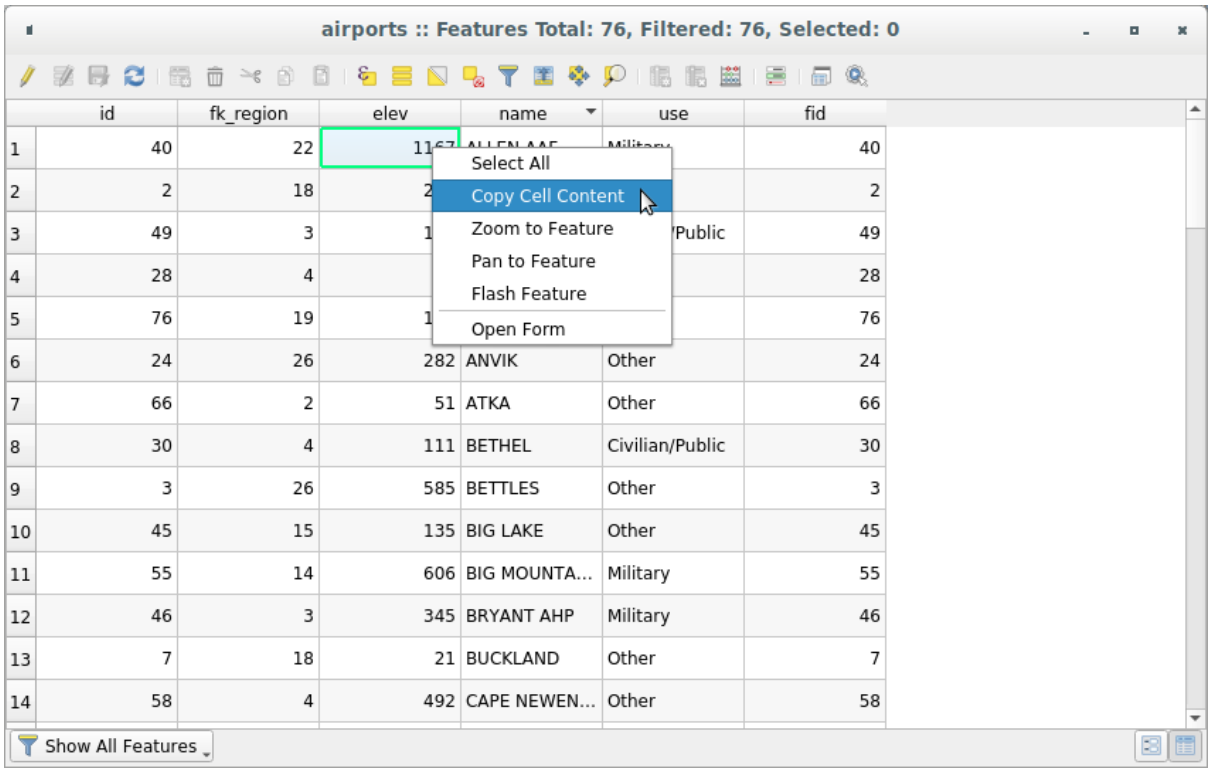
- *Filtrar dentro de («AND»)*
- *Extender filtro («OR»)*

Para borrar el filtro, selecciona la opción *Mostrar todas las entidades* del menú desplegable inferior izquierdo, o borre la expresión y haz click en :guilabel:` Aplicar` o presiona `Enter`.

Más acciones sobre objetos espaciales

Los usuarios tienen varias posibilidades para manipular objetos espaciales en una tabla de atributos. Haga clic con el botón derecho del ratón en una celda y podrá:

- *Seleccionar todas* (`Ctrl+A`) las entidades;
- Copiar el contenido de una celda en el portapapeles con *Copiar contenido de celda*;
- *Zoom a entidad* sin tener que seleccionarlo de antemano;
- *Navegar a entidad* sin tener que seleccionarla de antemano;
- *Destacar entidad*, para resaltarla en el lienzo de mapa;
- *Abrir formulario*: cambia la tabla de atributos a la vista de formulario con un enfoque en la entidad en la que se hizo click.
- Mostrar una lista *de acciones*, previamente activado en la pestaña *Propiedades de capa* ► *Acciones*.



	id	fk_region	elev	name	use	fid
1	40	22	1167	ALLEN AAF	Military	40
2	2	18	2			2
3	49	3	1		Public	49
4	28	4				28
5	76	19	1			76
6	24	26	282	ANVIK	Other	24
7	66	2	51	ATKA	Other	66
8	30	4	111	BETHEL	Civilian/Public	30
9	3	26	585	BETTLES	Other	3
10	45	15	135	BIG LAKE	Other	45
11	55	14	606	BIG MOUNTA...	Military	55
12	46	3	345	BRYANT AHP	Military	46
13	7	18	21	BUCKLAND	Other	7
14	58	4	492	CAPE NEWEN...	Other	58

Figura 12.74: Botón de copiar contenido de celda

Si desea utilizar datos de atributos en programas externos (como Excel, LibreOffice o una aplicación web personalizada), seleccione una o varias filas y utilice el botón Copiar filas seleccionadas al portapapeles o pulse `Ctrl+C`. Además, en el menú *Configuración* ► *Opciones* ► *Fuentes de datos*, puede definir el formato en el que pegar con la opción *Copiar objetos espaciales como*. Más detalles en *Ajustes de Fuentes de Datos*.

12.2.4 Editar valores de atributo

Para modificar los datos de una tabla de atributos, primero debe activar la edición de la capa. Pulse el botón  Conmutar edición. Dependiendo del tipo de geometría de la capa y del estado del portapapeles, se habilitan algunas herramientas más en la barra de herramientas superior de la tabla de atributos.

La edición de los valores de los atributos puede hacerse de la siguiente manera:

- escribiendo el nuevo valor directamente en la celda, ya sea que la tabla de atributos esté en la vista de tabla o formulario. Por tanto, los cambios se realizan celda por celda, entidad por entidad;
- usando la *calculadora de campos*: actualiza de una vez un campo que ya puede existir o que se creará, pero para múltiples entidades. Se puede utilizar para crear campos virtuales;
- usando el campo rápido *barra de cálculo*: igual que el anterior pero solo para el campo existente;
- o usando el modo *multiedición* : actualiza de una vez múltiples campos para múltiples entidades.

Putting the layer into edit mode will also allow you to  Paste features from clipboard (`Ctrl+V`)  Cut selected rows to clipboard (`Ctrl+X`) or  Delete selected features. More details at *Cortar, copiar, y pegar objetos espaciales*.

Usando la Calculadora de Campos

El botón  Calculadora de campos en la tabla de atributos permite hacer cálculos sobre la base de valores de atributos existentes o funciones definidas, por ejemplo, para calcular la longitud o el área de entidades geométricas. Los resultados se pueden utilizar para actualizar un campo existente o escribir en un nuevo campo (que puede ser uno *virtual*).

La calculadora de campo está disponible en cualquier capa que admita la edición. Al hacer clic en el icono de la calculadora de campo, se abre el cuadro de diálogo (ver Figura 12.75). Si la capa no está en modo de edición, se muestra una advertencia y el uso de la calculadora de campo hará que la capa se ponga en modo de edición antes de realizar el cálculo.

Basado en el diálogo *Constructor de expresiones*, el cuadro de diálogo calculadora de campos ofrece una interfaz completa para definir una expresión y aplicarla a un campo existente o de nueva creación. Para utilizar el cuadro de diálogo de la calculadora de campos, debe seleccionar si desea:

1. aplicar cálculo en toda la capa o solo en entidades seleccionadas
2. crear un nuevo campo para el cálculo o actualizar uno existente.

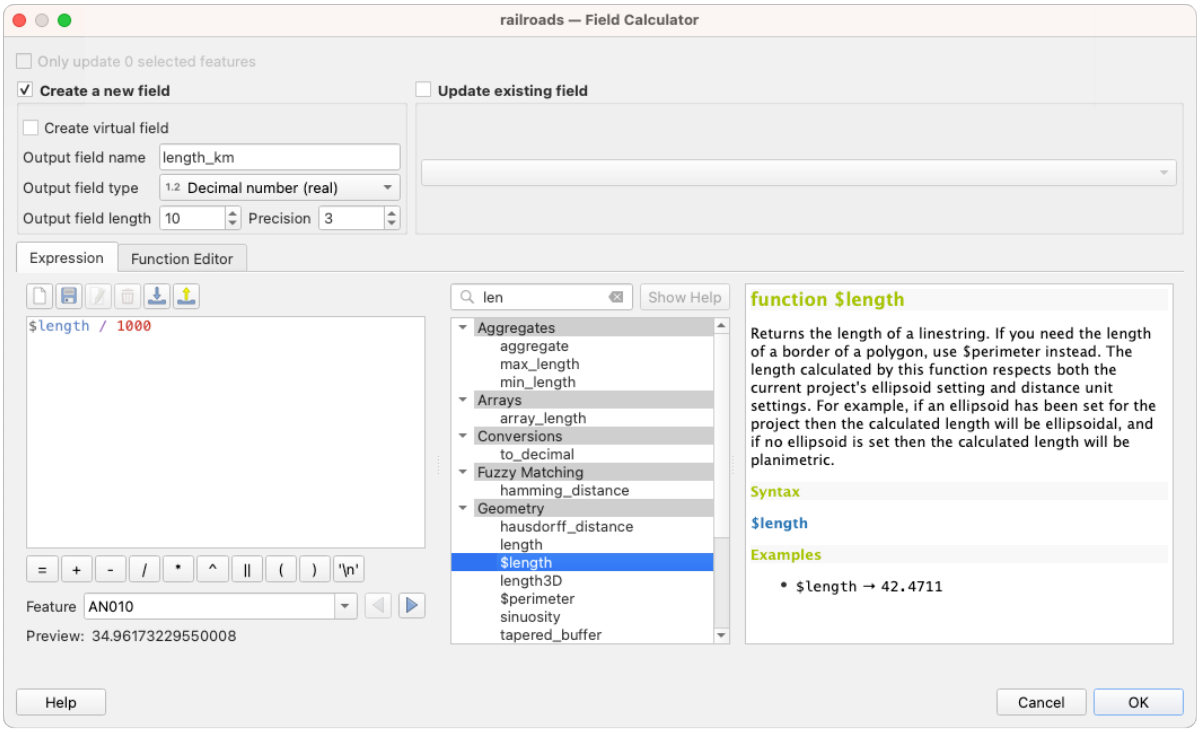


Figura 12.75: Calculadora de Campos

Si optas por agregar un nuevo campo, debe ingresar un nombre de campo, un tipo de campo (entero, real, fecha o cadena) y, si es necesario, la longitud total del campo y la precisión del campo. Por ejemplo, si elige una longitud de campo de 10 y una precisión de campo de 3, significa que tiene 7 dígitos antes del punto y 3 dígitos para la parte decimal.

Un breve ejemplo ilustra cómo funciona la calculadora de campo cuando se usa la pestaña *Expresión*. Queremos calcular la longitud en km de la capa `ferrocarriles` del conjunto de datos de muestra de QGIS:

1. Carga el archivo de forma `railroads.shp` en QGIS y pulsa Abrir tabla de atributos.
2. Click en conmutar el modo edición y abre el diálogo Calculadora de campos.
3. Selecciona la casilla de verificación *Crear un campo nuevo* para guardar los cálculos en un nuevo campo.
4. Establece *Nombre del campo de salida* como `length_km`
5. Selecciona `Número decimal (real)` como *Tipo de campo de salida*
6. Ajusta la *longitud del campo de salida* a `10` y la *Precisión* a `3`
7. Haga doble clic en `$length` en el grupo *Geometría* para añadir la longitud de la geometría en el cuadro de expresión de la calculadora de campo (empezará a ver una vista previa de la salida, de hasta 60 caracteres, debajo del cuadro de expresión actualizándose en tiempo real a medida que se monta la expresión).
8. Completa la expresión escribiendo `/ 1000` en el cuadro de expresión de la calculadora de campo y click en *Aceptar*.
9. Puede ahora encontrar un nuevo campo *length_km* en la tabla de atributos.

Creando un campo Virtual

Un campo virtual es un campo basado en una expresión calculada sobre la marcha, lo que significa que su valor se actualiza automáticamente en cuanto cambia un parámetro subyacente. La expresión se aplica a todas las prestaciones de la capa y se establece una vez; ya no es necesario recalcular el campo cada vez que cambian los valores subyacentes. Por ejemplo, puede que desee utilizar un campo virtual si necesita que el área se evalúe a medida que digitaliza características o para calcular automáticamente una duración entre fechas que pueden cambiar (por ejemplo, utilizando la función `now()`).

La creación de un campo virtual se realiza a través del diálogo *Calculadora de campos* y sigue el *mismo procedimiento* como para los campos regulares. Simplemente recuerde marcar la opción *Crear campo virtual* y utilizar un tipo de campo compatible con los datos que generaría su expresión.

La edición de un campo virtual se realiza a través de la pestaña *Campos* del diálogo de propiedades de la capa (véase menu_campos_vectoriales). La expresión que define el campo se expone en la columna *Comentario*, y al pulsar el botón **|expresión|** situado junto a ella se abre una ventana del editor de expresiones para su actualización.

Nota

Uso de Campos Virtuales

- Un campo sólo puede ser virtual en el momento de su creación.
- Los campos virtuales no son permanentes en los atributos de capa, lo que significa que solo se guardan y están disponibles en el archivo de proyecto que se han creado.

Usando la Barra de cálculo de campo rápido

Si bien la calculadora de campo siempre está disponible, la barra de cálculo de campo rápido en la parte superior de la tabla de atributos solo es visible si la capa está en modo de edición. Gracias al motor de expresión, ofrece un acceso más rápido para editar un campo ya existente:

1. Seleccione el campo a actualizar en la lista desplegable.
2. Llene el cuadro de texto con un valor, una expresión que escriba o cree directamente usando el botón expresión.
3. Haga click en el botón *Actualizar todo*, *Actualizar seleccionado* o *Actualizar filtrado* según sus necesidades.

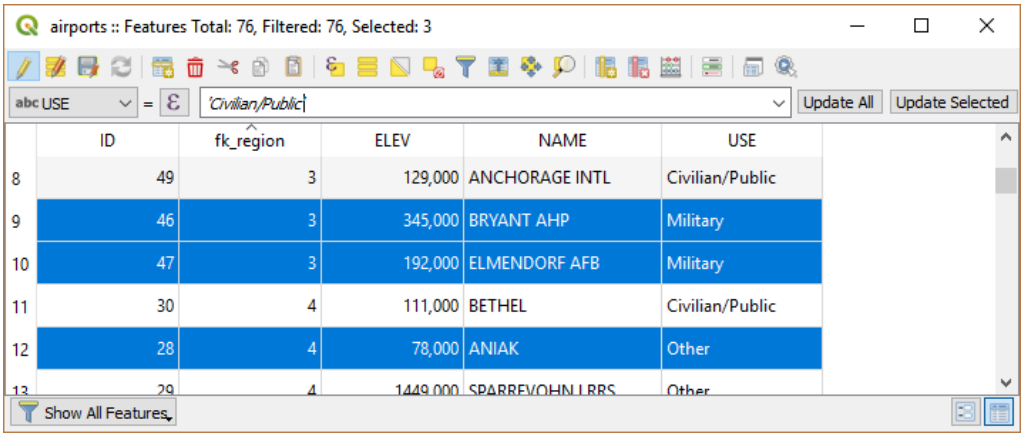


Figura 12.76: Barra de cálculo de campo rápido

Editando múltiples campos

A diferencia de las herramientas anteriores, el modo de edición múltiple permite editar simultáneamente varios atributos de diferentes entidades. Cuando se cambia la capa para editar, se puede acceder a las capacidades de edición múltiple:

- usando el botón  Conmutar modo multiedición desde la barra de herramientas dentro del cuadro de diálogo de la tabla de atributos;
- o seleccionando el menú *Editar* ►  *Modificar atributos de entidades seleccionadas*.

Nota

A diferencia de la herramienta de la tabla de atributos, presionando el botón *Editar* ► *Modificar Atributos de Entidades seleccionadas* le proporciona un diálogo modal para completar los cambios de atributos. Por lo tanto, se requiere la selección de entidades antes de la ejecución.

Con el objetivo de editar múltiples campos a la vez:

1. Selecciona las entidades que quieras editar.
2. Desde la barra de herramientas de la tabla de atributos, click en el botón  . Esto conmutará el diálogo a su vista de formulario. La selección de entidades también se puede realizar en este paso.
3. En el lado derecho de la tabla de atributos, se muestran los campos (y valores) de las entidades seleccionadas. Aparecen nuevos widgets junto a cada campo que permiten mostrar el estado actual de edición múltiple:
   -  El campo contiene diferentes valores para las entidades seleccionadas. Se muestra vacío y cada entidad mantendrá su valor original. Puede restablecer el valor del campo desde la lista desplegable del widget.
   -  Todas las entidades seleccionadas tienen el mismo valor para este campo y se mantendrá el valor que se muestra en el formulario.
   -  El campo ha sido editado y el valor ingresado se aplicará a todas las entidades seleccionadas. Aparece un mensaje en la parte superior del cuadro de diálogo que lo invita a aplicar o restablecer su modificación.

   Hacer click en cualquiera de estos widgets le permite establecer el valor actual del campo o restablecer el valor original, lo que significa que puede revertir los cambios campo por campo.

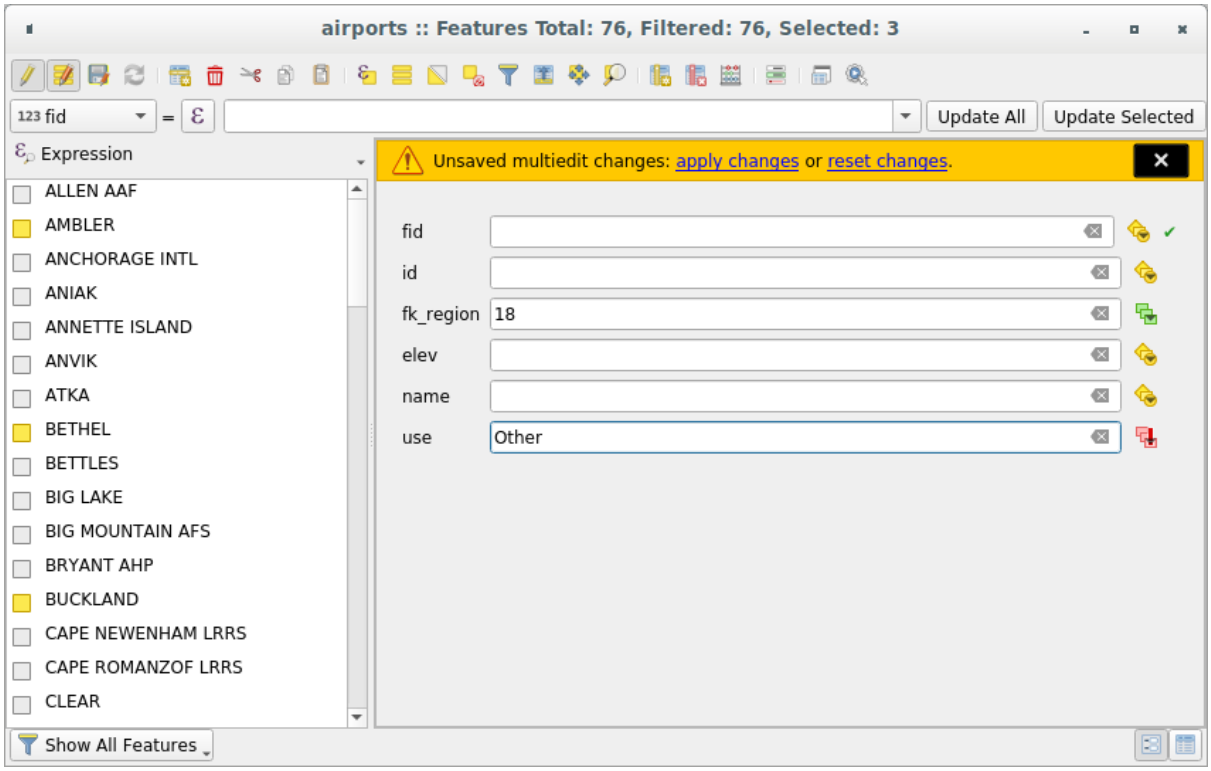


Figura 12.77: Editando campos de múltiples entidades

4. Haz cambios en los campos que quieras.
5. Click en **Aplicar cambios** en el texto del mensaje superior o cualquier otra función en el panel izquierdo.

Los cambios se aplicarán a ** todas las entidades seleccionadas **. Si no se selecciona ninguna función, toda la tabla se actualiza con sus cambios. Las modificaciones se realizan como un solo comando de edición. Así que presionando Deshacer revertirá los cambios de atributo para todas las entidades seleccionadas a la vez.

Nota

El modo de edición múltiple solo está disponible para formularios autogenerados y de arrastrar y soltar (ver *Personalizando un formulario para sus datos*); no es compatible con formularios de interfaz de usuario personalizados.

12.2.5 Exploración de los atributos de los objetos espaciales mediante la herramienta de Identificación

La herramienta *Identificar prestaciones* puede utilizarse para mostrar todos los atributos de un objeto espacial en el lienzo del mapa. Es una forma rápida de ver y verificar todos los datos sin tener que buscarlos en la tabla de atributos.

Para utilizar la herramienta *Identificar objetos espaciales* para capas vectoriales, siga estos pasos:

1. Seleccione la capa vectorial en el panel Capas.
2. Haga clic en la herramienta *Identificar objetos espaciales* de la barra de herramientas o pulse `Ctrl+Mayús+I`.
3. Haga clic en un objeto espacial en la vista del mapa.

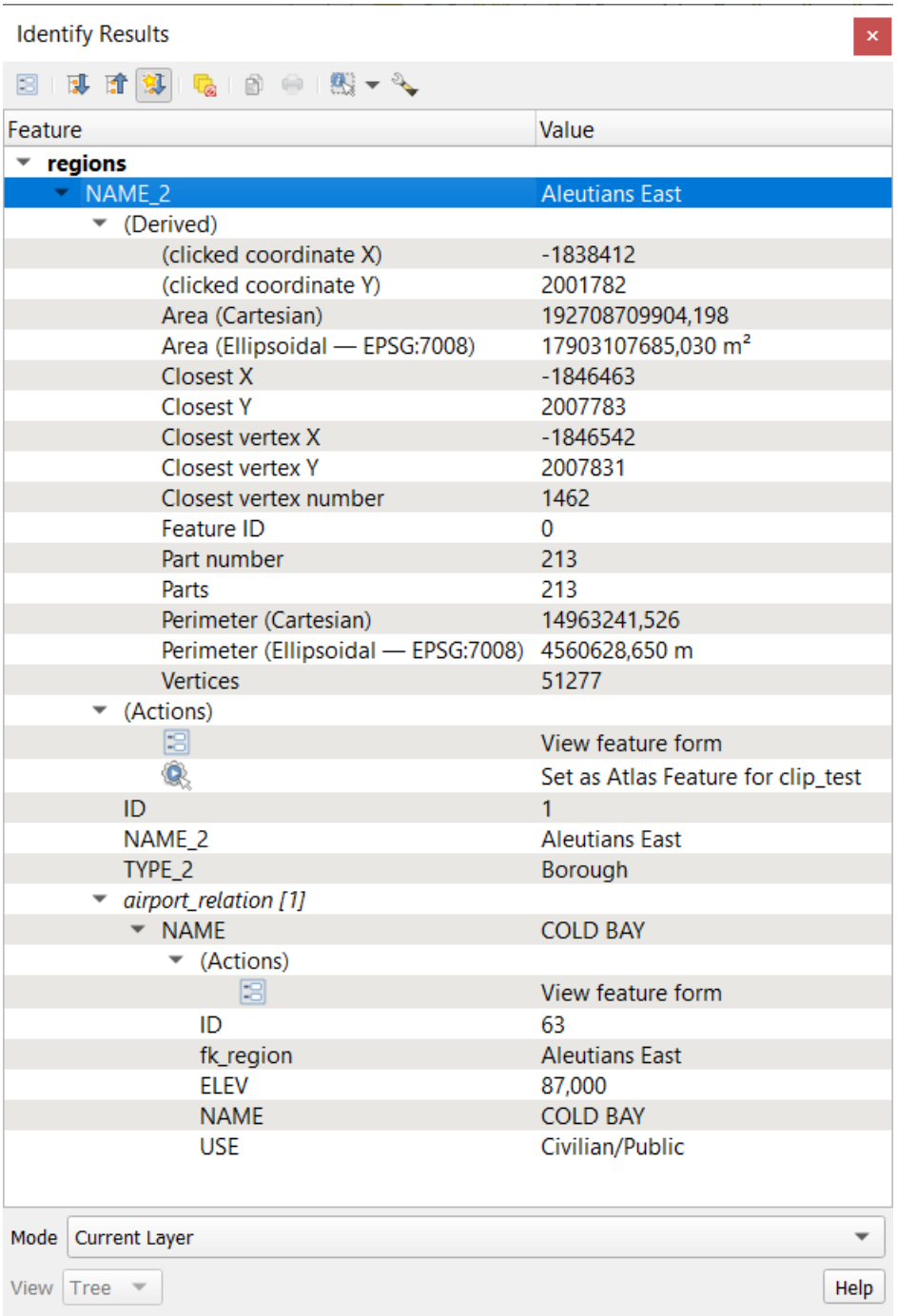


Figura 12.78: Identify Results dialog

The *Identify results* panel will display different features information in a tree view. There are two columns in the panel: *Feature* and *Value*. The first item is the name of the layer and its children are its identified feature(s). Each feature is identified by the *display name* field along with its value. Other information about the feature follows:

- Sección **derivada** - son las informaciones calculadas o derivadas de otras informaciones de la capa. Por ejemplo, el área de un polígono o la longitud de una línea. Información general que se puede encontrar en esta sección:
  - Depending on the geometry type, the cartesian measurements of length, perimeter or area in the layer's CRS units. For 3D line vectors, the cartesian line length is available.
  - Dependiendo del tipo de geometría y si se ha establecido un elipsoide en el diálogo *Propiedades del proyecto* (*General* ► *Medidas*), valores elipsoidales de longitud, perímetro o área utilizando las unidades especificadas.

  - El recuento de piezas de geometría en el objeto espacial y el número de la pieza en la que se ha hecho clic.
  - Número de vértices del objeto espacial.
  - Coordinate information, using the project properties *Coordinates display* settings:
    * `X` and `Y` coordinate values of the clicked point
    * the number of the closest vertex to the clicked point
    * `X` and `Y` coordinate values of the closest vertex (and `Z`/`M` if applicable)
    * if you click on a curved segment, the radius of that section is also displayed.
    * if both the vector layer and the project have vertical datums set and they differ, the `Z` value will be displayed for both datums.

- **Actions**: Actions can be added to the identify feature windows. The action is run by clicking on the action label. By default, only one action is added, namely `View feature form` for editing. You can define more actions in the layer's properties dialog (see *Propiedades de acciones*).
- **Atributos de datos**: esta es la lista de campos de atributos y valores para la entidad en la que se ha hecho clic.

  Nota

  Links in the feature's attributes are clickable from the *Identify Results* panel and will open in your default web browser.

- When a vector layer has defined *relations*, the *Identify Results* panel can display both **referenced** and **referencing** related features. To view these relations, ensure that the *Show relations* option is enabled in the *Identify Settings*. Available information for each related feature include:
  - the name of the relation and the linked layer
  - the entry in the referenced or referencing field, identifying the feature
  - the activated actions
  - the list of attribute fields and values

  You can expand related features to explore connected records through multiple levels, including many-to-many (n:m) and polymorphic relations. Only nodes you explicitly expand are loaded, preventing excessive nesting.

  To focus on a specific related record, right-click it and choose *Identify Feature*. This re-centers the identify results on the selected feature, starting a new identification tree from it and effectively limiting the visible nesting depth. Features already shown in ancestor nodes are automatically omitted to avoid duplicates or circular relations.

12.3 Editando

QGIS has various capabilities for editing OGR, SpatiaLite, PostgreSQL, MS SQL Server and Oracle Spatial vector layers and tables. They can be of 2D or 3D geometry type.

Nota

El procedimiento para edición de capas GRASS es diferente - vea la sección *Digitalizar y editar una capa vectorial GRASS* para más detalles.

Atención

Ediciones simultáneas

QGIS no rastrea si alguien más está editando el mismo objeto espacial al mismo tiempo que tú. La última persona en guardar las ediciones gana.

Truco

Validación de ediciones

La validación continua se puede activar por capas en el:menuselection:*Propiedades de capa –> Pestaña Digitalización. Más en :ref:`digitizingmenu*.

12.3.1 Configuración de la tolerancia de ajuste y el radio de búsqueda

En el *Configuración ► Opciones… ► Menú de digitalización, QGIS proporciona una serie de parámetros para configurar el comportamiento predeterminado de las herramientas de edición. Mas información en:ref:`digitizing_options*.

Para una edición óptima y precisa de geometrías de capas vectoriales, debemos establecer un valor adecuado de tolerancia de ajuste y radio de búsqueda para los vértices de las entidades. El grupo guilabel:*Snapping* proporciona opciones relacionadas, a saber, el manejo de la tolerancia de ajuste y el radio de búsqueda.

- Tolerancia de ajuste: cuando agrega un nuevo vértice o mueve uno existente, la tolerancia de ajuste es la distancia que usa QGIS para buscar el vértice o segmento más cercano al que está tratando de conectarse. Si no está dentro de la tolerancia de ajuste, QGIS dejará el vértice donde suelta el botón del mouse, en lugar de ajustarlo a un vértice o segmento existente.

  La configuración de tolerancia afecta a todas las herramientas que funcionan con ajuste y se aplica de forma predeterminada a nuevas capas y proyectos. Sin embargo, se puede anular a nivel de capa (see *Opciones de ajuste y digitalización*).

- Radio de búsqueda: *Radio de búsqueda para ediciones de vértices* es la distancia que utiliza QGIS para `buscar` el vértice a seleccionar al hacer clic en el mapa. Si no está dentro del radio de búsqueda, QGIS no encontrará ni seleccionará ningún vértice para editarlo.

La tolerancia de autoensamblado y el radio de búsqueda se establecen en `unidades de mapa` o `píxeles`. Es posible que deba experimentar para hacerlo bien. Si especifica una tolerancia demasiado grande, QGIS puede ajustarse al vértice incorrecto, especialmente si hay una gran cantidad de vértices en las proximidades. Cuanto menor sea el radio de búsqueda, más difícil será dar con lo que quieres mover.

12.3.2 Opciones de ajuste y digitalización

ajustes de rotura y digitalización globales (modo de ajuste, valor de tolerancia y unidades…) pueden ser anulados en el proyecto desde el menú *Proyecto ► Opciones de Ajuste…*. En *Opciones de Ajuste y Digitalización*, usted también puede configurar algunas otras propiedades capas de ajuste, límite de escala, topología…) La *Barra de Herramientas de Ajuste* le da acceso a la mayoría de estas funciones.

De forma predeterminada, el ajuste está deshabilitado en un proyecto hasta que presione el botón  Enable snapping o presione `S`. El modo de ajuste, el valor de tolerancia y las unidades también se pueden configurar en esta barra de herramientas.

Propiedades de ajuste

Hay tres opciones para seleccionar la(s) capas a las que ajustar:

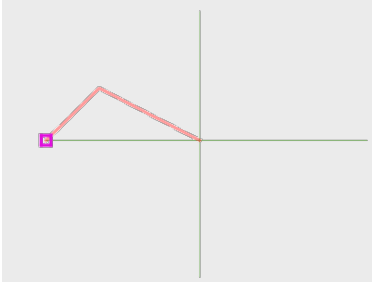
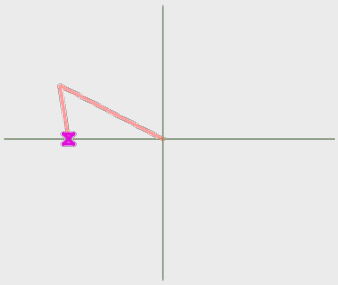
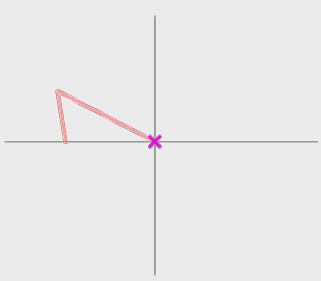
- *Todas las capas*: configuración rápida para todas las capas visibles en el proyecto para que el puntero se ajuste a todos los vértices y/o segmentos. En la mayoría de los casos, es suficiente usar este modo de ajuste, pero tenga cuidado al usarlo para proyectos con muchas capas vectoriales, ya que puede afectar el rendimiento.
- *capa actual*: solo se utiliza la capa activa, una forma conveniente de garantizar la coherencia topológica dentro de la capa que se está editando.
- *Configuración avanzada*: le permite habilitar y ajustar el modo de ajuste, la tolerancia y las unidades, las superposiciones y las escalas de ajuste en una capa (ver Figura 12.79). Si necesita editar una capa y ajustar sus vértices a otra, asegúrese de que la capa de destino esté marcada y aumente la tolerancia de ajuste a un valor más alto. El ajuste no ocurrirá en una capa que no esté seleccionada en el cuadro de diálogo de opciones de ajuste.

Al mover o crear vértices, puedes optar por los siguientes modos de ajuste:

- *Vertex*
- *Segmento*: se ajusta a lo largo de una línea o del perímetro de un polígono. Si la edición topológica está activada, se añade un nuevo vértice en el punto de ajuste.
- *Área*: garantiza que el punto de encaje se encuentra en cualquier parte del área de un polígono, no necesariamente en su límite.
- *Centroido*: se ajusta al centroide de la geometría de un objeto espacial. En el caso de una geometría multiparte, el punto de destino puede ser distinto de las partes existentes.
- *Medio de segmentos* en característica de línea o polígono
- *Puntos finales de línea*: se ajusta al primer o último vértice de cada parte de un objeto espacial de línea o polígono.

QGIS mostrará diferentes iconos de *pinzado* dependiendo del tipo de *pizado*.

Tabla 12.2: Iconos Snapping

		
Pinzado a un vértice: icono caja	Pinzando a un segmento: icono de reloj de arena	Pinzado a una intersección: icono en cruz

Tenga en cuenta que es posible cambiar el color de estos iconos en el:guilabel:*Digitalización* de parte de la configuración global.

Los valores de tolerancia se pueden establecer en las `unidades del mapa` del proyecto o en `píxeles`. La ventaja de elegir `píxeles` es que mantiene el ajuste constante en diferentes escalas de mapa. Normalmente, de 10 a 12 píxeles es un buen valor, pero depende del DPI de su pantalla. El uso de unidades de mapa permite relacionar la tolerancia con las distancias reales del suelo. Por ejemplo, si tiene una distancia mínima entre elementos, esta opción puede ser útil para asegurarse de no agregar vértices demasiado cerca unos de otros.

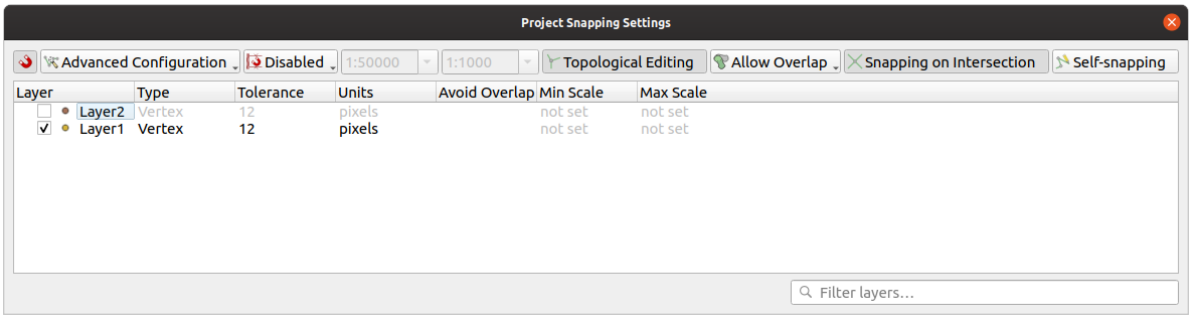


Figura 12.79: Opciones de autoensamblado (modo de Configuración Avanzada)

Nota

De forma predeterminada, solo las entidades visibles (las entidades cuyo estilo se muestra, excepto las capas donde la simbología es «Sin símbolos») se pueden autoensamblar. Puede habilitar el autoensamblado de entidades invisibles marcando *Habilitar el autoensamblado en entidades invisibles* bajo la *Configuración -> Opciones -> Ficha Digitalización*.

Truco

Activar autoensamblado por defecto

Puede configurar el autoensamblado para que esté habilitado de forma predeterminada en todos los proyectos nuevos en la pestaña *Configuración -> Opciones -> Digitalización*. También puede establecer el modo de autoensamblado predeterminado, el valor de tolerancia y las unidades, que llenarán el cuadro de diálogo :guilabel:`Opciones de autoensamblado".

Habilitar autoensamblado en intersecciones

Otra opción disponible es usar *autoensamblar en intersecciones*, que le permite ajustar a las intersecciones de geometría de las capas habilitadas para autoensamblar, incluso si no hay vértices en las intersecciones.

Limitar pinzado a un rango de escala

En algunos casos, el chasquido puede volverse muy lento. A menudo, esto se debe a la cantidad de entidades en algunas capas que requieren un índice elevado para calcular y mantener. Existen algunos parámetros para habilitar el ajuste solo cuando la vista del mapa está dentro de un rango de escala relevante. Esto permite realizar únicamente el costoso cálculo del índice relacionado con el ajuste a una escala en la que el dibujo es relevante.

EL límite de escala para pinzar se configura en *Proyecto* ► *Opciones de pinzado…*. Limitar pinzado a escala está solo disponible en el modo de *Configuración Avanzada*.

Para limitar el pinzado a un rango de escala tiene disponibles tres modos:

- *Deshabilitado*: El ajuste está habilitado sea cual sea la escala del mapa actual. Este es el modo por defecto.
- *Global*: El ajuste es limitado y solo se habilita cuando la escala actual del mapa se encuentra entre un valor mínimo global y un valor máximo global. Al seleccionar este modo, quedan disponibles dos widgets para configurar el rango de escalas en las que se habilita el ajuste.
- *Por capa*: El límite del rango de escala de pinzado se define para cada capa. Al seleccionar este modo, quedan disponibles dos columnas para configurar las escalas mínima y máxima para cada capa.

Tenga en cuenta que las escalas mínima y máxima siguen la convención de QGIS: la escala mínima es la escala más «alejada» mientras que la escala máxima es la más «acercada». Una escala mínima o máxima que se establece en «0» o «no establecida» se considera no limitante.

Auto-ajustable

La opción  Self-snapping le permite ajustar a la geometría que se está editando. Combinado con el *advanced digitizing panel*, esto proporciona una forma práctica de digitalizar nuevos bordes en relación con los bordes o vértices anteriores. El ajuste automático puede causar geometrías no válidas, use con precaución.

Ajuste en cuadrícula personalizada

Una distancia de ajuste también se puede personalizar en una base de capa en la pestaña *Digitalizando* del cuadro de diálogo de propiedades de la capa. Al establecer la distancia de *Precisión de geometría*, habilita una cuadrícula de puntos visible cuando el lienzo del mapa está en una escala coherente para su visualización. A continuación, se puede realizar el ajuste en los puntos de la cuadrícula: una geometría añadida o modificada tendrá todos sus vértices ajustados automáticamente al nodo más cercano de la cuadrícula. Mas información en *Propiedades de Digitalizado*.

12.3.3 Edición topológica

Además de estas opciones de autoensamblado, el cuadro de diálogo *Opciones de autoensamblado…* (*Proyecto* -> *Opciones de autoensamblado*) y la barra de herramientas *Autoensamblado* le permiten habilitar/deshabilitar algunas otras funcionalidades topológicas.

Habilitar edición topológica

El botón  Edición topológica ayuda al editar y mantener entidades con límites comunes. Con esta opción habilitada, QGIS “detecta” límites compartidos. Cuando mueva vértices / segmentos comunes, QGIS también los moverá en las geometrías de las entidades vecinas.

La edición topológica funciona con entidades de diferentes capas, siempre que las capas estén visibles y en modo de edición.

En capas con valores Z o M, la edición topológica interpolará el valor Z o M del vértice basándose en el valor de la arista utilizada para la conexión.

Control superpuesto

La superposición le impide dibujar nuevas entidades que se superponen a las existentes en la capa seleccionada, lo que acelera la digitalización de los polígonos adyacentes. Se puede controlar con la herramienta de superposición. Hay tres modos disponibles:

1.  *Permitir superposición* (predeterminado)
2.  *Evitar superposición en capa activa*: evita cualquier superposición con otras entidades de la capa que se está editando. Digitalice las nuevas geometrías para que se superpongan con sus vecinas y QGIS cortará las parte(s) superpuestas de las nuevas geometrías y las ajustará al límite de las características existentes. La ventaja es que no tiene que digitalizar los vértices comunes en el límite.
3.  *Seguir configuración avanzada*: permite que la configuración superpuesta se establezca por capas en el modo de vista *Configuración avanzada*.

Nota

Si la nueva geometría está totalmente cubierta por las existentes, se borra y QGIS mostrará un mensaje de error.

Advertencia

Utilice con precaución la opción*Evitar superposición*

Dado que esta opción cortará nuevas geometrías superpuestas de cualquier capa de polígono, puede obtener geometrías inesperadas si olvida desmarcarla cuando ya no la necesite.

Rastreo Automático

Por lo general, al usar herramientas de captura de mapas (agregar entidad, agregar parte, agregar anillo, remodelar y dividir), debe hacer click en cada vértice de la entidad. Con el modo de rastreo automático, puede acelerar el proceso de digitalización, ya que ya no necesita colocar manualmente todos los vértices durante la digitalización:

1. Active la herramienta  Trazado`(en la barra de herramientas :guilabel:`Autoensamblado) pulsando el icono o presionando la tecla `T`.
2. *Ajustar a* un vértice o segmento de una entidad sobre la que desea trazar.
3. Mueva el ratón sobre otro vértice o segmento que le gustaría ajustar y, en lugar de la línea recta habitual, la banda elástica digitalizadora representa una ruta desde el último punto que encajó a la posición actual. La herramienta también funciona con geometrías curvas.

   QGIS en realidad usa la topología de entidades subyacentes para construir la ruta más corta entre los dos puntos. El rastreo requiere que se active el autoensamblado en capas rastreables para construir el camino. También debe ajustar a un vértice o segmento existente mientras digitaliza y asegurarse de que los dos nodos se puedan conectar topológicamente a través de los bordes de entidades existentes; de lo contrario, QGIS no puede conectarlos y, por lo tanto, traza una sola línea recta.
4. Haga click y QGIS coloca los vértices intermedios siguiendo la ruta mostrada.

Despliegue el icono  Habilitar Trazado y ajuste la opción *Offset* para digitalizar una ruta paralela a las entidades en lugar de to digitize a path parallel to the features instead de trazar a lo largo de ellas. Un valor positivo desplaza el nuevo dibujo al lado izquierdo de la dirección de trazado y un valor negativo hace lo contrario.

Nota

** Ajuste la escala del mapa o la configuración de ajuste para un trazado óptimo **

Si hay demasiadas entidades en la visualización del mapa, el rastreo se deshabilita para evitar una preparación de la estructura de rastreo potencialmente larga y una gran sobrecarga de memoria. Después de acercar o deshabilitar algunas capas, el rastreo se habilita nuevamente.

Nota

** No agrega puntos topológicos **

Esta herramienta no agrega puntos a las geometrías poligonales existentes incluso si *Edición topológica* está habilitada. Si la precisión de la geometría está activada en la capa editada, es posible que la geometría resultante no siga exactamente una geometría existente.

Truco

**Habilite o deshabilite rápidamente el rastreo automático presionando la tecla ** `T`

Al presionar la tecla `T`, el rastreo se puede habilitar/deshabilitar en cualquier momento (incluso mientras se digitaliza una entidad), por lo que es posible digitalizar partes de la entidad con rastreo habilitado y otras partes con rastreo deshabilitado. Las herramientas se comportan como de costumbre cuando el rastreo está deshabilitado.

Truco

Convertir trazado a geometrías curvas

Usando *Configuración* ► *Opciones* ► *Digitalizar* ► *Trazado* puede crear geometrías curvas mientras digitaliza. Ver *opciones de digitalización*.

12.3.4 Digitalizando una capa existente

De forma predeterminada, QGIS carga capas de solo lectura. Esta es una protección para evitar editar accidentalmente una capa si se produce un deslizamiento del ratón. Sin embargo, puede optar por editar cualquier capa siempre que el proveedor de datos lo admita (consulte *Explorando campos y formatos de datos*) y la fuente de datos subyacente se pueda escribir (es decir, sus archivos no son de solo lectura).

Truco

Restringir el permiso de edición en capas dentro de un proyecto

Desde la tabla *Proyecto* ► *Propiedades…* ► *Fuentes de Datos* ► *Capacidades de Capas*, puede elegir establecer cualquier capa como de solo lectura independientemente del permiso del proveedor. Esta puede ser una forma práctica, en un entorno de múltiples usuarios, para evitar que usuarios no autorizados editen capas por error (por ejemplo, Shapefile), por lo tanto, datos potencialmente corruptos. Tenga en cuenta que esta configuración solo se aplica dentro del proyecto actual.

En general, las herramientas para editar capas vectoriales se dividen en una barra de herramientas de digitalización y una barra de herramientas de digitalización avanzada, descritas en la sección *Digitalización avanzada*. Puede seleccionar y anular la selección de ambas en *Ver* ► *Barras de herramientas* ►.

Con las herramientas de digitalización básicas, puede realizar las siguientes funciones:

Tabla 12.3: Barra de herramientas de edición básica de capas vectoriales

Herramienta	Propósito	Herrar	Propósito
	Acceso para guardar, revertir o cancelar cambios en todas las capas o en las seleccionadas simultáneamente		Active o desactive el estado de edición de las capa(s) seleccionadas en función del estado de la capa activa
	Guardar ediciones en la capa activa		
	Digitalizar utilizando segmentos rectos		Digitalizar mediante líneas curvas
	Activar la digitalización a mano alzada		Digitalizar polígono de forma regular
	Añadir objeto espacial		Añadir objeto espacial: Añadir Punto
	Añadir objeto espacial: Añadir línea		Añadir objeto espacial: Añadir Polígono
	Herramienta de Vértices (Todas las capas)		Herramienta de Vértices (capa actual)
 *Mostrar Editor de Vértices*	Establece si el panel del editor de vértices debe abrirse automáticamente		Modificar los atributos de todos los objetos seleccionados de forma simultánea
	Eliminar entidades seleccionadas de la capa activa		Cortar entidades de la capa activa
	Copie las entidades seleccionadas de la capa activa		Pegar entidades en la capa activa
	Deshacer cambios en la capa activa		Rehacer cambios en la capa activa

Tenga en cuenta que mientras utiliza cualquiera de las herramientas de digitalización, aún puede *zoom o navegar* en el lienzo del mapa sin perder el foco en la herramienta.

Todas las sesiones de edición comienzan eligiendo la opción  Toggle editing que se encuentra en el menú contextual de una capa dada, desde el cuadro de diálogo de la tabla de atributos, la barra de herramientas de digitalización o el menú *Capa*.

Una vez que la capa está en modo de edición, los botones de herramientas adicionales en la barra de herramientas de edición estarán disponibles y los marcadores aparecerán en los vértices de todas las entidades a menos que *Mostrar marcadores solo para los objetos seleccionados* en la opción de menú *Configuración* ► *Opciones…* ► *Digitalización* esté marcada.

Truco

Guardar regularmente

Recuerde  Guardar cambios en capa regularmente. Esto también comprobará que su fuente de datos pueda aceptar todos los cambios.

Técnicas de edición de geometría

Cuando una herramienta de dibujo geométrico (principalmente las que añaden, dividen, remodelan características) está activada para una capa basada en líneas o polígonos, puede seleccionar la técnica para añadir nuevos vértices:

- La función  Digitalizar con Segmento: dibuja un segmento recto cuyos puntos inicial y final están definidos por clics izquierdos.
- La función  Digitalizar con Curva: dibuja una línea curva basada en tres nodos consecutivos definidos por clics izquierdos (inicio, punto a lo largo del arco, final). Si el tipo de geometría no admite curvas, se utilizan segmentos consecutivos más pequeños para aproximar la curvatura.
- La función  Digitalización de corrientes: dibuja líneas en modo a mano alzada, es decir, los nodos se añaden siguiendo el movimiento del cursor en el lienzo del mapa y una *Tolerancia de flujo*. La tolerancia de flujo define el espaciado entre vértices consecutivos. Actualmente, la única unidad admitida son los píxeles (`px`). En este modo sólo son necesarios el clic izquierdo inicial y el clic derecho final.
-  Digitalizar Formas: activa herramientas en la *Barra de Herramientas Digitalizar Formas* para dibujar un polígono de una forma regular.

La técnica seleccionada permanece mientras se cambia entre las herramientas de digitalización. Puede combinar cualquiera de los tres primeros métodos mientras dibuja la misma geometría.

Añadir objetos espaciales

Dependiendo del tipo de capa, puede usar los iconos  Añadir objeto espacial,  Añadir Punto,  Añadir Línea o  Añadir Polígono en la barra de herramientas para añadir nuevos objetos espaciales en la capa actual.

Para agregar una función sin geometría, haga clic en el botón  Add Record y puede ingresar atributos en el formulario de características que se abre.

Para crear entidades con las herramientas habilitadas espacialmente, primero digitaliza la geometría y luego ingresa sus atributos. Para digitalizar la geometría:

1. (Opcional ya que es el valor por defecto) Seleccione el método de dibujo de geometría  Digitalizar con Segmento
2. Haga clic izquierdo en el área del mapa para crear el primer punto de su nueva entidad. Para las características de puntos, esto debería ser suficiente y activar, si es necesario, el formulario de características para completar sus atributos.
3. Para geometrías de línea o polígono, siga haciendo clic izquierdo para cada punto adicional que desee capturar. Puedes confiar en la opción *snapping to features* , el *snap-to-grid* o el panel *advanced digitizing* para colocar con precisión cada vértice.

   Además de dibujar segmentos rectos entre los nodos que pulse uno a uno, se pueden trazar líneas y polígonos:

   - *trazado automáticamente*, acelera la digitalización. Esto creará líneas rectas consecutivas entre los vértices que usted coloque siguiendo las características existentes.
   - digitalizado a mano alzada, pulsando `R` o activando  Digitalización de flujos.
   - dibujada como curva, pulsando `Ctrl+Mayús+G` o activando **|digitalizarConCurva|** Digitalizar con Curva.

   Nota

   Mientras digitaliza geometrías lineales o poligonales, puede alternar entre los métodos de dibujo geométrico, lo que le permite crear características mezclando segmentos rectos, a mano alzada y partes curvas.

4. Pulse la tecla `Borrar` o `Retroceso` para revertir el último o los últimos nodos que haya añadido erróneamente.
5. Cuando haya terminado de agregar puntos, haga click derecho en cualquier parte del área del mapa para confirmar que ha terminado de ingresar la geometría de ese objeto espacial.

> **Truco**
>
> **Personaliza la banda elástica de digitalización**
>
> Mientras capturas el polígono, la banda elástica roja predeterminada puede ocultar las entidades subyacentes o los lugares en los que le gustaría capturar un punto. Esto se puede solucionar estableciendo una opacidad más baja (o canal alfa) para la goma elástica *Color de relleno* en el menú :menuselection:` Configuración -> Opciones -> Digitalización`. También puedes evitar el uso de la banda elástica marcando *No actualizar la banda elástica durante la edición de nodos*.

6. Para función de línea presionando `Shift` +clic derecho cerrará la línea automáticamente.
7. Aparecerá la ventana de atributos, permitiéndole ingresar la información para la nueva característica. Figura 12.80 muestra atributos de configuración para un nuevo río ficticio. Sin embargo, en el *digitalizando* menú en el menú *Configuración* ► *Opciones*, también puede:
   - *Suprimir las ventanas emergentes de atributos después de cada entidad creada* para evitar la apertura del formulario;
   - o *Reutilizar los últimos valores de atributo ingresados* para que los campos se llenen automáticamente al abrir el formulario y solo tener que escribir los valores cambiantes.

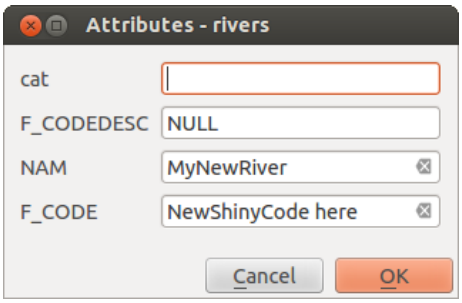


Figura 12.80: Entrar en el diálogo de valores de atributo después de digitalizar una nueva entidad vectorial

Herramienta Vértices

QGIS proporciona dos herramientas para interactuar con vértices de características vectoriales:

- Vertex Tool (Current Layer): solo entidades superpuestas en la capa activa (en *Capas* panel) se ven afectados
- Vertex Tool (All Layers): cualquier entidad superpuesta en todas las capas editables se ve afectada. Esto le permite editar características sin cambiar la capa activa o editar varias capas a la vez (por ejemplo, los límites del país y sus regiones)

Para cualquier capa vectorial editable, las herramientas de vértice brindan capacidades de manipulación de vértices de características similares a los programas CAD. Es posible seleccionar múltiples vértices a la vez y moverlos, agregarlos o eliminarlos por completo. Las herramientas de vértice también admiten la función de edición topológica. Son persistentes en la selección, por lo que cuando se realiza alguna operación, la selección permanece activa para esta función y herramienta.

> **Truco**
>
> **Dibujar una serie de nuevos vértices**
>
> La herramienta de vértices no admite *trazado automático* y está optimizada para editar vértices individuales y mover o eliminar una selección arbitraria de múltiples vértices. Pruebe a utilizar la herramienta  Reformar objetos espaciales cuando necesite sustituir o insertar una serie de vértices nuevos, especialmente si desea trazar otros objetos espaciales.

Es importante ajustar la propiedad *Configuración* ► *Opciones* ► *Digitalización* ► *Radio de búsqueda:*  a un número mayor que cero. De lo contrario, QGIS no podrá decir qué vértice se está editando y mostrará una advertencia.

> **Truco**
>
> **Marcadores de vértices**
>
> QGIS admite diferentes tipos de marcadores de vértice: "Círculo semitransparente", "Cruz" y "Ninguno". Para cambiar el estilo del marcador, seleccione *Opciones* del menú *Configuración*, haga clic en la pestaña *Digitalizando* y seleccione la entrada adecuada.

Operaciones básicas

Dada una capa en modo de edición, comience activando la herramienta de vértice. Aparecerán círculos rojos cuando se desplacen los vértices.

- **Selecting vertices**: Puede seleccionar vértices por:
  - Haciendo clic en ellas una a la vez sosteniendo la tecla `Shift`
  - Hacer clic y arrastrar un rectángulo que rodea los vértices de destino
  - Dibujar un polígono que rodee los vértices de destino: Mantener `Alt` y haga clic usando la herramienta de vértice para comenzar a digitalizar un polígono. Cada clic subsiguiente agrega un nuevo vértice al polígono elástico. `Backspace` or `Delete` elimina el último vértice de banda elástica agregado.:kbd:*Esc* ccancela el modo de selección de polígonos, al igual que retrocede/borra todos los vértices de la banda elástica. El clic derecho finaliza la digitalización del polígono y selecciona todos los vértices dentro del polígono elástico.

  Cuando se selecciona un vértice, su color cambia a azul. Para agregar más vértices a la selección actual, mantenga presionada la tecla `Shift` mientras continúa como se indicó anteriormente. Para eliminar vértices de la selección, mantenga presionado `Ctrl`.

  > **Truco**
  >
  > **Herramienta de vértice de límites de selección de entidades**
  >
  > Los vértices se pueden seleccionar a través de diferentes características (o capas). Si está buscando vértices de una característica específica en un lugar lleno de gente, primero seleccione esa característica. Luego dibuje el selector de rectángulo o polígono con la herramienta de vértice alrededor de los vértices: solo se seleccionan los vértices de la entidad seleccionada.
  >
  > Este también es el caso si muestra la función en el panel *vertex editor*.

- **Modo de selección de vértices por lotes**: El modo de selección por lotes se puede activar presionando `Shift + R`. Seleccione un primer nodo con un solo click y luego coloque el cursor **sin hacer click** en otro vértice. Esto seleccionará dinámicamente todos los nodos intermedios utilizando la ruta más corta (para polígonos).

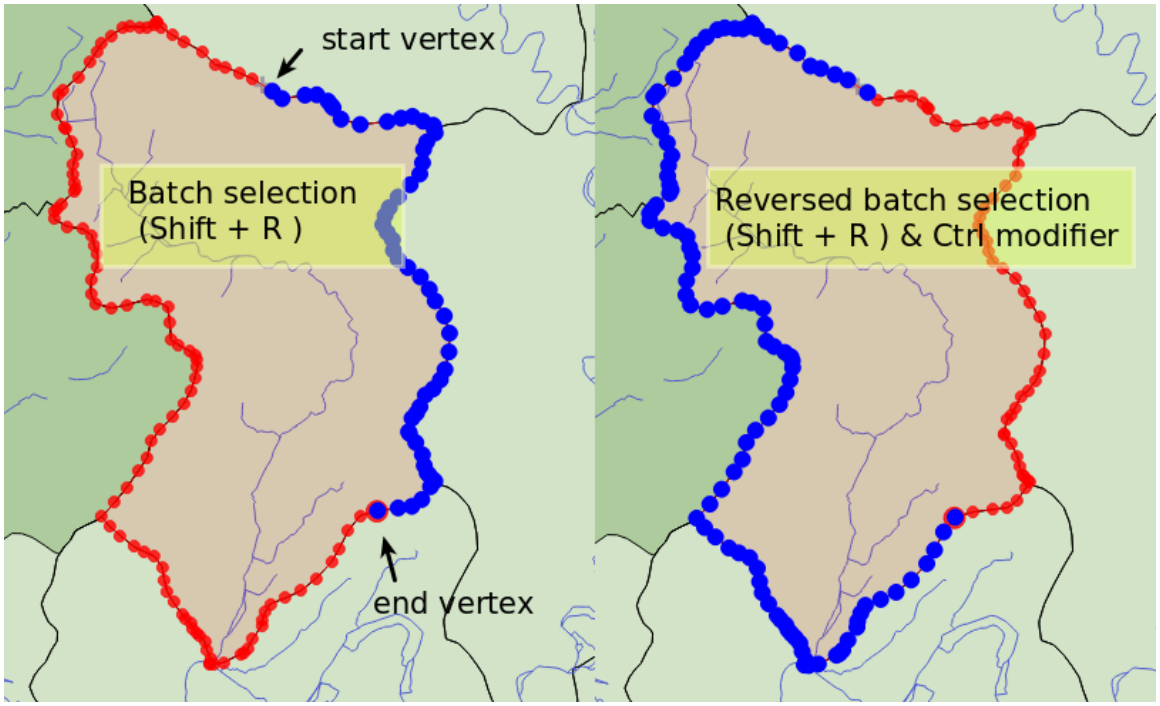


Figura 12.81: Selección de lotes de vértices usando `Shift+R`

Presiona `Ctrl` invertirá la selección, seleccionando la ruta más larga a lo largo del límite de la entidad. Terminando su selección de nodo con un segundo click, o presionando `Esc` saldrá del modo de selección por lotes.

- **Adding vertices**: Para agregar un vértice a una geometría de línea o polígono, mantenga presionada la tecla `Shift` y haga doble clic en el lugar del segmento.

  Al pasar el cursor sobre un segmento, aparece un nuevo nodo virtual en el centro. Haga clic en él, mueva el cursor a una nueva ubicación y haga clic nuevamente para agregar un nuevo vértice. Para las líneas, también se propone un nodo virtual en ambos extremos: haga clic en él, haga clics posteriores y termine con un clic derecho; esto permite extender fácilmente una línea existente.

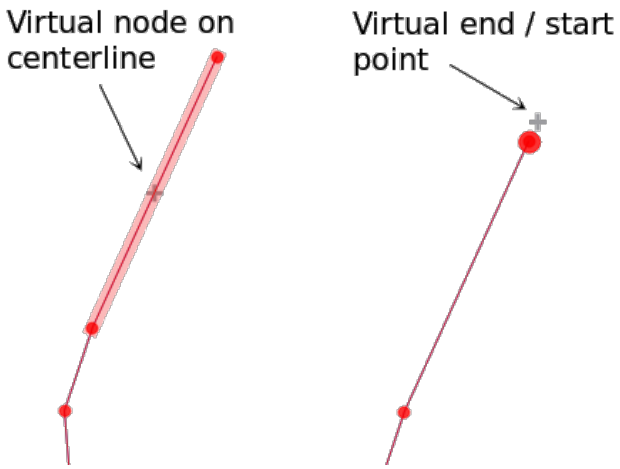


Figura 12.82: Nodos virtuales para la adición de vértices

- **Eliminando vértices**: Selecciona los vértices y haz click en la tecla `Delete`. Eliminar todos los vértices de una entidad genera, si es compatible con la fuente de datos, una entidad sin geometría. Tenga en cuenta que esto no elimina la función completa, solo la parte de geometría. Para eliminar una función completa, utilice la

herramienta Eliminar seleccionados.

- **Moving vertices**: Seleccione todos los vértices que desea mover, haga clic en un vértice o borde seleccionado y haga clic en la nueva ubicación deseada. Puede utilizar *snapping to feature capabilities* y las restricciones *Advanced Digitizing Panel* para la distancia, los ángulos y la ubicación exacta de X e Y antes del segundo clic. Todos los vértices seleccionados serán trasladados.

  Sin embargo, si la opción *snap-to-grid* está habilitada, los vértices seleccionados se ajustan a la intersección de cuadrícula más cercana a su posición traducida. Los vértices no seleccionados también se mueven a su intersección de cuadrícula más cercana. No hay una traducción sencilla.

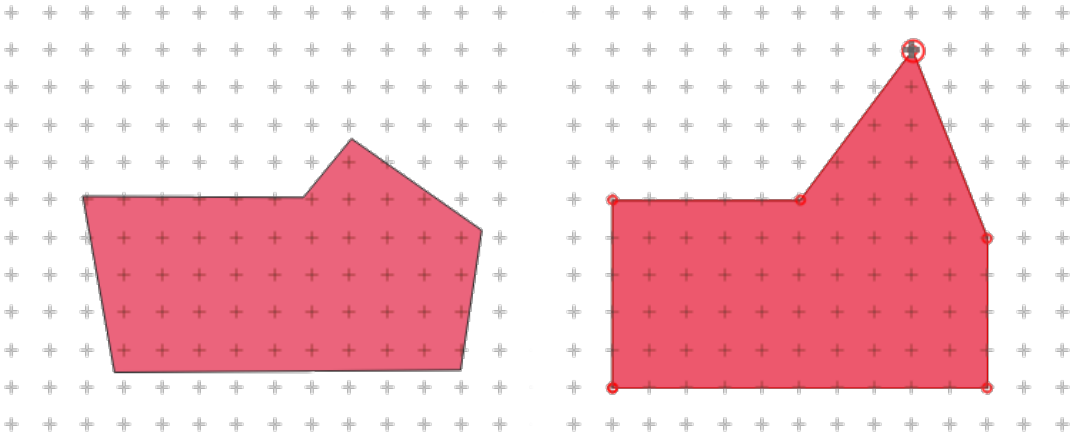


Figura 12.83: Mover el vértice superior ajusta todos los vértices a la cuadrícula

- **Convertir segmentos adyacentes a/desde curva**: Seleccione el vértice central del segmento que desea convertir, pulse la tecla de letra `O`. Si el vértice estaba en una curva, la curva se convierte en rectas. Si el vértice estaba entre dos rectas, éstas se convierten en una curva. Un primer o un último vértice de una recta no se pueden convertir en una curva de vértice central. La capa debe ser compatible con el tipo de geometría de curva.

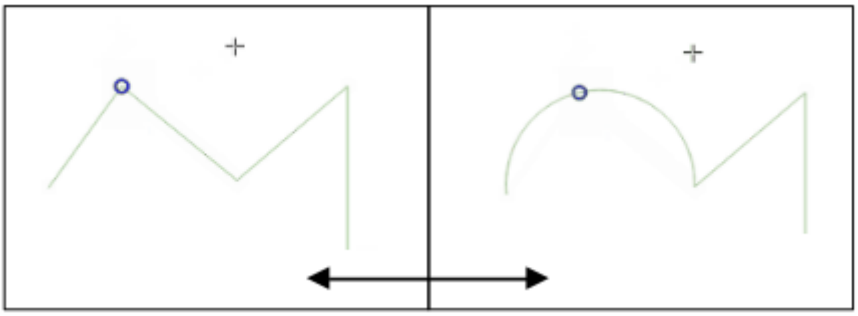


Figura 12.84: Cambiar de líneas curvas a rectas con `O` letra

Cada cambio realizado con la herramienta de vértice se almacena como una entrada independiente en el cuadro de diálogo *Undo*. Recuerde que todas las operaciones admiten la edición topológica cuando está activada. También se admite la proyección sobre la marcha.

El Panel de Editor de Vértices

Al activar una herramienta de vértices, también se abre el panel *Editor de vértices*. Al hacer clic con el botón derecho del ratón sobre un elemento, el panel se llena con la lista de todos los vértices del elemento con sus coordenadas *x*, *y* (*z*, *m* si procede) y *r* (para el radio, en caso de geometría circular). La característica también se convierte en exclusiva para la edición, lo que significa que se desactiva la edición de cualquier otra característica:

- Al seleccionar una fila en la tabla, se selecciona el vértice correspondiente en el lienzo del mapa, y viceversa.
- Al hacer clic o arrastrar sobre el lienzo del mapa sólo se seleccionarán o moverán los vértices y segmentos de esa característica
- Cambia una coordenada en la tabla y la posición del vértice se actualiza. Esta es una forma conveniente de editar la coordenada Z o el valor M en los vértices.
- También puede seleccionar varias filas y eliminarlas todas juntas.
- Sólo se pueden añadir nuevos vértices al elemento delimitado

Si no desea que el panel *Editor de vértices* se muestre inmediatamente cada vez que interactúe con las herramientas de vértices (y potencialmente oculte otros paneles o perturbe la colocación de los paneles), desactive la entrada *Auto-abrir tabla* en el menú ☰ Opciones en la parte superior del panel. También puede cerrar el panel. Para volver a abrir el panel, haga clic con el botón derecho del ratón sobre un panel o barra de herramientas y selecciónelo en la lista o marque la opción *Mostrar editor de vértices* de la *Barra de herramientas de digitalización*.

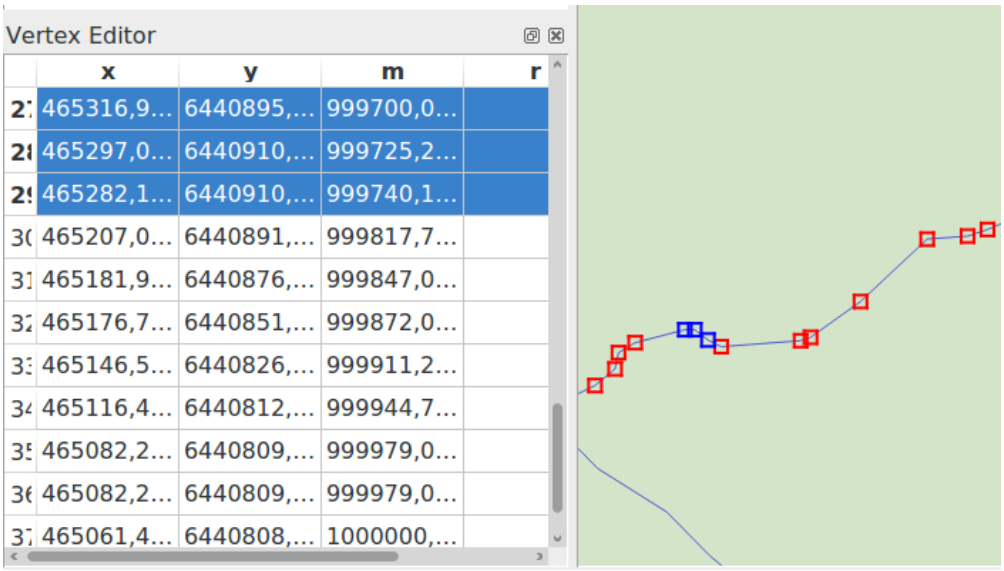


Figura 12.85: El panel de Editor de Vértices mostrando los nodos seleccionados

Reglas de asignación de la coordenada Z o del valor M

Digitalizar objetos espaciales vectoriales 3D u objetos espaciales con valor M no es tan diferente de (X,Y) capas 2D". Las herramientas y opciones descritas en este capítulo siguen estando disponibles y le ayudan a situar el vértice o punto en un entorno plano. Entonces puede que necesite manejar la asignación de la coordenada Z (o valor M):

- De modo predeterminado, QGIS asignará a los nuevos vértices el *Valor Z predeterminado* (respectivamente al *Valor M predeterminado*) establecido en la pestaña *Ajustes* ► *Opciones* ► *Digitalización*. Si el *Panel de Digitalización Avanzada* está en uso se toma el valor de su widget *z* (respectivamente *m*).
- Cuando se ajusta a un vértice, el vértice nuevo o movido toma el valor Z o M del ajustado.
- Cuando se ajusta a un segmento mientras la edición topológica está activada, el nuevo valor Z o M del vértice se interpola a lo largo del segmento.

- Si el widget *z* (respectivamente *m*) del *Panel Avanzado de Digitalización* está **|bloqueado|** bloqueado, entonces su valor se aplica al vértice, teniendo prioridad sobre cualquier valor Z o M de vértice o segmento ajustado.

Para editar los valores Z o M de una característica existente, usted puede usar el *Panel editor de vértice*. Para crear características con valores Z o M personalizados es posible que desee confiar en el *Panel de Digitalización Avanzada*.

Cortar, copiar, y pegar objetos espaciales

Las entidades seleccionadas se pueden cortar, copiar y pegar entre capas en el mismo proyecto QGIS, siempre que las capas de destino estén configuradas de antemano en  Conmutar edición.

> **Truco**
>
> **Transformar polígono a línea y viceversa usando copiar/pegar**
>
> Copie una entidad de línea y péguela en una capa de polígono: QGIS pega en la capa de destino un polígono cuyo límite corresponde a la geometría cerrada de la entidad de línea. Esta es una forma rápida de generar diferentes geometrías de los mismos datos.

Las funciones también se pueden pegar en aplicaciones externas como texto. Es decir, las entidades se representan en formato CSV, y los datos de geometría aparecen en formato OGC Well-Known Text (WKT). Las características de WKT y GeoJSON de fuera de QGIS también se pueden pegar en una capa dentro de QGIS.

¿Cuándo sería útil la función de copiar y pegar? Bueno, resulta que puedes editar más de una capa a la vez y copiar/pegar entidades entre capas. ¿Por qué querríamos hacer esto? Digamos que necesitamos trabajar un poco en una nueva capa, pero solo necesitamos uno o dos lagos, no los 5,000 en nuestra capa `big_lakes`. Podemos crear una nueva capa y usar copiar/pegar para colocar los lagos necesarios en ella.

Como un ejemplo, copiaremos algunos lagos a una nueva capa:

1. Cargar la capa desde donde desee copiar (capa fuente)
2. Cargar o crear la capa a la que desee copiar (capa destino)
3. Comenzar a editar la capa destino
4. Hacer la capa de fuente activa haciendo clic sobre ella en la leyenda
5. Use la herramienta  Seleccionar entidades por área o click simple para seleccionar la entidad(es) en la capa fuente
6. Click en la herramienta  Copiar Entidades
7. Hacer la capa de destino activa haciendo clic en la leyenda.
8. Click en la herramienta  Pegar entidades
9. Detener edición y guardar los cambios

¿Qué pasa si la capa de origen y destino tienen diferentes esquemas (los nombres de campo y tipo no son los mismos)? QGIS poblará los que coinciden e ignorará el resto. Si no son importantes los atributos que se copian a la capa de destino, no importa la forma de diseñar los campos y tipos de datos. Si desea asegurarse de que todo - el objeto espacial y sus atributos - se copia, asegúrese de que los esquemas coincidan.

When you paste features into another layer, QGIS checks the attribute values against the *constraints* defined in the destination layer (for example, `NOT NULL`, `Unique`, or expression constraints). If one or more pasted features contain invalid attribute values that do not meet these constraints, a dialog will appear, listing all affected features and fields. When pasting a single feature, you can decide whether to:

- Edit the invalid values directly in the dialog. The corrected values will be used when pasting.
- *Paste Anyway*: the feature is pasted as-is, even with invalid values.

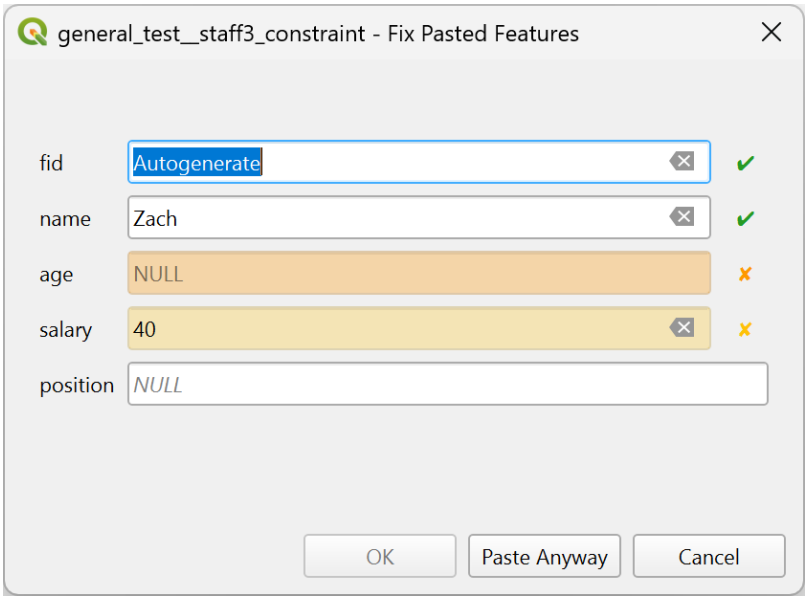


Figura 12.86: Dialog for handling invalid attribute values when pasting a single feature

When pasting multiple features, the dialog lists all invalid fields across all features, allowing bulk review and correction. You can decide per field or per feature whether to:

- Edit the invalid values directly in the dialog. The corrected values will be used when pasting.
- *Discard All*: no features pasted.
- *Discard All Invalid*: pasted only features with valid attribute values.
- *Paste All (Including Invalid)*: all features are pasted as-is, even with invalid values.
- *Skip*: skip the specific feature, allowing proceeding with the rest.

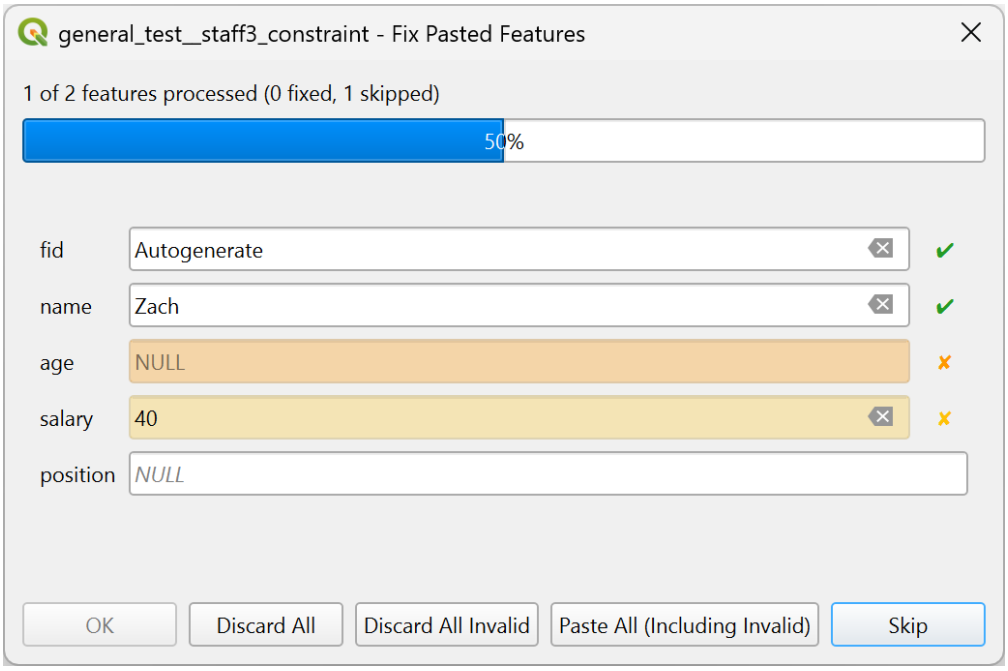


Figura 12.87: Dialog for handling invalid attribute values when pasting multiple features

Nota

Congruencia del pegado de objetos espaciales

Si su capa de origen y destino utilizan la misma proyección, entonces el pegado de objetos espaciales tendrá la geometría idéntica a la capa origen. Sin embargo, si el destino tiene una proyección diferente, entonces QGIS no puede garantizar que la geometría sea idéntica. Esto es simplemente porque hay pequeños errores de redondeo involucrados en la conversión entre las proyecciones.

Truco

Copiar atributos de texto en otro

Si ha creado una nueva columna en su tabla de atributos con el tipo «cadena» y desea pegar valores de otra columna de atributos que tenga una longitud mayor, la longitud del tamaño de la columna se ampliará en la misma medida. Esto se debe a que el controlador GDAL Shapefile sabe ampliar automáticamente los campos de cadena y de número entero para adaptarse dinámicamente a la longitud de los datos que se van a insertar.

Borrar objetos espaciales seleccionados

Si queremos eliminar una entidad completa (atributo y geometría), podemos hacerlo seleccionando primero la geometría usando la herramienta Seleccionar objetos espaciales por área o clic único. La selección también se puede hacer desde la tabla de atributos. Una vez que haya configurado la selección, presione la tecla `Delete` o :kbd:` Backspace` o use la herramienta Borrar lo seleccionado para eliminar los objetos. Se pueden eliminar varios objetos espaciales seleccionados a la vez.

La herramienta Cortar objetos espaciales en la barra de herramientas de digitalización también se puede utilizar para borrar objetos espaciales. Este efectivamente borra el objeto espacial pero también lo coloca en un «portapapeles espacial». Por lo tanto, cortamos el objeto espacial para borrar. Entonces podríamos utilizar la herramienta Pegar objetos espaciales para colocarlo de nuevo, nos da una capacidad de deshacer de un nivel. Cortar, copiar y pegar funcionan sobre los objetos espaciales seleccionados, lo que significa que podemos operar más de una a la vez.

Deshacer y rehacer

Las herramientas Deshacer y Rehacer le permiten deshacer o rehacer operaciones de edición de vectores. También hay un control acoplable, que muestra todas las operaciones en el historial de deshacer/rehacer (ver Figura 12.88). Este widget no se muestra de forma predeterminada; se puede mostrar haciendo clic derecho en la barra de herramientas y activando la casilla de verificación *Panel Deshacer/Rehacer*. Sin embargo, la función Deshacer/Rehacer está activa, incluso si el widget no se muestra.

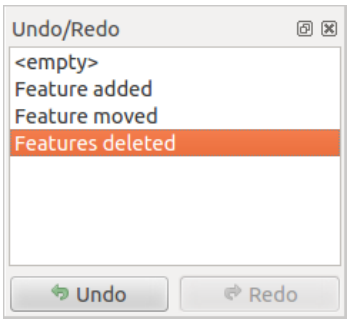


Figura 12.88: Rehacer y deshacer pasos digitalizados

Cuando se pulsa Deshacer o se presiona `Ctrl+Z` (or `Cmd+Z`), el estado de todas las entidades y atributos se revierte al estado anterior a la operación revertida. Los cambios que no sean las operaciones normales de edición de vectores (por ejemplo, cambios realizados por un complemento) pueden revertirse o no, dependiendo de cómo se realizaron los cambios.

Utilizar el widget de historial de deshacer/rehacer, sólo haga click para seleccionar la operación en la lista del histórico. Todos los objetos espaciales se revertirán al estado que tenían después de la operación de seleccionada.

Guardar capas editadas

Cuando una capa está en modo de edición, cualquier cambio permanece en la memoria de QGIS. Por lo tanto, no se confirman / guardan inmediatamente en la fuente de datos o el disco. Si deseas guardar las ediciones en la capa actual pero desea seguir editando sin salir del modo de edición, puede hacer click en el botón  Guardar cambios en capa.
Cuando desactivas el modo de edición con  Conmutar edición (o salir de QGIS para el caso), también se te preguntará si deseas guardar tus cambios o descartarlos.

Si los cambios no se pueden guardar (por ejemplo, disco lleno, o los atributos tienen valores que están fuera de rango), el estado de memoria de QGIS es preservado. Esto le permite ajustar sus ediciones e intentar de nuevo.

> **Truco**
>
> **Integridad de datos**
>
> Siempre es buena idea respaldar sus datos fuente antes de iniciar la edición. Mientras los autores de QGIS han hecho todo el esfuerzo para preservar la integridad de sus datos, nosotros no ofrecemos garantía en este sentido.

Guardando capas múltiples de una vez

Este objeto espacial permite la digitalización de múltiples capas. Elegir  *Guardar las capas seleccionadas* para guardar todos los cambios que se hicieron en múltiples capas. Se tiene la oportunidad para  *Revertir las capas seleccionadas*, así que la digitalización puede ser retirada de todas las capas seleccionadas. Si se desea detener la edición de las capas seleccionadas,  *Cancelar para la capa seleccionada(s)* es una forma fácil.

Las mismas funciones están disponibles para editar todas las capas del proyecto.

> **Truco**
>
> ** Utilice el grupo de transacciones para editar, guardar o revertir los cambios de varias capas a la vez **
>
> Cuando trabaje con capas de la misma base de datos PostGreSQL, active la opción *Crear grupos de transacción automáticamente cuando sea posible* en *Proyecto* ► *Propiedades…* ► *Fuentes de datos* para sincronizar su comportamiento (entrar o salir del modo de edición, guardar o deshacer cambios al mismo tiempo).

12.3.5 Digitalización avanzada

Tabla 12.4: Barra de herramientas de edición avanzada de capas vectoriales

Icono	Propósito	Icono	Propósito
	Habilitar herramientas de digitalización avanzada		
	Mover Entidad(es)		Copiar y mover Entidad(es)
	Rotar objeto(s)		Simplificar objeto espacial
	Escala objeto espacial		
	Añadir anillo		Añadir parte
	Rellenar anillo		Cambiar dirección
	Borrar anillo		Borrar parte
	Desplazar curva		Remodelar objetos espaciales
	Dividir partes		Dividir objetos espaciales
	Combinar atributos de objetos espaciales seleccionados		Combinar objetos espaciales seleccionados
	Rotar símbolos de puntos		Símbolos de punto de desplazamiento
	Función de recorte o extensión		

Mover Entidad(es)

La herramienta  Mover Entidad(es) te permite mover entidades existentes:

1. Selecciona la entidad(es) a mover.
2. Haga clic en el lienzo del mapa para indicar el punto de origen del desplazamiento; puede confiar en las capacidades de ajuste para seleccionar un punto preciso.

   También puede aprovechar las *restricciones de digitalización avanzadas* para establecer con precisión las coordenadas del punto de origen. En ese caso:

   1. Primero haga click en el botón  para activar el panel.
   2. Escriba `x` e ingrese el valor correspondiente para el punto de origen que le gustaría usar. Luego presione el botón  junto a la opción para bloquear el valor.
   3. Haz lo mismo para la coordenada `y`.
   4. Haga click en el lienzo del mapa y su punto de origen se coloca en las coordenadas indicadas.
3. Muévase sobre el lienzo del mapa para indicar el punto de destino del desplazamiento, aún usando el modo de ajuste o, como se indicó anteriormente, use el panel de digitalización avanzada que proporcionaría restricciones de ubicación complementarias de `distancia` y `ángulo` para colocar el punto final de la traducción.
4. Haga clic en el lienzo del mapa: todas las entidades se mueven a una nueva ubicación.

Del mismo modo, puede crear una copia trasladada de la entidad(es) utilizando la herramienta  Copiar y mover objeto espacial(es).

Nota

Si no se selecciona ninguna entidad cuando hace click por primera vez en el lienzo del mapa con cualquiera de las herramientas *Mover entidad(es)* o *Copiar y mover entidad(es)*, entonces solo la entidad debajo del ratón es afectado por la acción. Por lo tanto, si deseas mover varias entidades, primero debes seleccionarlas.

Rotar objeto(s)

Use la herramienta  Rotar objeto(s) para rotar una o varias entidades en el lienzo del mapa:

1. Presiona el icono  Rotar objeto(s)
2. Luego haga click en la función para rotar. El centroide de la entidad se hace referencia como centro de rotación, se muestra una vista previa de la entidad rotada y se abre un widget que muestra el ángulo actual *Rotación*.
3. Haga click en el lienzo del mapa cuando esté satisfecho con la nueva ubicación o ingrese manualmente el ángulo de rotación en el cuadro de texto. También puede usar el cuadro *Autoensamblar* para restringir los valores de rotación.
4. Si deseas rotar muchas entidades a la vez, primero deben ser seleccionadas, y la rotación es por defecto alrededor del centroide de sus geometrías combinadas.

También puede usar un punto de ancla diferente del centroide de entidad predeterminado: presione el botón `Ctrl`, haga click en el lienzo del mapa y ese punto se usará como el nuevo centro de rotación.

Si se mantiene `Shift` antes de hacer clic en el mapa, la rotación se hará en pasos de 45 grados, que pueden ser modificados después en el widget de entrada del usuario.

Para cancelar la rotación de funciones, presione el botón: kbd: *ESC* o haga clic en el icono  Rotar objeto(s).

Escala objeto espacial

La herramienta  Scale Feature es similar a Rotate feature. Aunque en lugar de realizar una rotación de las características seleccionadas, vuelve a escalar su geometría. El cambio se realiza en relación con el punto de anclaje y la relación de escala se puede especificar manualmente en el widget que aparece en la esquina superior del lienzo.

Simplificar objeto espacial

La herramienta  Simplificar Objeto Espacial le permite remodelar de forma interactiva una geometría de línea o polígono reduciendo o densificando el número de vértices, siempre que la geometría siga siendo válida:

1. Selecciona la herramienta  Simplificar Objeto Espacial.
2. Click en la entidad o deslice un rectángulo sobre las entidades
3. Aparecerá un diálogo que le permitirá definir el *Método* a aplicar, es decir, si desea:
   - *simplificar el objeto espacial*, lo que significa menos vértices que el original. Los métodos disponibles son `Simplificar por distancia`, `Simplifica al ajustar a la cuadrícula` o `Simplificar por área (Visvalingam)`. Luego, deberá indicar el valor de *Tolerancia* en `Unidades de capa`, `Píxeles` o `Unidades de capa` para simplificar. Cuanto mayor sea la tolerancia, más vértices se pueden eliminar.
   - o *densificar las geometrías* con nuevos vértices gracias a la opción `Suavizar`: para cada vértice existente, se colocan dos vértices en cada uno de los segmentos que se originan en él, a una distancia *Desplazamiento* que representa el porcentaje de la longitud del segmento. También puede establecer el número de *Iteraciones* para procesar la ubicación: cuantas más iteraciones, más vértices y más suave es la entidad.

La configuración que utilizaste se guardará al salir de un proyecto o una sesión de edición. De esta manera podrás volver a los mismos parámetros la próxima vez que simplifique una función.

4. En la parte inferior del cuadro de diálogo se muestra un resumen de las modificaciones que se aplicarían, enumerando el número de entidades y el número de vértices (antes y después de la operación y la proporción que representa el cambio). Además, en el lienzo del mapa, la geometría esperada se muestra sobre la existente, utilizando el color de la banda elástica.
5. Cuando la geometría esperada se ajuste a sus necesidades, haga clic en *Aceptar* para aplicar la modificación. De lo contrario, para cancelar la operación, puede presionar *Cancelar* o hacer clic con el botón derecho en el lienzo del mapa.

Nota

A diferencia de la opción de simplificación de objetos espaciales en *Configuración -> Opciones -> Representación* del menú que simplifica la geometría solo para su representación, la herramienta Simplificar objeto espacial modifica permanentemente la geometría de la entidad en la fuente de datos.

Añadir parte

Puede Añadir parte a un objeto espacial seleccionado que genera una entidad multipunto, multilínea o multipolígono. La parte nueva debe digitalizarse fuera de la existente que debe seleccionarse previamente.

Añadir Parte también se puede utilizar para agregar una geometría a una función sin geometría. Primero, selecciona la entidad en la tabla de atributos y digitaliza la nueva geometría con la herramienta Añadir parte.

Nota

Orden de los vértices en las partes del polígono

A diferencia de las normas OGC, QGIS no obliga a que los vértices del límite exterior de una característica poligonal se ordenen en sentido contrario a las agujas del reloj. Por lo tanto, se pueden encontrar ambas direcciones en una capa. Sin embargo, todas las partes de una misma característica multipoligonal tendrán sus vértices exteriores ordenados siguiendo la misma dirección.

Sin embargo, puede utilizar el algoritmo *Fuerza la regla de la mano derecha* para restringir las características de una capa para que los vértices de sus límites exteriores estén ordenados en el sentido de las agujas del reloj.

Borrar parte

La herramienta Borrar parte le permite eliminar partes de múltiples características (por ejemplo, para eliminar polígonos de una característica de múltiples polígonos). Esta herramienta funciona con todas las geometrías de varias partes: punto, línea y polígono. Además, se puede utilizar para eliminar totalmente el componente geométrico de una entidad. Para eliminar una parte, simplemente haga clic dentro de la parte objetivo.

Añadir anillo

Puede crear polígonos de anillo usando el icono  :sup:»Añadir anillo» en la barra de herramientas. Esto significa que dentro de un área existente, es posible digitalizar más polígonos que se producirán como un *agujero*, por lo que solo el área entre los límites de los polígonos exterior e interior permanece como un polígono de anillo.

Para añadir un anillo:

1. Seleccione el(los) objeto(s) espacial(es) que desea modificar
2. Activar la herramienta  Añadir Anillo
3. Dibuja un polígono dentro de las geometrías seleccionadas, utilizando las *técnicas* mencionadas anteriormente. Aparece un agujero en las geometrías seleccionadas.
4. Si no hay ninguna geometría seleccionada cuando se dibuja el anillo, entonces se añade un agujero a cada uno de los polígonos sobre los que se dibuja el anillo.

Nota

Orden de los vértices en los anillos poligonales

A diferencia de las normas OGC, QGIS no obliga a que los vértices del límite exterior de un objeto espacial polígono se ordenen en sentido contrario a las agujas del reloj. Por lo tanto, se pueden encontrar ambas direcciones en una capa. Sin embargo, todos los anillos del mismo objeto espacial (multi)polígono tendrán sus vértices ordenados en la dirección opuesta a la del límite exterior.

Sin embargo, puede utilizar el algoritmo *Fuerza la regla de la mano derecha* para restringir los objetos espaciales de una capa para que los vértices de sus límites exteriores estén ordenados en el sentido de las agujas del reloj, y los vértices de sus anillos interiores estén ordenados en el sentido contrario a las agujas del reloj.

Rellenar anillo

La herramienta  Rellenar anillo le ayuda a crear una entidad poligonal que se encuentra totalmente dentro de otra sin ningún área superpuesta; esa es el nuevo objeto espacial que cubre un hueco dentro del existente. Para crear tal objeto espacial:

1. Selecciona la herramienta  Rellenar Anillo.
2. Dibuja un nuevo polígono sobre la entidad existente: QGIS agrega un anillo a su geometría (como si usaras la herramienta  Añadir Anillo) y crea una nueva entidad cuya geometría coincide con el anillo (como si tú *trazaras* sobre los límites interiores con la herramienta  Añadir Polígono).
3. O alternativamente, si el anillo ya existe en la función, coloque el mouse sobre el anillo y haga clic con el botón izquierdo mientras presiona `Shift`: se dibuja una nueva entidad que llena el agujero en ese lugar.

   El formulario *Artibutos de objeto espacial* de la nueva entidad se abre, precargado con valores de la entidad *principal* y/o *restricciones de campo*.

Borrar anillo

La herramienta Borrar Anillo le permite eliminar anillos dentro de un polígono existente, haciendo click dentro del agujero. Esta herramienta solo funciona con entidades poligonales y de múltiples polígonos. No cambia nada cuando se usa en el anillo exterior del polígono.

Remodelar objetos espaciales

Puede cambiar la forma de las entidades poligonales y de línea con la herramienta Remodelar Objetos en la barra de herramientas. Para las líneas, reemplaza la parte de la línea desde la primera hasta la última intersección con la línea original.

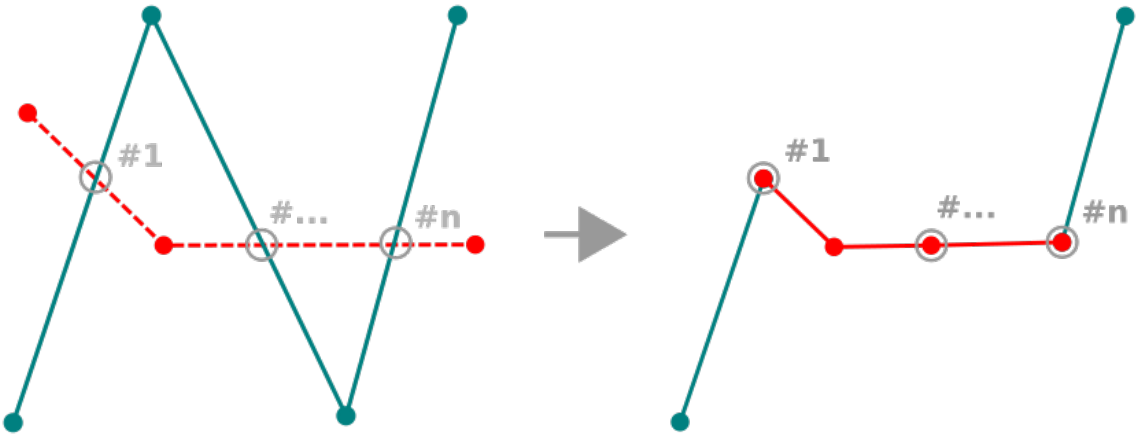


Figura 12.89: Remodelar linea

Truco

Extienda las geometrías de cadenas de líneas con la herramienta de remodelación

Utilice la herramienta Remodelar Objetos para extender las geometrías de cadenas de líneas existentes: seleccione el primer o último vértice de la línea y dibuje uno nuevo. Valide y la geometría de la función se convierte en la combinación de las dos líneas.

Para polígonos, cambiará la forma del límite del polígono. Para que funcione, la línea de la herramienta de remodelación debe cruzar el límite del polígono al menos dos veces. Para dibujar la línea, haga clic en el lienzo del mapa para agregar vértices. Para terminarlo, simplemente haz clic derecho. Al igual que con las líneas, solo se considera el segmento entre la primera y la última intersección. Los segmentos de la línea de remodelación que están dentro del polígono resultarán en un recorte, donde los que están fuera del polígono lo extenderán.

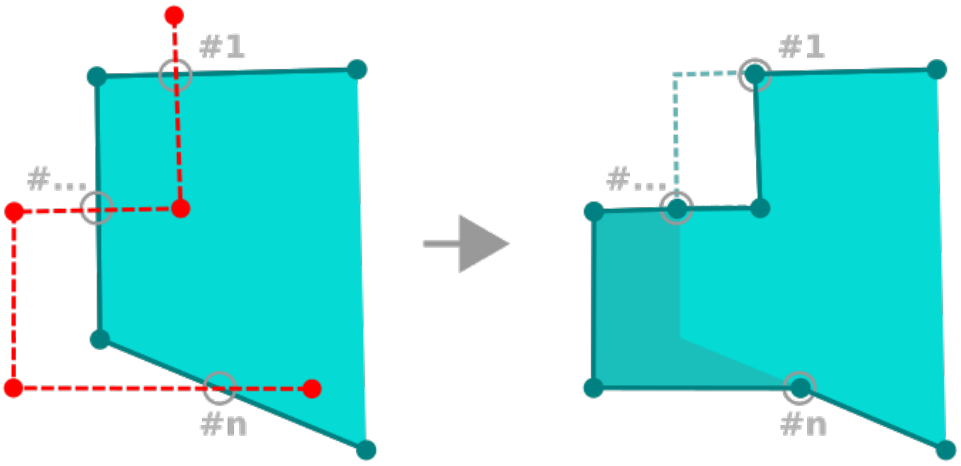


Figura 12.90: Remodelar polígono

Con los polígonos, la remodelación a veces puede dar lugar a resultados no deseados. Es principalmente útil para reemplazar partes más pequeñas de un polígono, no para revisiones importantes, y la línea de remodelación no puede cruzar varios anillos de polígono, ya que esto generaría un polígono no válido.

Nota

La herramienta de remodelar podría alterar la posición inicial de un anillo de polígono o una linea cerrada. Por lo tanto, el punto que está representado "dos veces" no será más el mismo. Esto puede no ser un problema para la mayoría de las aplicaciones, pero es algo a considerar.

Desplazar curva

La herramienta Curva de desplazamiento crea desplazamientos paralelos de capas de líneas. La herramienta puede aplicarse a la capa editada (se modifican las geometrías) o también a capas de fondo (en cuyo caso crea copias de las líneas / anillos y las añade a la capa editada). Por tanto, es ideal para la creación de capas de líneas de distancia. Aparece el diálogo *Entrada del usuario*, que muestra la distancia de desplazamiento y otros ajustes.

Para crear un desplazamiento de una capa de línea o polígono, primero debe entrar en modo edición y activar la herramienta Curva de desplazamiento. A continuación, haga clic en un objeto espacial para desplazarlo. Mueva el ratón y haga clic donde desee o introduzca la distancia deseada en el widget de entrada de usuario. Si mantiene pulsado `Ctrl` durante el segundo clic, se realizará una copia del desplazamiento. Los cambios pueden guardarse con la herramienta Guardar ediciones de capa.

Para geometrías en capas de fondo asegúrese de que el ajuste está activado y mantenga pulsado `Ctrl` para seleccionar la geometría del fondo. Mantenga también pulsado `Ctrl` cuando haga el segundo clic. Las geometrías se convertirán al tipo de geometría de la capa de destino.

El cuadro de diálogo Opciones de QGIS (pestaña Digitalización y a continuación la sección **Herramientas de desplazamiento de curvas**) o el icono **|configuración|** del cuadro de diálogo *Entrada del usuario* permiten configurar *algunos parámetros* como **Estilo de unión**, **Segmentos de cuadrante**, **Límite de inglete** y **Estilo de tapa final**.

Línea inversa

Changing the direction of a line geometry can be useful for cartographical purposes (e.g., for orienting symbols or labels that follow the line direction) or when preparing for network analysis where the direction of flow or travel matters.

This tool works only on line layers. For multipart lines, you can reverse individual parts without affecting the other parts of the feature.

Para cambiar la dirección de una línea:

1. Select a line layer in the *Layers* panel and make it editable.
2. Activate the reverse line tool by clicking  Reverse line in the *Advanced Digitizing* toolbar.
3. Move the pointer over a line feature. The part under the cursor is highlighted.
4. Click on the line: the direction of the line (or part) is reversed.
5. You can reverse other lines by clicking on them.

> **Truco**
>
> For bulk line direction changes, use the *Línea de dirección reversible* processing algorithm which can reverse all lines in a layer or a selection at once.

Dividir objetos espaciales

Use la  Dividir objetos herramienta para dividir una entidad en dos o más entidades nuevas e independientes, es decir. cada geometría correspondiente a una nueva fila en la tabla de atributos.

Para dividir entidades de línea o polígono:

1. Selecciona la herramienta  Dividir Objetos espaciales.
2. Trace una línea sobre el(los) objeto(s) espacial(es) que desea dividir. Si hay una selección activa, sólo se dividirán los elementos seleccionados. A continuación, se asigna al objeto espacial original la geometría más grande resultante de la división y se crean nuevos objetos espaciales para las partes restantes. Los campos de los objetos espaciales se rellenan/actualizan según las reglas del proveedor de la fuente de datos o su política de división ..
3. A continuación, como de costumbre, puede modificar cualquiera de los atributos de cualquier objeto espacial resultante.

> **Truco**
>
> **Divida una polilínea en nuevos objetos espaciales con un solo click**
>
> Usando la herramienta  Dividir Objetos Espaciales, ajustar y hacer clic en un vértice existente de una entidad polilínea para dividir esa entidad en dos entidades nuevas.

Dividir partes

En QGIS ahora es posible dividir las partes de un objeto espacial múlti-parte, de modo que se incrementa el número de partes. Sólo se tiene que dibujar una línea en la parte que se desea dividir utilizando el icono  Dividir partes

> **Truco**
>
> **Divida una polilínea en partes nuevas con un solo click**
>
> Usando la herramienta | splitParts | Dividir partes, ajuste y haga clic en un vértice existente de una entidad polilínea para dividir la entidad en dos nuevas polilíneas pertenecientes a la misma entidad.

Combinar objetos espaciales seleccionados

La herramienta  Combinar Objetos Espaciales Seleccionados le permite crear una nueva entidad fusionando las existentes: sus geometrías se fusionan para generar una nueva. Si las entidades no tienen límites comunes, se crea una entidad multipolígono / multipolínea / multipunto.

1. Primero, seleccione los objetos que quiere combinar.
2. Entonces presione el botón  Combinar Objetos Espaciales Seleccionados.
3. In the new dialog, the *Merge* line at the bottom of the table shows the attributes of the resulting feature. If any *Merge policy* have been defined, they are automatically applied to populate this row. However, you can override these values manually. Modify the values in the following ways:
   - reemplazar manualmente el valor en la celda correspondiente;
   - seleccionando una fila en la tabla y presionando *Tomar atributos del objeto espacial seleccionado* para usar los valores de este objeto inicial;
   - pulsando la etiqueta *Tomar los atributos de la geometría más grande* para utilizar los atributos de la característica de línea más larga, del polígono más grande o de los multipuntos con más partes;
   - presionando *Omitir todos los campos* para usar atributos vaciós;
   - expandiendo el menú desplegable en la parte superior de la tabla, seleccione cualquiera de las opciones anteriores para aplicar solo al campo correspondiente. Allí, también puede optar por agregar los atributos de características iniciales (Mínimo, Máximo, Mediana, Suma, Recuento, Concatenación… según el tipo de campo. Consulte *Panel de resumen estadístico* para ver la lista completa de funciones).

   > **Nota**
   >
   > Si la capa tiene valores predeterminados o clausulas presentes en los campos, estos se utilizan como valor inicial para la entidad fusionada.

4. Presiona *Aceptar* para aplicar las modificaciones. Un (multi)objeto espacial simple es creado en la capa, reemplazando aquellos previamente seleccionados.

Combinar atributos de objetos espaciales

La herramienta Combinar atributos de objetos espacioales seleccionados le permite aplicar los mismos atributos a entidades sin fusionar sus límites. El cuadro de diálogo es el mismo que el de la herramienta `Combinar objetos espaciales seleccionados`, excepto que, a diferencia de esa herramienta, los objetos seleccionados se mantienen con su geometría, mientras que algunos de sus atributos se hacen idénticos.

Rotar símbolos de puntos

The :sup:"Rotar símbolos de puntos" le permite cambiar individualmente la rotación de los símbolos de puntos en el lienzo del mapa.

1. Primero necesitará indicar el campo donde almacenar el valor de rotación. Esto se hace asignando un campo a la propiedad rotación del símbolo *definida por datos*:
   1. En *Propiedades de capa -> Simbología*, busque el diálogo del editor de símbolos.
   2. Haga clic en el widget *Suplantación definida por datos* próxima a la opción *Rotación* de la parte superior *Nivel de marcador* (preferiblemente) de las capas de símbolo.
   3. Elija un campo en el cuadro combinado *Tipo de campo*. Por lo tanto, los valores de este campo se utilizan para rotar en consecuencia el símbolo de cada objeto.

      También puede marcar la entrada Almacenar dato en proyecto para generar un campo *almacenamiento auxiliar de datos* para controlar el valor de la rotación.

> **Nota**
>
> ** Asegúrese de que se asigne el mismo campo a todas las capas de símbolos **
>
> La configuración del campo de rotación definido por datos en el nivel superior del árbol de símbolos lo propaga automáticamente a todas las capas de símbolos, un requisito previo para realizar la rotación de símbolos gráficos con la herramienta *Rotar símbolos de puntos*. De hecho, si una capa de símbolo no tiene el mismo campo adjunto a su propiedad de rotación, la herramienta no funcionará.

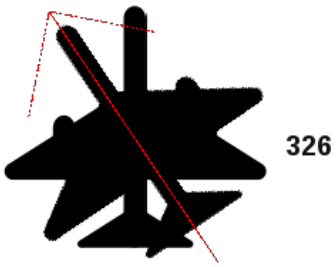


Figura 12.91: Rotando un símbolo de punto

2. Luego haga click en un símbolo de punto en el lienzo del mapa con la herramienta Rotar símbolos de puntos
3. Mueva el ratón alrededor. Se visualizará una flecha roja con el valor de rotación (ver Figura 12.91). Si mantiene presionada la tecla `Ctrl` mientras se mueve, la rotación se realizará en pasos de 15 grados.
4. Cuando obtenga el valor de ángulo esperado, haga click de nuevo. El símbolo se representa con esta nueva rotación y el campo asociado se actualiza en consecuencia.

Puede hacer click derecho para abortar la rotación del símbolo.

Símbolos de punto de desplazamiento

El  Desplazamiento de Símbolos de Punto le permite cambiar interactivamente la posición renderizada de los símbolos de puntos en el lienzo del mapa. Esta herramienta se comporta como la herramienta  Rotar símbolos de puntos excepto que requiere que conectes un campo a la propiedad :guilabel:`Desplazamiento (X, Y) `definida por los datos de cada capa del símbolo. Luego, el campo se completará con las coordenadas de desplazamiento de las entidades cuyo símbolo se mueve en el lienzo del mapa.

1. Asocie un campo al widget definido por datos de la propiedad *Desplazamiento (X, Y)* del símbolo. Si el símbolo está hecho con muchas capas, es posible que desee asignar el campo a cada una de ellas.
2. Seleccione la herramienta  Desplazar Símbolo de Puntos
3. Click en un símbolo de punto
4. Mueva a una nueva ubicación
5. Click de nuevo. El símbolo será movido a la nueva ubicación. Los valores de desplazamiento desde la posición original son almacenados en el campo enlazado.

   Puede hacer click derecho para abortar el desplazamiento del símbolo.

Nota

La herramienta  Desplazamiento de Símbolos de puntos no mueve la entidad de puntos en sí; deberías usar la herramienta  Herramientas de Vértices (Capa actual) o  Mover objeto espacial para este propósito..

Cortar/Extender Objeto espacial

The  Trim/Extend tool allows you to shorten or lengthen segments of a (multi)line or (multi)polygon geometry to converge with a selected segment (the cutting line). This results in a modified geometry with a vertex snapped to a reference segment or in its prolongation. Depending on how the selected geometries are placed in relation to each other, the tool will either:

- **Recortar**: quita partes del segmento de línea o de los límites del polígono, más allá de la linea de corte
- **Extender**: extiende los limites del polígono o los segmentos de la línea de modo que puedan encajar con la línea de corte.

Con el objetivo de recortar o extender geometrías existentes:

1. Seleccione la herramienta  Recortar/Extender
2. Click the reference segment, i.e., the segment with respect to which you want to extend or trim another segment. It doesn't need to be on the active layer. A red dotted line appears, visually extending the reference segment across the map canvas according to the active layer's CRS.
3. Hover over the target segment, i.e., the one you want to trim or extend. It does not need to be the last segment of the geometry, but has to be on the active layer. QGIS displays a preview of the feature's geometry with the segment limited to its intersection with the highlighted red dotted extending line.
   1. If an extension of the segment is required, click anywhere on the segment.
   2. In the case of a trim, click the part that should remain.
   3. In either case, the feature's geometry is updated accordingly. When both segments are in 3D, the tool performs an interpolation on the limit segment to get the Z value.

4. If you need to use the same reference segment for trimming or extending many features:
    1. Press `Shift` while selecting the reference segment.
    2. Click consecutively on the segments to modify and each will be trimmed or extended accordingly.

Nota

Snapping is automatically enabled when this tool is activated. Your original snapping settings will be restored once the tool is deactivated.

Atención

Preste atención a la geometría modificada mientras utilice la herramienta  Recortar/Extender. Dependiendo de las entradas, puede crear geometrías no válidas, con el resultado potencial de un fallo en la capa guardada.

12.3.6 Digitalización de formas

La barra de herramientas *Digitalizar forma* ofrece un conjunto de herramientas para dibujar líneas o polígonos de prestaciones de forma regular. Está sincronizada con el  Digitalizar forma *método de dibujo de geometría* puede seleccionar en la *Barra de herramientas de digitalización*. Para utilizarla:

1. Visualizar la barra de herramientas: *Ver* ► *Barras de herramientas* ► *Digitalización de formas*.
2. Seleccione una herramienta que cree o modifique la forma de una geometría, por ejemplo  Añadir objeto espacial línea,  Añadir objeto espacial polígono,  Añadir parte,  Añadir anillo,  Reformar objeto espacial, …
3. El botón  Digitalizar con segmento de la *Barra de herramientas de digitalización* está activado. La primera vez, es posible que tenga que cambiarlo por el botón  Digitalizar forma para activar las herramientas de la barra de herramientas *Digitalización de formas*.
4. Elige una herramienta de digitalización de formas y dibuja.

Cuerda circular por radio

El botón  Cadena circular por radio permite añadir características de línea o polígono con una geometría circular, dados dos nodos en la curva y un radio:

1. Haga clic con el botón izquierdo del ratón dos veces para colocar los dos puntos en la geometría.
2. Un widget *Radio* en la esquina superior derecha del lienzo del mapa muestra el radio actual (correspondiente a la distancia entre los puntos). Edite ese campo con el valor que desee.
3. Mientras se desplaza el cursor, se muestra un resumen de los arcos que coinciden con estas restricciones. Haga clic con el botón derecho para validar cuando se muestre el arco esperado.
4. Añada un nuevo punto para empezar a dar forma a otro arco.

Nota

** Las geometrías curvas se almacenan como tales solo en el proveedor de datos compatible **

Although QGIS allows to digitize curved geometries within any editable data format, you need to be using a data provider (e.g. PostgreSQL, Oracle Spatial, memory layer, GML or WFS) that supports curves to have features stored as curved, otherwise QGIS segmentizes the circular arcs.

Dibujar Círculos

Hay un conjunto de herramientas para dibujar círculos. Las herramientas se describen a continuación.

Los círculos se convierten en cadenas circulares. Por lo tanto, como se explica en *Cuerda circular por radio*, si el proveedor de datos lo permite, se guardará como una geometría curva, si no, QGIS segmentará los arcos circulares.

- **|circulo2Puntos|** Círculo a partir de 2 puntos: Los dos puntos definen el diámetro y la orientación del círculo. (Clic-izquierdo, clic-derecho)
- **|circulo3Puntos|** Círculo a partir de 3 puntos: Dibuja un círculo a partir de tres puntos conocidos del círculo. (Clic-izquierdo, clic-izquierdo, clic-derecho)
- Círculo por un punto central y otro punto: Dibuja un círculo con un centro dado y un punto en el círculo (Clic-izquierdo, clic-derecho). Cuando se utiliza con el panel_de_digitalización_avanzada esta herramienta puede convertirse en una herramienta de «Añadir círculo a partir de centro y radio» estableciendo y bloqueando el valor de la distancia tras el primer clic.
- Círculo a partir de 3 tangentes: Dibuja un círculo tangente a tres segmentos. **Tenga en cuenta que debe activar el ajuste a segmentos** (Ver *Configuración de la tolerancia de ajuste y el radio de búsqueda*). Haz clic en un segmento para añadir una tangente. Si dos tangentes son paralelas, las coordenadas del clic en la primera tangente paralela se utilizan para determinar la posición del círculo. Si tres tangentes son paralelas, aparece un mensaje de error y se borra la entrada. (Clic izquierdo, clic izquierdo, clic derecho)
- Círculo a partir de 2 tangentes y un punto: Similar al círculo a partir de 3 tangentes, excepto que tiene que seleccionar dos tangentes, introducir un radio y seleccionar el centro deseado.

Dibujar elipses

Hay un conjunto de herramientas para dibujar elipses. Las herramientas se describen a continuación.

Las elipses no se pueden convertir como cadenas circulares, por lo que siempre estarán segmentadas.

- Elipse de centro y dos puntos: Dibuja una elipse con un centro, eje mayor y eje menor dados. (Clic-izquierdo, clic-izquierdo, clic-derecho)
- Elipse de centro y un punto: Dibuja una elipse en una caja delimitadora con el centro y un vértice. (Clic-izquierdo, clic-derecho)
- Elipse a partir de la extensión: Dibuja una elipse en una caja delimitadora con dos esquinas opuestas. (Clic-izquierdo, clic-derecho)
- Elipse a partir de focos: Dibuja una elipse a partir de 2 puntos como focos y un punto en la elipse. (Clic-izquierdo, clic-izquierdo, clic-derecho)

Dibujar Rectángulos

Hay un conjunto de herramientas para dibujar rectángulos. Las herramientas se describen a continuación.

- Rectángulo desde el centro y un punto: Dibuja un rectángulo desde el centro y una esquina. (Clic izquierdo, clic derecho)
- Rectángulo desde extensión: Dibuja un rectángulo desde dos esquinas opuestas. (Clic izquierdo, clic derecho)
- Rectángulo a partir de 3 puntos (distancia): Dibuja un rectángulo orientado a partir de tres puntos. El primer y segundo punto determinan la longitud y el ángulo de la primera arista. El tercer punto determina la longitud de la otra arista. Se puede utilizar panel_de_digitalización_avanzada para establecer la longitud de los bordes. (Clic izquierdo, clic izquierdo, clic derecho)
- Rectángulo desde 3 puntos (proyectado): Igual que la herramienta anterior, pero la longitud del segundo borde se calcula a partir de la proyección del tercer punto en el primer borde. (Clic izquierdo, clic izquierdo, clic derecho)

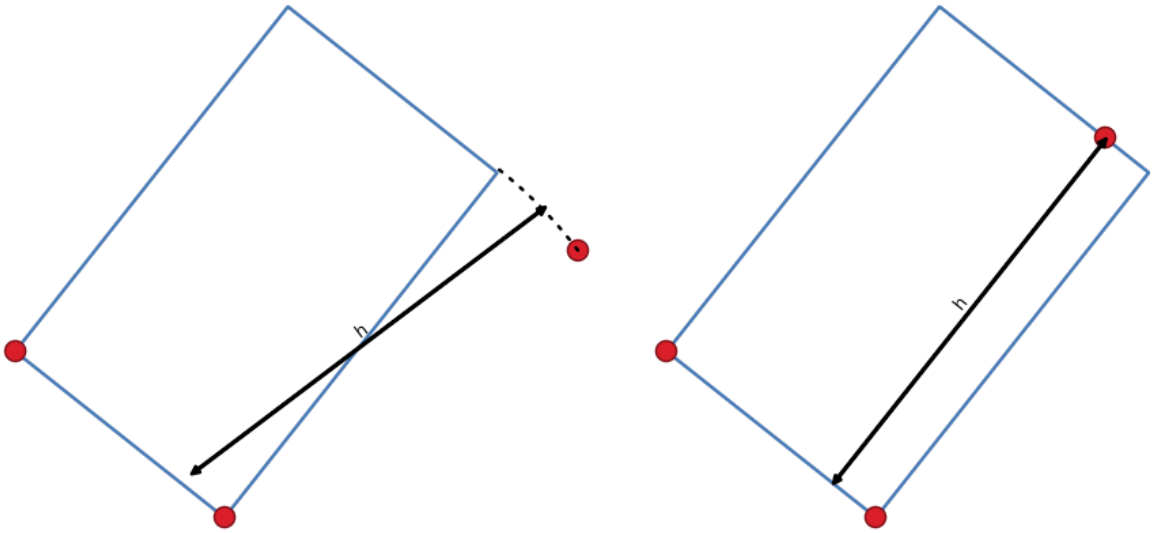


Figura 12.92: Dibuja un rectángulo desde 3 puntos usando la distancia (derecha) y proyectada (izquierda)

Dibujar Polígonos Regulares

Existe un conjunto de herramientas para dibujar polígonos regulares. Las herramientas se describen a continuación. Haga clic izquierdo para colocar el primer punto. Aparece un cuadro de diálogo donde puede establecer el número de bordes de polígono. Haz clic derecho para terminar el polígono regular.

- Polígono regular a partir de dos puntos: Dibuja un polígono regular donde los dos puntos determinan la longitud y el ángulo del primer borde.
- Polígono regular desde el centro y un punto: Dibuja un polígono regular desde el punto central proporcionado. El segundo punto determina el ángulo y la distancia a la mitad de un borde.
- Polígono regular desde centro y esquina: Igual que la herramienta anterior, pero el segundo punto determina el ángulo y distante a un vértice.

12.3.7 El panel de Digitalización Avanzada

Al capturar, remodelar o dividir geometrías nuevas o existentes, también tiene la posibilidad de utilizar el panel de digitalización avanzada. Puede digitalizar líneas exactamente paralelas o perpendiculares a un ángulo determinado o bloquear líneas a ángulos específicos. Además, puede definir con precisión su nueva geometría introduciendo coordenadas X e Y, así como Z para objetos espaciales 3D, o valores M.

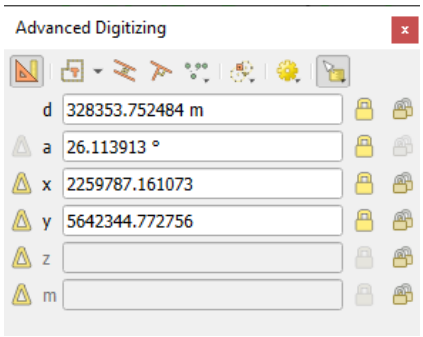


Figura 12.93: El panel de Digitalización Avanzada

El panel *Digitalización avanzada* puede abrirse haciendo clic con el botón derecho del ratón en la barra de herramientas, desde el menú *Ver* ► *Paneles* ► o pulsando `Ctrl+4`. Una vez visible el panel, haga clic en el botón Habilitar herramientas de digitalización avanzada para activar el conjunto de herramientas.

> **Nota**
>
> Las herramientas no están habilitadas si la vista del mapa esta en coordenadas geográficas.

El objetivo de la herramienta de digitalización avanzada es bloquear coordenadas, longitudes y ángulos al mover el mouse durante la digitalización en el lienzo del mapa.

También puede crear restricciones con referencia relativa o absoluta. Referencia relativa significa que los valores de las siguientes restricciones de vértice serán relativos al vértice o segmento anterior.

La Barra de Herramientas

En la parte superior del *Digitizing panel*, se encuentran los siguientes botones:

- Habilitar herramientas de digitalización avanzada
- Construction mode: permite capturar las posiciones de los clics para reutilizarlos como puntos de referencia para bloquear valores relativos de distancia, ángulo, X, Y, Z o M. Más detalles en *Modo de construcción*.
- Parallel o dibujar una línea paralela a una existente (más en *Líneas paralelas y perpendiculares*)
- Perpendicular para dibujar una línea perpendicular a una existente (más en *Líneas paralelas y perpendiculares*)
- Herramientas de Construcción proporciona un par de opciones que restringen la colocación de los vértices basándose en coordenadas extrapoladas de elementos existentes:
  - *Extensión de línea*: pase el ratón sobre un segmento y obtendrá una línea punteada púrpura que extiende el segmento a través del lienzo del mapa. Puede fijar el vértice en cualquier punto de esta línea virtual.
  - *Punto X/Y*: pasa el ratón por encima de un vértice y obtendrás una línea de puntos púrpura a lo largo de su coordenada X o Y, a través del lienzo del mapa. Puede fijar el vértice en cualquier punto de esta línea

virtual. Incluso es posible pasar el ratón sobre dos vértices diferentes, generando líneas de coordenadas virtuales para ambos, y ajustarse a su intersección.

-  Intersección de dos círculos: permite digitalizar un punto o vértice en la intersección de dos círculos. (más información en *Intersección de puntos de 2 círculos*).
- **|configuración|** Ajustar a ángulos comunes: al mover el cursor, muestra una línea virtual a la que puede ajustarse para añadir el siguiente vértice. La línea de ajuste está definida por el último vértice añadido y un ángulo (absoluto o relativo al segmento anterior) de una lista preestablecida (siguiendo pasos de 0,1°, 0,5°, 1°, 5°, 10°, 15°, 18°, 22,5°, 30°, 45° o 90°). Seleccione *No ajustar a ángulos comunes* para desactivar esta prestación.

  Ajustes a objetos espaciales puede utilizarse junto con el ajuste a ángulos comunes para una digitalización precisa. Para un control más preciso de cómo se conserva el elemento de destino al que se ajusta, puede indicar si desea dar prioridad al ajuste a los objetos espaciales sobre los ángulos comunes, y viceversa, en la entrada *Prioridad de ajuste*. Puede cambiar de un método a otro durante la operación de digitalización, y esto evita desactivar cualquiera de las opciones de ajuste mientras tanto. Pulse `N` (o `Mayúsculas+N`) durante una operación de digitalización para recorrer la lista de ángulos.
-  Configuración del flotador: si la opción *Mostrar flotador* está marcada, un menú contextual con información de digitalización sigue al cursor durante la digitalización. Se puede acceder a los valores utilizando los *accesos directos del panel*, editado y **|bloqueado|** Bloqueado tras la validación (pulsando `Intro`). El tipo de información a mostrar puede seleccionarse en la parte inferior del menú:
  - *Mostrar distancia*
  - *Mostrar ángulo*
  - *Mostrar coordenadas XY*
  - *Mostrar valor Z*
  - *Visualizar valor M*
  - *Mostrar rumbo/azimut.*
  - *Mostrar ángulo de autoajuste común*

Debajo de la barra de herramientas, encontrará una serie de cuadros de texto cuyo valor refleja por defecto la posición o el movimiento del cursor en el lienzo del mapa. La edición de estos valores le ayuda a restringir la posición de los elementos que edita:

- *d* para la distancia desde una posición de referencia, normalmente el último vértice editado
- *a* para el ángulo (absoluto o relativo) desde una posición de referencia, normalmente el último segmento editado
- *x* para la coordenada X del puntero
- *y* para la coordenada Y del puntero
- *z* para el valor Z por defecto o la coordenada Z del vértice o segmento bajo el puntero
- *m* para el valor M por defecto o el valor M del vértice o segmento bajo el puntero

Atajos de teclado

Para acelerar el uso del Panel de digitalización avanzado, hay un par de atajos de teclado disponibles:

Tabla 12.5: Atajos de teclado de las herramientas del Panel de Digitalización Avanzada

Clave	Simple	`Ctrl+` o `Alt+`	`Shift+`
`D`	Establecer distancia	Bloquear distancia	
`A`	Establecer ángulo	Bloquear ángulo	Conmutar ángulo relativo a último segmento
`X`	Establecer Coordenada X	Bloquear coordenada X	Conmutar X relativa al último vértice
`Y`	Establecer Coordenada Y	Bloquear coordenada Y	Conmutar Y relativa al último vértice
`Z`	Establecer coordenada Z	Bloquear coordenada Z	Alternar Z relativo al último vértice
`M`	Establecer valor M	Bloquear valor M	Alternar M relativo al último vértice
`C`	Conmutar modo construcción		
`P`	Conmutar modos perpendicular y paralelo		

Nota

Las opciones de coordenada Z y valor M solo están disponibles si son compatibles con la dimensión de la geometría de la capa.

Digitalización de referencia absoluta

Al dibujar una nueva geometría desde cero, es muy útil tener la posibilidad de comenzar a digitalizar vértices en coordenadas dadas.

Por ejemplo, para agregar una nueva entidad a una capa poligonal, haga clic en el botón  . Puede ingresar las coordenadas exactas donde desea comenzar a editar la función, es decir:

1. Haga clic en el cuadro de texto *x* (o utilice el método abreviado de teclado:kbd:*X*).
2. Escriba el valor de la coordenada X que desee y presione `Enter` o haga clic en  a su derecha para bloquear el mouse en el eje X en el lienzo del mapa.
3. Haga clic en el cuadro de texto *y* (o use el atajo de teclado `Y`).
4. Escriba el valor de la coordenada Y que desee y presione `Enter` o haga clic en  a su derecha para bloquear el mouse en el eje Y en el lienzo del mapa.
5. Si la capa tiene coordenadas Z o valores M, el widget *z* o *m* correspondiente se activa y muestra su valor por defecto, tal y como se ha establecido en la pestaña *Configuración* ► *Opciones* ► *Digitalización*.
    1. Haga clic en el cuadro de texto *z* o *m* (o utilice respectivamente el atajo de teclado `Z` o `M`).
    2. Escriba el valor de coordenadas que desee y pulse `Intro` o haga clic en el botón  situado a su derecha para bloquear el valor en el widget.

    Nota

    Lea *Reglas de asignación de la coordenada Z o del valor M* para más detalles sobre cómo se determinan automáticamente las coordenadas Z y los valores M a partir de los objetos espaciales existentes.

6. Dos líneas de puntos azules y una cruz verde identifican las coordenadas exactas introducidas. Haz clic en el lienzo del mapa para añadir un vértice en la posición de la cruz verde.

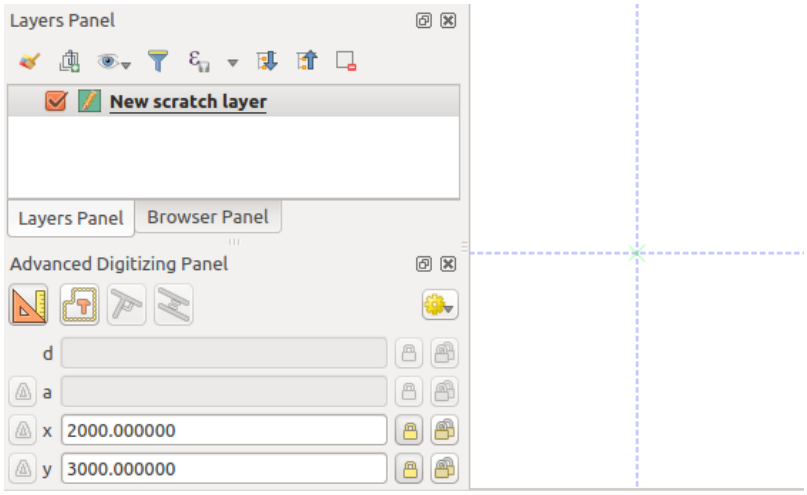


Figura 12.94: Empiece a dibujar en las coordenadas dadas

7. Puede proceder como en el caso anterior, añadiendo un nuevo conjunto de coordenadas para el siguiente vértice, o cambiar a otro *modo de digitalización* (por ejemplo, segmento, curva o flujo).
8. Si desea dibujar un segmento de una longitud dada:
    1. Haga clic en el cuadro de texto *d (distance)* (atajo de teclado `D`)
    2. Escriba el valor de la distancia (en unidades de mapa)
    3. Presione `Enter` o haga clic en  a la derecha para bloquear el mouse en el lienzo del mapa a la longitud del segmento. En el lienzo del mapa, el último vértice está rodeado por un círculo cuyo radio es el valor ingresado en el cuadro de texto de distancia. Una cruz en el círculo muestra la posición del siguiente vértice si hace clic.

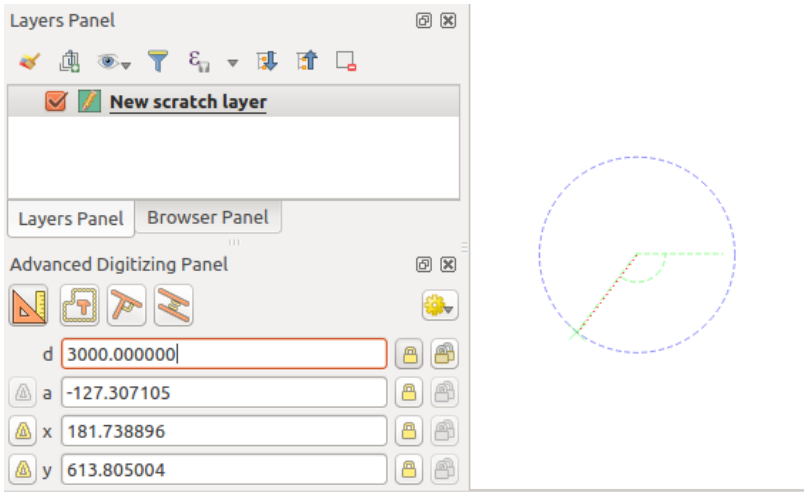


Figura 12.95: Segmento de longitud fija

9. También puede restringir la posición del vértice, configurando el ángulo del segmento. Como se describió antes:
    1. Haga clic en el cuadro de texto *a (angle)* (atajo de teclado `A`)
    2. Escriba el valor del ángulo (en grados)
    3. Presione `Enter` o haga clic en el botón  de la derecha para bloquearlo. En el lienzo del mapa aparece una línea que pasa por el último vértice y gira en función del ángulo establecido, y una cruz muestra la siguiente posición del vértice si hace clic.

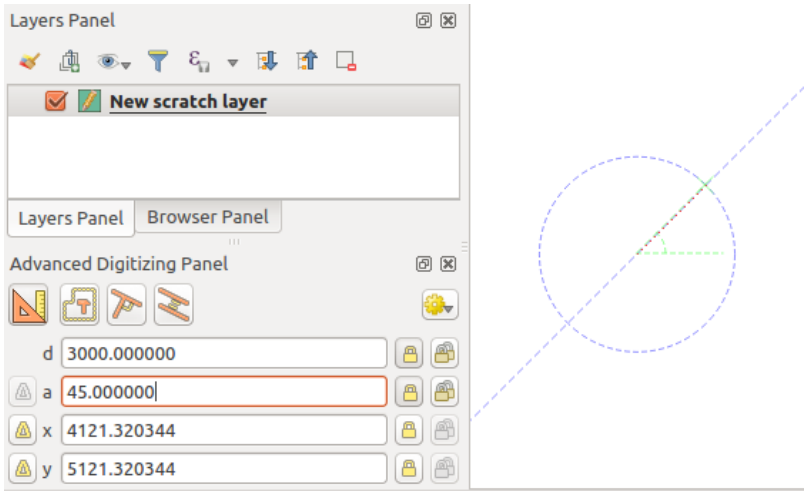


Figura 12.96: Segmento de ángulo fijo

Consejo

Presionar `Ctrl+<key>` o `Alt+<key>` automáticamente bloquea la propiedad de destino y pone su valor en edición. Modifique, presione `Enter` y listo. Combinado con el |flotante| Toggle floater, esto puede ser un ahorro de tiempo real, con la digitalización del teclado.

Digitalización de referencia relativa

En lugar de utilizar valores absolutos de ángulos o coordenadas, también puede utilizar valores relativos al último vértice o segmento digitalizado.

Para los ángulos, puede hacer clic en el botón  a la izquierda del cuadro de texto *a* (o presione :kbd:` Shift + A`) para alternar los ángulos relativos al segmento anterior. Con esa opción activada, los ángulos se miden entre el último segmento y el puntero del mouse.

Para las coordenadas, haga clic en el  a la izquierda de los cuadros de texto *x*, *y*, *z* o *m* (o presione `Shift+<key>`) para cambiar las coordenadas relativas a el vértice anterior. Con estas opciones activadas, la medición de coordenadas considerará el último vértice como el origen de las coordenadas establecidas.

Bloqueo continuo

Tanto en la digitalización de referencias absolutas como relativas, las restricciones de ángulo, distancia, X, Y, Z y M se pueden bloquear de forma continua haciendo clic en *Continuous lock*. El uso de bloqueo continuo le permite digitalizar varios puntos o vértices usando las mismas restricciones.

Modo de construcción

Puede habilitar y deshabilitar el *construction mode* haciendo clic en el ícono Construction mode o con el atajo de teclado `C`. Mientras está en el modo de construcción, hacer clic en el lienzo del mapa no agregará nuevos vértices, pero capturará las posiciones de los clics para que pueda usarlos como puntos de referencia para luego bloquear la distancia, el ángulo o los valores relativos X, Y, Z, M.

En Modo de construcción encontrará un menú desplegable en el que podrá elegir:

- *Guías de construcción de registros*: Todos los pasos de construcción se representan como líneas discontinuas. Cada paso que realice se representa visualmente, permitiéndole seguir el proceso de construcción. Las guías se muestran mientras *Digitalización avanzada* permanezca activa.
- *Mostrar Guías de Construcción*: Permite hacer visibles u ocultar las guías de construcción en el lienzo. Cuando está activada, se muestran todas las guías activas, ofreciendo una mejor orientación espacial y precisión durante el proceso de digitalización.
- *Ajustar a guías de construcción visibles*: Las guías son encajables, lo que le permite iniciar nuevos pasos de construcción desde cualquier punto a lo largo de las guías existentes.
- Elija *Borrar Guías de Construcción* para eliminar todas las guías del lienzo.

Como ejemplo, el modo de construcción se puede utilizar para dibujar algún punto a una distancia exacta de un punto existente.

Con un punto existente en el lienzo del mapa y el modo de alineación activado correctamente, puede dibujar fácilmente otros puntos a distancias y ángulos dados desde él. Además del botón, tiene que activar también el *construction mode* haciendo clic en el ícono Construction mode o con el atajo de teclado `C`.

Haga clic junto al punto desde el cual desea calcular la distancia y haga clic en el cuadro *d* (:kbd: acceso directo` D`) escriba la distancia deseada y presione `Enter` para bloquear la posición del mouse en el lienzo del mapa:

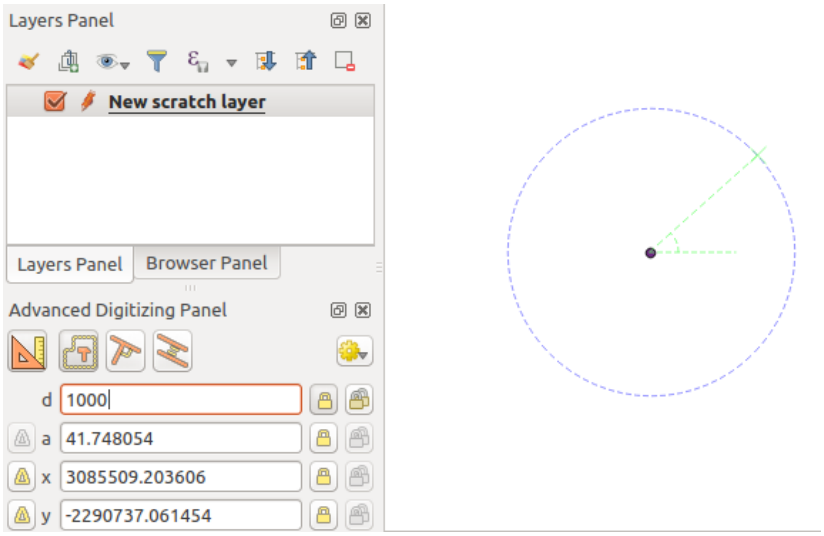


Figura 12.97: Distancia desde punto

Antes de agregar el nuevo punto, presione `C` para salir del modo de construcción. Ahora, puede hacer clic en el lienzo del mapa y el punto se colocará a la distancia ingresada.

También puede usar la restricción de ángulo para, por ejemplo, crear otro punto a la misma distancia del original, pero en un ángulo particular desde el punto recién agregado. Haga clic en el ícono Construction mode o con el atajo de teclado `C` para ingresar al modo de construcción. Haga clic en el punto agregado recientemente y luego en el otro para establecer un segmento de dirección. Luego, haga clic en el cuadro de texto *d* (acceso directo a `D`) escriba la distancia deseada y presione `Enter`. Haga clic en el cuadro de texto *a* (acceso directo a `A`), escriba el ángulo que desee y presione `Enter`. La posición del mouse se bloqueará tanto en distancia como en ángulo.

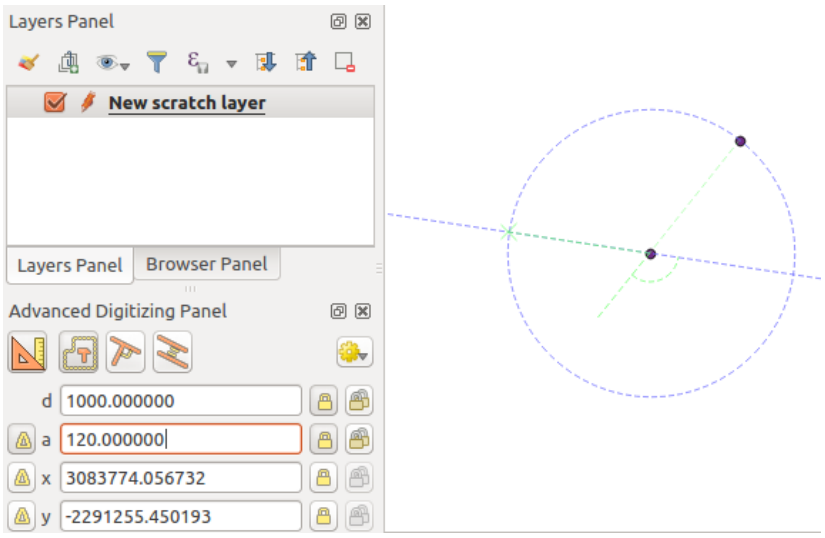


Figura 12.98: Distancia y ángulo desde puntos

Antes de agregar el nuevo punto, presione `C` para salir del modo de construcción. Ahora, puede hacer clic en el lienzo del mapa y el punto se colocará a la distancia y el ángulo ingresados. Repitiendo el proceso, se pueden agregar varios puntos.

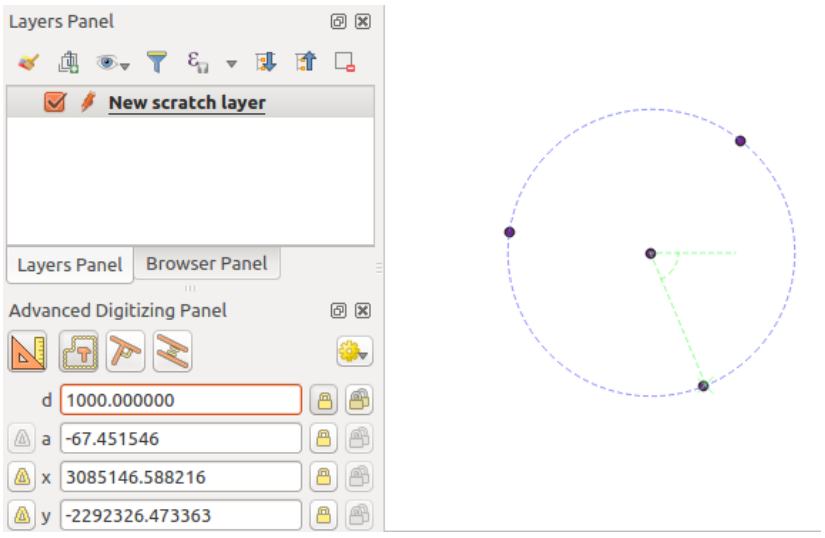


Figura 12.99: Puntos a una distancia y un ángulo dados

Líneas paralelas y perpendiculares

Todas las herramientas descritas anteriormente se pueden combinar con el Perpendicular y las herramientas Parallel. Estas dos herramientas permiten dibujar segmentos perfectamente perpendiculares o paralelos a otro segmento. El segmento de destino puede estar en otra capa, otra entidad dentro de la capa o la entidad que se está digitalizando (requiere *self-snapping option*).

Para dibujar un segmento *perpendicular*:

1. Primero agrega uno de los vértices del segmento.
2. Haga clic en Perpendicular (atajo de teclado `P`) para activarlo.
3. Haga clic en el segmento al que desea ser perpendicular.
4. Aparece una línea de puntos virtual perpendicular al segmento que pasa por el vértice anterior. La propiedad de ángulo está bloqueada, restringiendo el siguiente vértice en esa línea y una cruz indica la posición proyectada del cursor en la línea. Haga clic para colocar el nuevo vértice.

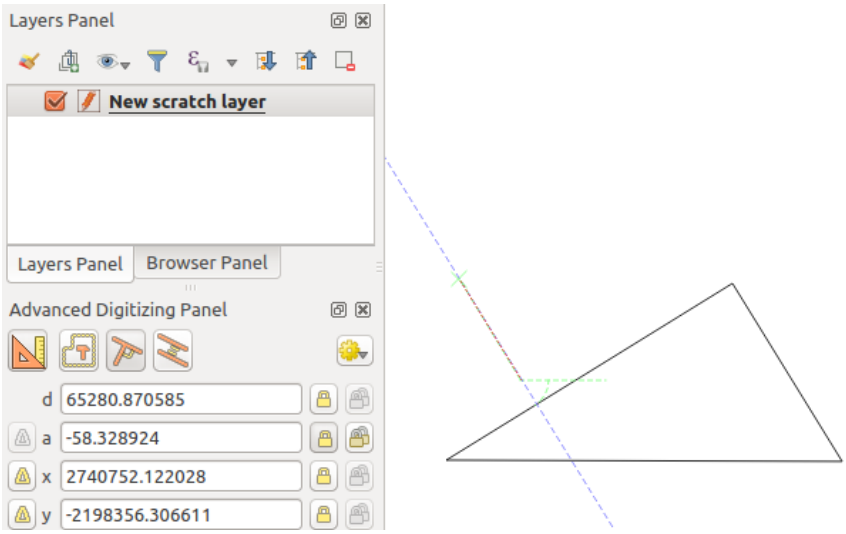


Figura 12.100: Digitalización perpendicular

Para dibujar un segmento *parallel*, los pasos son los mismos excepto que debe hacer clic en el ícono Parallel (atajo de teclado `P` dos veces).

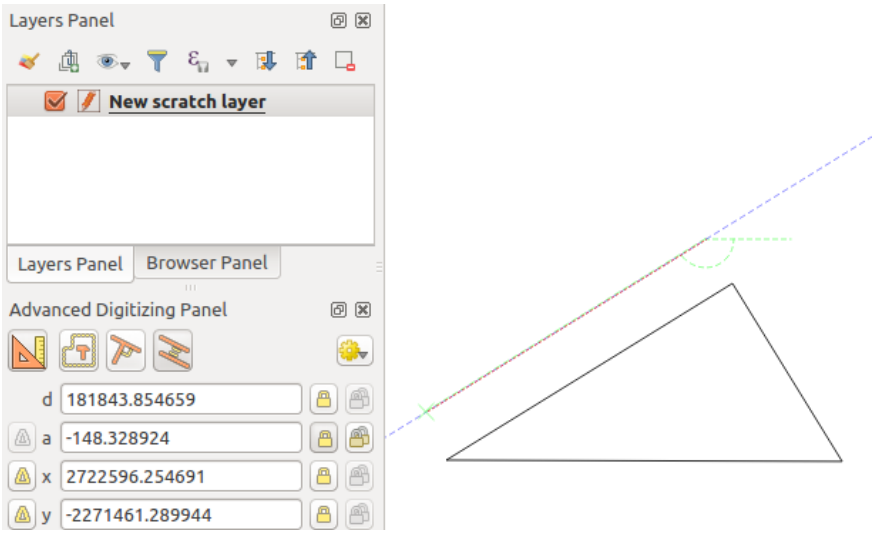


Figura 12.101: Digitalización paralela

Estas dos herramientas solo encuentran el ángulo correcto del ángulo perpendicular y paralelo y bloquean este parámetro durante su edición. Desbloquee el parámetro de ángulo para cancelar su uso en medio del proceso.

Intersección de puntos de 2 círculos

Para añadir un vértice en la intersección de dos círculos, sigue estos pasos:

1. Haga clic en el icono Intersección de puntos de 2 círculos.
2. Se abrirá un cuadro de diálogo en el que podrá definir los parámetros de *Círculo #1* y *Círculo #2*.
3. Haga clic en el lienzo del mapa y la herramienta calculará automáticamente las coordenadas *X* y *Y* de los centros de ambos círculos.
4. Introduzca la distancia *d* para cada círculo.
5. La herramienta calculará y mostrará los dos puntos de intersección de los círculos.
6. Haga clic en uno de los puntos de intersección para añadir el nuevo vértice.

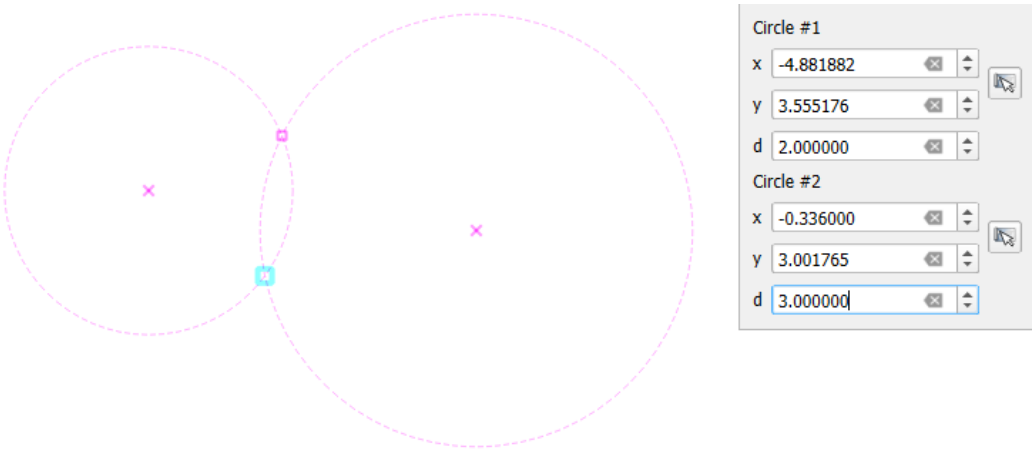


Figura 12.102: Intersección de puntos de 2 círculos

12.3.8 El modificador de capa de procesamiento in situ

El *Menú de procesamiento* proporciona acceso a un gran conjunto de herramientas para analizar y crear nuevos objetos espaciales basados en las propiedades de las entidades de entrada o sus relaciones con otras entidades (dentro de la misma capa o no). Si bien el comportamiento común es crear nuevas capas como salidas, algunos algoritmos también permiten modificaciones en la capa de entrada. Esta es una forma práctica de automatizar la modificación de múltiples funciones mediante operaciones avanzadas y complejas.

Para editar entidades in situ:

1. Seleccione la capa para editar en el panel *Capas*.
2. Seleccione las funciones correspondientes. Puede omitir este paso, en cuyo caso la modificación se aplicará a toda la capa.
3. Presione el botón Editar funciones in situ en la parte superior de :ref:`Caja de herramientas de procesamiento <caja de herramientas de procesamiento>`. La lista de algoritmos se filtra, mostrando solo aquellos compatibles con modificaciones in situ, es decir:
   - Funcionan en la fuente de la entidad y no a nivel de capa.
   - No cambian la estructura de capas, p. Ej. agregar o eliminar campos.
   - No cambian el tipo de geometría, p. Ej. de línea a capa de puntos.

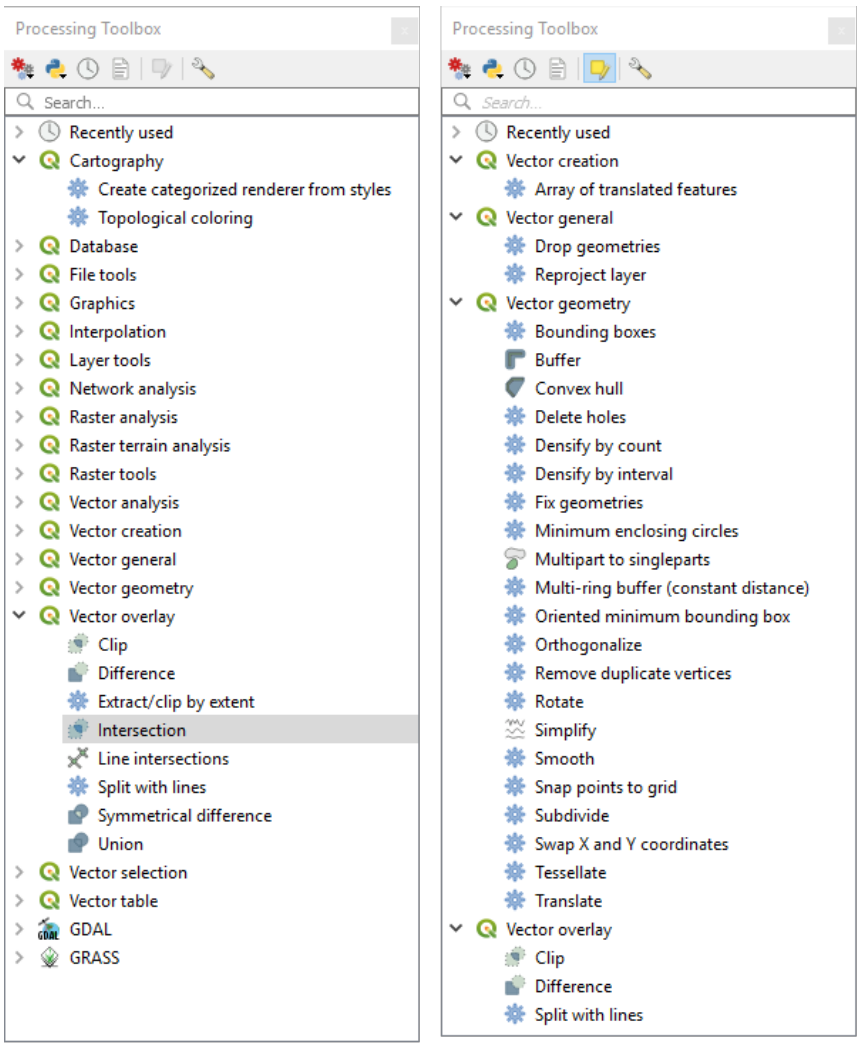


Figura 12.103: Algoritmos de procesamiento: todos (izquierda) frente a editores in situ de polígonos (derecha)

4. Busque el algoritmo que le gustaría ejecutar y haga doble clic en él.

   Nota

   Si el algoritmo no necesita ningún parámetro adicional establecido por el usuario (excluyendo los parámetros habituales de la capa de entrada y salida), entonces el algoritmo se ejecuta inmediatamente sin ningún cuadro de diálogo emergente.

   1. Si se necesitan parámetros distintos de las capas de entrada o salida habituales, aparece el cuadro de diálogo del algoritmo. Complete la información requerida.
   2. Click en *Modificar Objetos espaciales seleccionados* o *Modificar Todos los objetos Espaciales* dependiendo de si hay una selección activa.

   Los cambios se aplican a la capa y se colocan en el búfer de edición: de hecho, la capa se cambia al modo de edición con modificaciones no guardadas como se indica en  junto al nombre de la capa.

5. Como de costumbre, presione  Guardar cambios de la capa para confirmar los cambios en la capa. También puede presionar **|deshacer|** Deshacer para deshacer toda la modificación.

12.4 Conexión y edición de datos entre capas

Una de las funciones de los programas informáticos de SIG es conectar datos de distintas capas. Dicha conexión puede basarse en la relación espacial entre los objetos espaciales o en sus atributos compartidos. QGIS proporciona herramientas para manejar cualquiera de estas asociaciones, tales como:

- Algoritmos de procesamiento que pueden crear una nueva capa como resultado de la conexión, a saber *Unir atributos por localización*, *Unir atributos por proximidad*, *Unir atributos por valor de campo*, …
- Consultas SQL para crear una nueva capa a partir del *Gestor BD* o como *capa virtual*
- *Propiedades de las uniones* o *configuración de las relaciones* que amplían temporalmente los atributos de los objetos espaciales de una capa determinada, con los de objetos espaciales de otra capa en función de algún atributo o atributos coincidentes.

  Las uniones y relaciones son conceptos técnicos tomados de las bases de datos para sacar el máximo partido de los datos almacenados en tablas combinando sus contenidos. La idea es que objetos espaciales (filas) de distintas capas (tablas) puedan asociarse entre sí. El número de filas que coinciden entre sí puede ser de cualquier valor (cero, uno, muchos).

12.4.1 Unir objetos espaciales entre dos capas

La **uniones** en QGIS permiten asociar objetos espaciales de la capa actual a objetos espaciales de otra capa vectorial cargada. No importa si están habilitadas espacialmente ni el tipo de geometría. La unión se basa en un atributo compartido por las capas, en una relación de uno a uno.

Para crear una unión en una capa (identificada a continuación como `capa destino`):

1. Vaya a la pestaña de la capa *Propiedades* ►  *Uniones*
2. Click en el botón  Add new join. El diálogo *Add vector join* aparece.
3. Seleccione la *Join layer* que quiera conectar con la capa vectorial de destino
4. Especifique el campo *Unir* (de la `capa unir`) y el campo *Destino* (de la `capa destino`). Estos son los campos que se utilizan para encontrar objetos espaciales coincidentes en ambas capas, por lo que deben tener valores comunes.

5. Presione *OK* y se agrega un resumen de los parámetros seleccionados al panel *Join*.

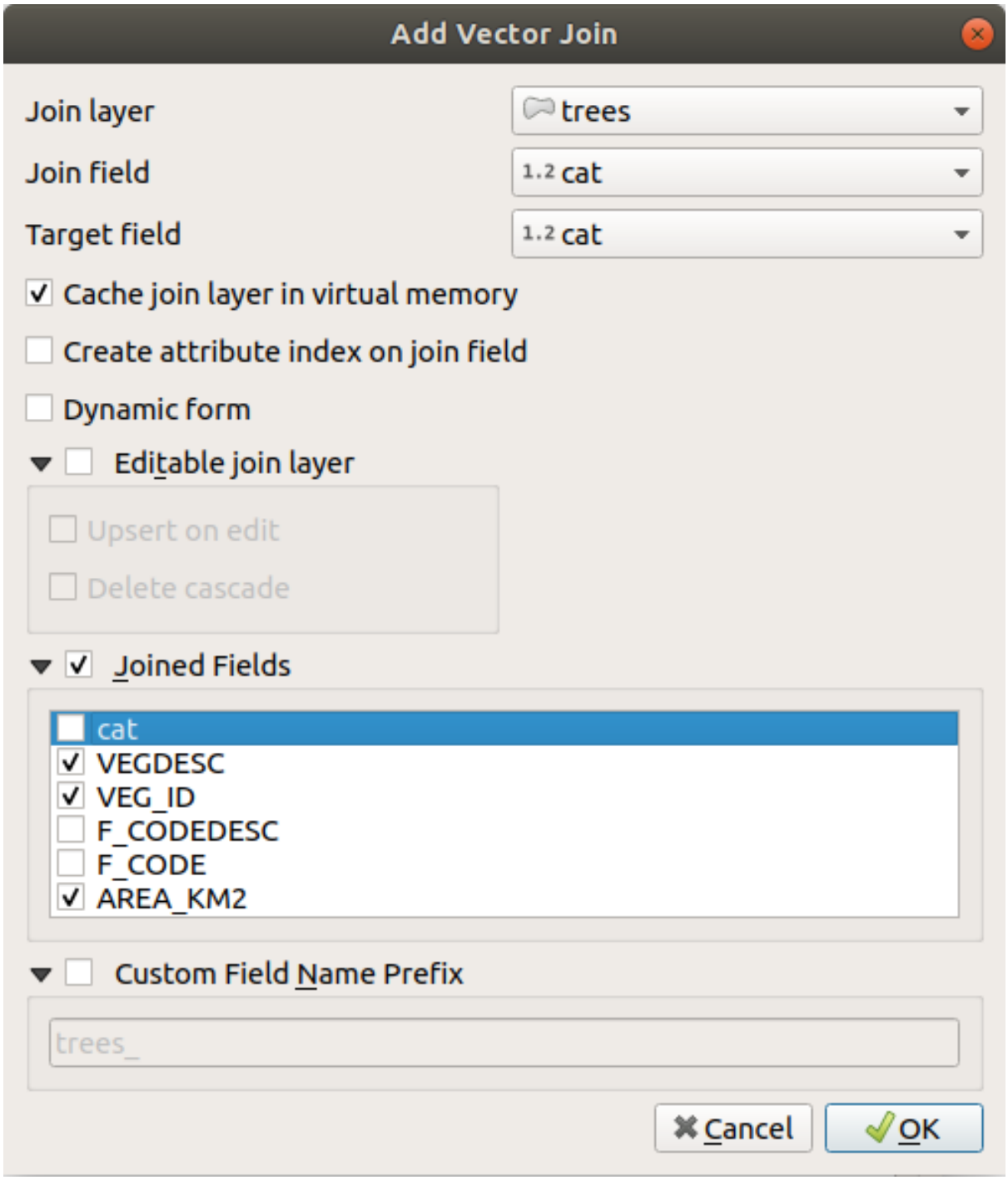


Figura 12.104: Unir una tabla de atributos a una capa vectorial existente

Los pasos anteriores crearán una unión en la que **TODOS** los atributos de la **primer objeto espacial coincidente** de la capa de unión se añadirán a la característica de la capa de destino. La siguiente lógica se utiliza para emparejar objetos espaciales durante un proceso de unión:

- Se devuelven todas los objetos espaciales de la capa de destino, independientemente de que coincidan
- Si el campo de destino contiene valores duplicados, a estos objetos espaciales se les asigna el mismo objeto espacial de la capa de unión.
- Si el campo de unión contiene valores coincidentes duplicados, sólo se selecciona el primero objeto espacial

obtenido.

> **Nota**
>
> Las uniones en QGIS se basan en la coincidencia de un solo campo, por lo que la mayoría de las veces, usted querrá asegurarse de que los valores en los campos coincidentes son únicos.

QGIS ofrece algunas opciones más para ajustar la unión:

- *Cache join layer in virtual memory*: le permite almacenar valores en la memoria caché (sin geometrías) de la capa unida para acelerar las búsquedas.
- *Crear índice de atributos en el campo join* para acelerar las búsquedas
- *Dynamic form*: ayuda a sincronizar campos de unión sobre la marcha, de acuerdo con el *Target field*. De esta forma, las restricciones para los campos de unión también se actualizan correctamente. Tenga en cuenta que está desactivado de forma predeterminada porque puede llevar mucho tiempo si tiene muchas funciones o una gran cantidad de combinaciones.
- Si la capa de destino es editable, se mostrarán algunos iconos en la tabla de atributos al lado de los campos, para informar sobre su estado:
  - : la capa de unión no está configurada para ser editable. Si desea poder editar características de unión desde la tabla de atributos de destino, debe marcar la opción *Editable join layer*.
  - : la capa de unión está bien configurada para ser editable, pero su estado actual es de solo lectura.
  - : la capa de unión es editable, pero los mecanismos de sincronización no están activados. Si desea añadir automáticamente una prestación en la capa de unión cuando se cree una prestación en la capa de destino, deberá activar la opción *Agregar al editar*. Simétricamente, la opción *Eliminar en cascada* puede activarse si desea eliminar automáticamente los objetos espaciales de unión.
- *Joined fields*: en lugar de agregar todos los campos de la capa unida, puede especificar un subconjunto.
- *Custom field name prefix* para campos unidos, para evitar la colisión de nombres

12.4.2 Establecer relaciones entre varias capas

A diferencia de las uniones, que definen un vínculo de uno a uno entre objetos espaciales de dos capas, las relaciones le ayudan a crear interconexiones entre múltiples prestaciones de dos o más capas. Como tales, las relaciones son ajustes a nivel de proyecto y se establecen en la pestaña *Proyecto* ► *Propiedades* ► |relaciones| *Relaciones*. Desde ahí, puede:

- *Añadir relación* cuyo tipo puede ser:
  - *relación uno a muchos*
  - *relación muchos a muchos*
  - *relación polimórfica* que puede añadir o editar con las herramientas específicas del menú desplegable de acciones.

> **Nota**
>
> Todavía no existe una forma sencilla de editar una relación no polimórfica una vez creada. Sólo se puede editar el nombre con un doble clic. Para cualquier otro parámetro de dicha relación tendrás que eliminarla y volver a crearla.

- *Descubrir relaciones*: QGIS es capaz de descubrir relaciones existentes desde los formatos de base de datos soportados (PostgreSQL, GeoPackage, ESRI File Geodatabase, …). Esta puede ser una buena forma de facilitar la definición de relaciones.
- *Eliminar relación*

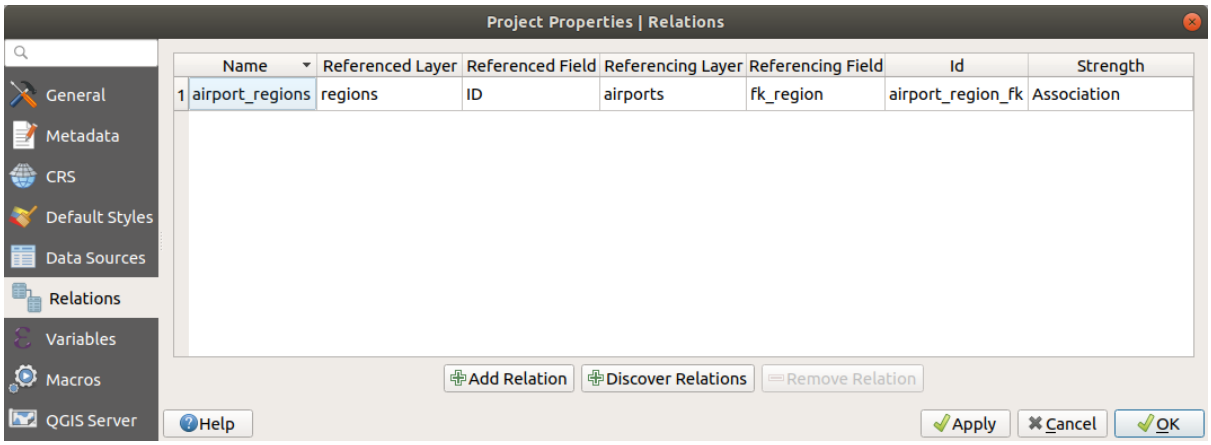


Figura 12.105: Pestaña Relaciones

Relaciones uno a muchos (1-N)

Por ejemplo, se tiene una capa con todas las regiones de Alaska (polígono) que proporciona algunos atributos sobre su nombre y tipo de región y un identificador único (que actúa como clave primaria).

Luego, obtiene otra capa de puntos o tabla con información sobre los aeropuertos que se encuentran en las regiones y también desea realizar un seguimiento de estos. Si desea agregarlos a la capa de regiones, debe crear una relación de uno a muchos utilizando claves externas, porque hay varios aeropuertos en la mayoría de las regiones.

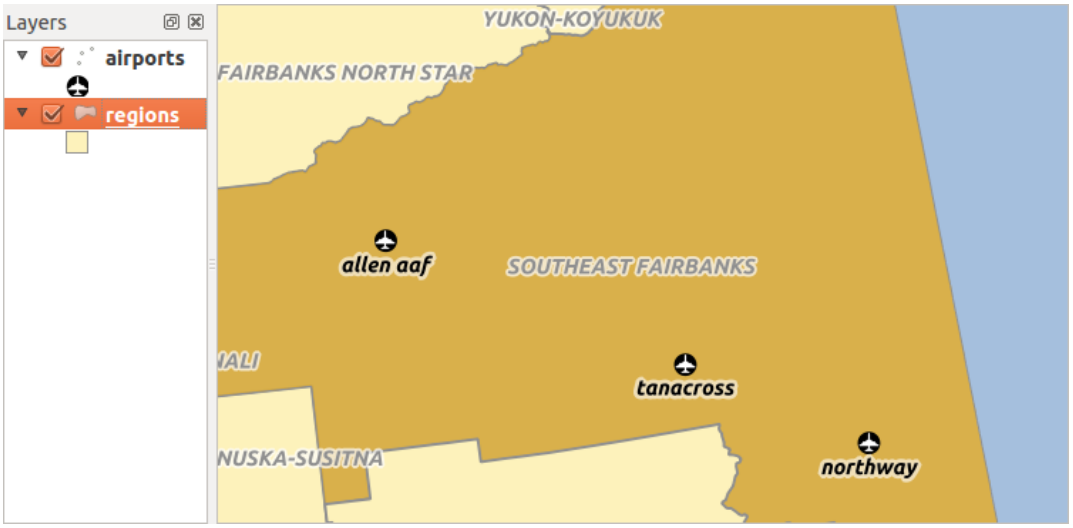


Figura 12.106: Región de Alaska con aeropuertos

Capas y llaves

QGIS no diferencia entre una tabla y una capa vectorial. Básicamente, una capa vectorial es una tabla con una geometría. Así que puede añadir su tabla como una capa vectorial. Para demostrar la relación 1-n, puede cargar las capas `regions` y `airports` en el conjunto de datos de ejemplo. En la práctica, cada aeropuerto pertenece exactamente a una región, mientras que cada región puede tener cualquier número de aeropuertos (una relación típica de uno a muchos).

Además de los atributos que describen los aeropuertos, la capa airports tiene otro campo `fk_region` que actúa como clave externa (si tienes una base de datos, probablemente querrás definir una restricción sobre él). Este campo `fk_region` siempre contendrá el id de una región. Puede verse como un puntero a la región a la que pertenece.

Todo lo que tienes que hacer es decirle a QGIS la relación entre las capas para que puedas diseñar un formulario de edición personalizado para la edición y QGIS se encarga de la configuración. Funciona con diferentes proveedores (por lo que también puedes usarlo con archivos shape y csv).

Definir relaciones 1-N

La primera cosa que vamos a hacer es informar a QGIS de las relaciones entre las capas. Esto se hace en *Proyecto* ► *Propiedades…*. Abre la pestaña *Relaciones* y haz click en *Agregar Relación*.

- **Name** se utilizará como título. Debe ser una cadena legible que describa para qué se utiliza la relación. En este caso, diremos **airport_relation**.
- **Capa de Referencia (Parental)** también considerada como capa padre, es la única con la clave primaria, a la que se señala, así que aquí la capa es `regions`. Debe definir la clave primaria de la capa a la que se hace referencia, así que es `ID`.
- **Capa Referenciada (Hija)** también considerada como capa hija, es la única con la clave externa en ella. En nuestro caso, es la capa `airports`. Para esta capa, debe agregar un campo de referencia que apunte a la otra capa, por lo que es `fk_region`.

> **Nota**
>
> A veces se necesita más de un campo para identificar de forma exclusiva los objetos espaciales de una capa. La creación de una relación con una capa de este tipo requiere una **clave compuesta**, es decir, más de un único par de campos coincidentes. Utilice el botón Añadir un nuevo par de campos como parte de una clave externa compuesta para añadir tantos pares como sean necesarios.

- **Id** se utilizará para fines internos y debe ser único. Puede que lo necesite para construir *formularios personalizados*. Si lo deja vacío, se generará uno para usted, pero puede asignar uno usted mismo para obtener uno que sea más fácil de manejar.
- **Intensidad de la relación** establece la intensidad de la relación entre la capa principal y la secundaria. El tipo *Association* por defecto significa que la capa principal está *simplemente* vinculada a la secundaria, mientras que el tipo *Composition* le permite duplicar también las funciones secundarias al duplicar las principales y al eliminar una función los hijos también se eliminan, lo que da como resultado una cascada en todos los niveles (significa que los hijos de los hijos de… también se eliminan).

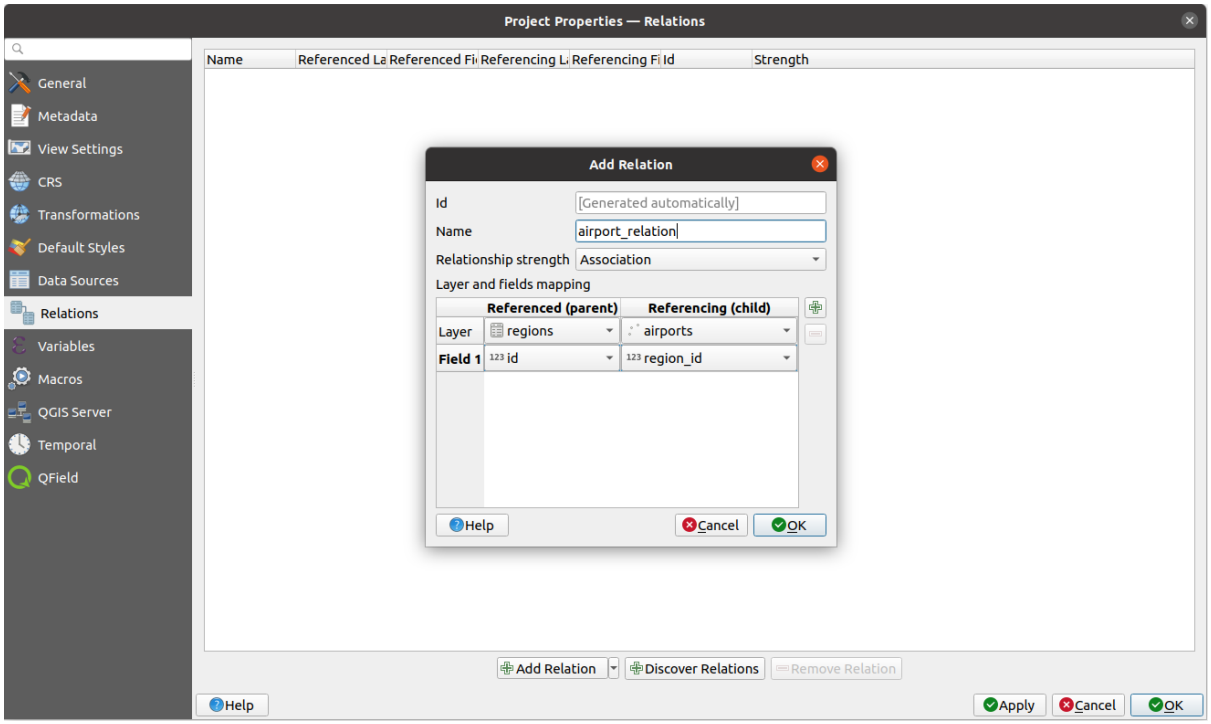


Figura 12.107: Agregar una relación entre las capas regions y airports

En la pestaña *Relaciones*, puede también presionar el botón *Descubrir relación* para obtener las relaciones disponibles de los proveedores de las capas cargadas. Esto es posible para capas almacenadas en proveedores de datos como PostgreSQL o SpatiaLite.

Formularios para relaciones 1-N

Ahora que QGIS conoce la relación, la utilizará para mejorar los formularios que genera. Como no hemos cambiado el método de formulario predeterminado (autogenerado), simplemente añadirá un nuevo widget en nuestro formulario. Así que seleccionemos la región de la capa en la leyenda y utilicemos la herramienta identificar. Dependiendo de su configuración, el formulario podría abrirse directamente o tendrá que elegir abrirlo en el diálogo de identificación bajo acciones.

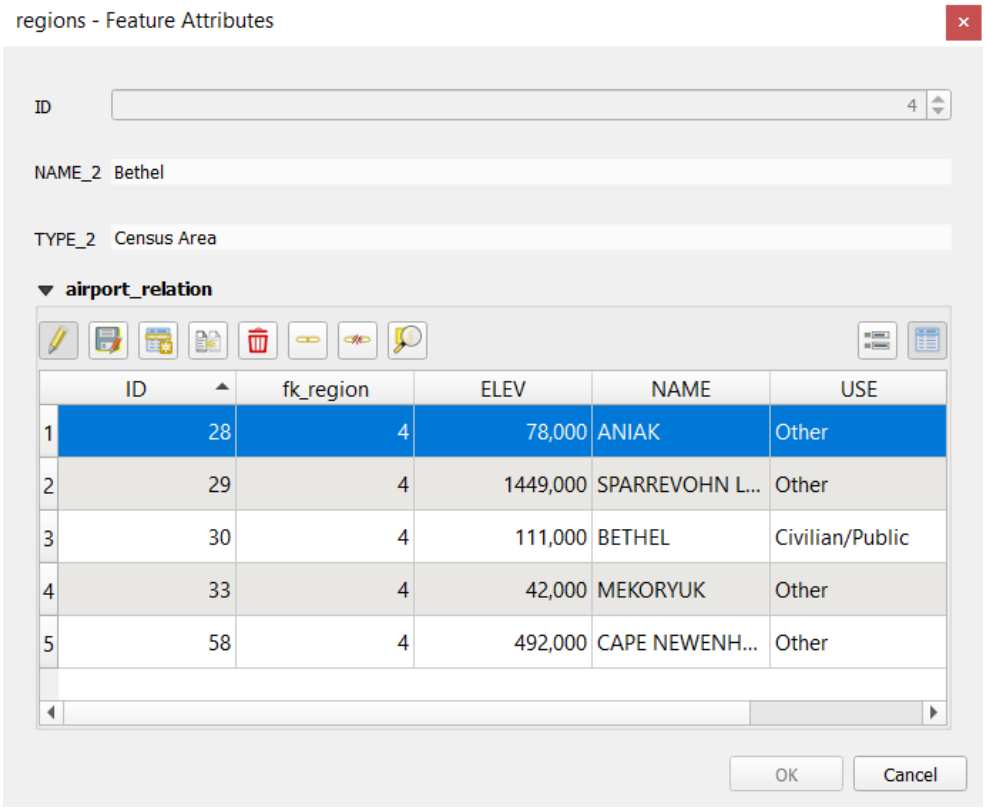


Figura 12.108: Regiones de diálogo de identificación con relación a aeropuertos

Como puede ver, los aeropuertos asignados a esta región en particular se muestran en una tabla. Y también hay algunos botones disponibles. Repasemos en breve:

- El botón es para alternar el modo de edición. Tenga en cuenta que cambia el modo de edición de la capa de aeropuerto, aunque estamos en la forma de entidad de una entidad de la capa de región. Pero la tabla representa entidades de la capa del aeropuerto.
- El botón es para guardar todas las ediciones en la capa secundaria (aeropuerto).
- El botón permite digitalizar la geometría del aeropuerto en el lienzo del mapa y asigna el nuevo objeto espacial a la región actual predeterminada. Tenga en cuenta que el icono cambiará según el tipo de geometría.
- El botón agrega un nuevo registro a la tabla de atributos de la capa del aeropuerto y asigna la nueva entidad a la región actual de forma predeterminada. La geometría se puede dibujar posteriormente con la herramienta de digitalización *Añadir pieza*.
- El botón permite copiar y pegar una o más entidades secundarias dentro de la capa secundaria. Posteriormente, se pueden asignar a una función principal diferente o se pueden modificar sus atributos.
- El botón elimina los aeropuerto(s) seleccionados de forma permanente
- El símbolo abre un nuevo cuadro de diálogo donde puede seleccionar cualquier aeropuerto existente que luego se asignará a la región actual. Esto puede ser útil si creó el aeropuerto en la región equivocada por accidente.
- El símbolo desvincula los aeropuerto(s) seleccionados de la región actual, dejándolos sin asignar (la clave externa se establece en NULL) de manera efectiva.
- Con el botón puede acercar el mapa a las entidades secundarias seleccionadas.

- Los dos botones y a la derecha cambia entre la *table view and form view* de las funciones secundarias relacionadas.

Si utiliza el *Diseñador de arrastrar y soltar* para el objeto espacial de las regiones, puede seleccionar qué herramientas están disponibles. Incluso puede decidir si se abre un nuevo formulario cuando se añade un nuevo objeto espacial utilizando la opción *Forzar la ocultación del formulario en la función de añadir*. Tenga en cuenta que esta opción implica que los atributos no nulos deben tomar un valor válido por defecto para funcionar correctamente.

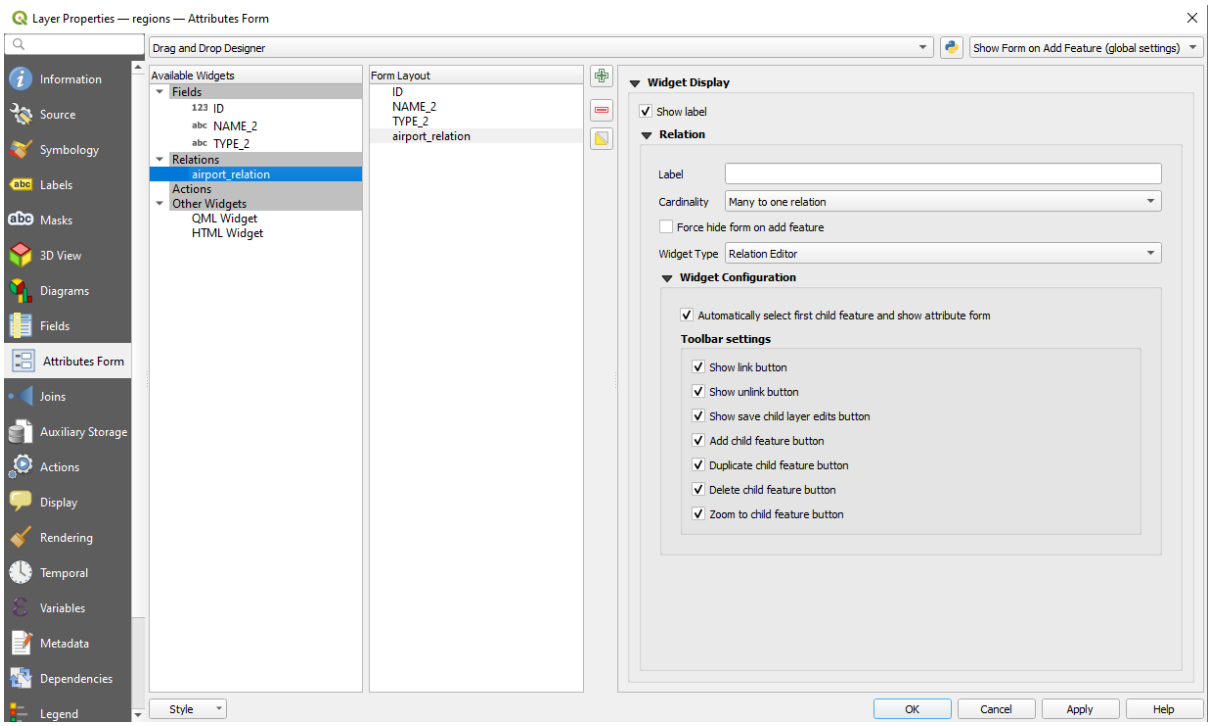


Figura 12.109: Diseñador de arrastrar y soltar para configurar las herramientas de relación entre regiones y aeropuertos

En el ejemplo anterior, la capa de referencia tiene geometrías (por lo que no es solo una tabla alfanumérica), por lo que los pasos anteriores crearán una entrada en la tabla de atributos de capa que no tiene una característica geométrica correspondiente. Para agregar la geometría:

1. Elige *Abrir Tabla de Atributos* para la capa de referencia.
2. Selecciona el registro que se ha agregado previamente dentro de la forma de entidad de la capa referenciada.
3. Usa la herramienta de digitalización Añadir parte para adjuntar una geometría al registro de la tabla de atributos seleccionado.

Si trabajas en la tabla del aeropuerto, el widget Relation Reference se configura automáticamente para el campo `fk_region` (el que se usa para crear la relación), consulta *Relation Reference widget*.

En el formulario del aeropuerto verás el en el lado derecho del campo `fk_region`: si haces click en el botón, se abrirá el formulario de la capa de la región. Este widget te permite abrir fácil y rápidamente los formularios de las funciones principales vinculadas.

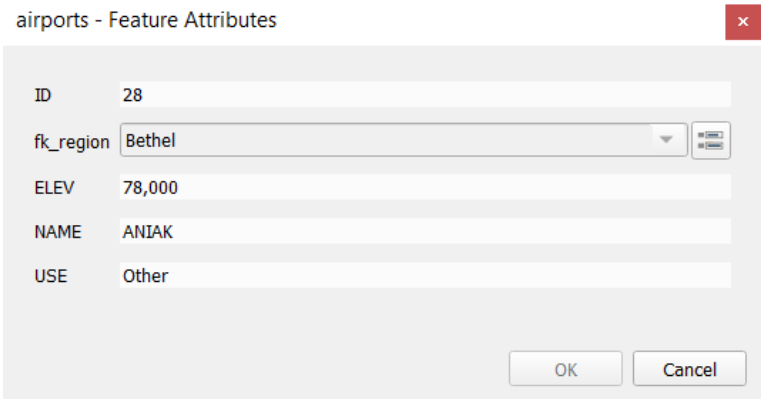
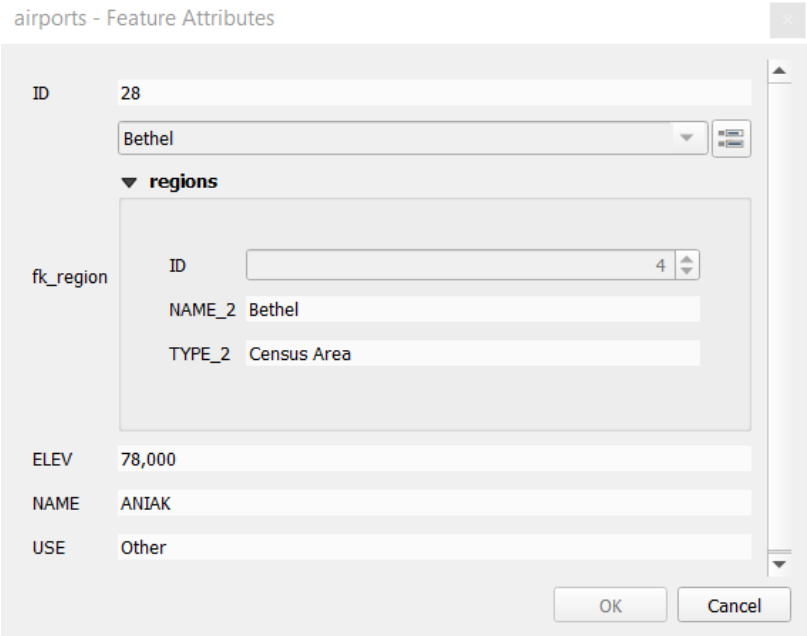


Figura 12.110: Diálogo de identificación de aeropuerto con relación a las regiones

El widget Relation Reference también tiene una opción para incrustar la forma de la capa principal dentro de la secundaria. Está disponible en el menú *Propiedades* ► *Formulario de atributos* de la capa del aeropuerto: seleccione el campo `fk_region` y marca la opción `Mostrar formulario incrustado`.

Si mira ahora el cuadro de diálogo del objeto espacial, verá que el formulario de la región está incrustado dentro del formulario de los aeropuertos e incluso tendrá un combobox, que le permitirá asignar el aeropuerto actual a otra región.



Además, si cambias el modo de edición de la capa del aeropuerto, el campo `fk_region` también tiene una función de autocompletar: mientras escribes, verás todos los valores del campo `` id "" de la capa de la región. Aquí es posible digitalizar un polígono para la capa de región usando el botón  si elegiste la opción `Permitir agregar nuevas entidades` en la *Propiedades* -> *Formulario de atributos* de la capa del aeropuerto.

La capa secundaria también se puede utilizar en la herramienta *Seleccionar objetos por valor* para seleccionar entidades de la capa principal según los atributos de sus hijos.

En Figura 12.111, se seleccionan todas las regiones donde la altitud media de los aeropuertos es superior a 500 metros sobre el nivel del mar.

Encontrarás que hay muchas funciones de agregación diferentes disponibles en el formulario.

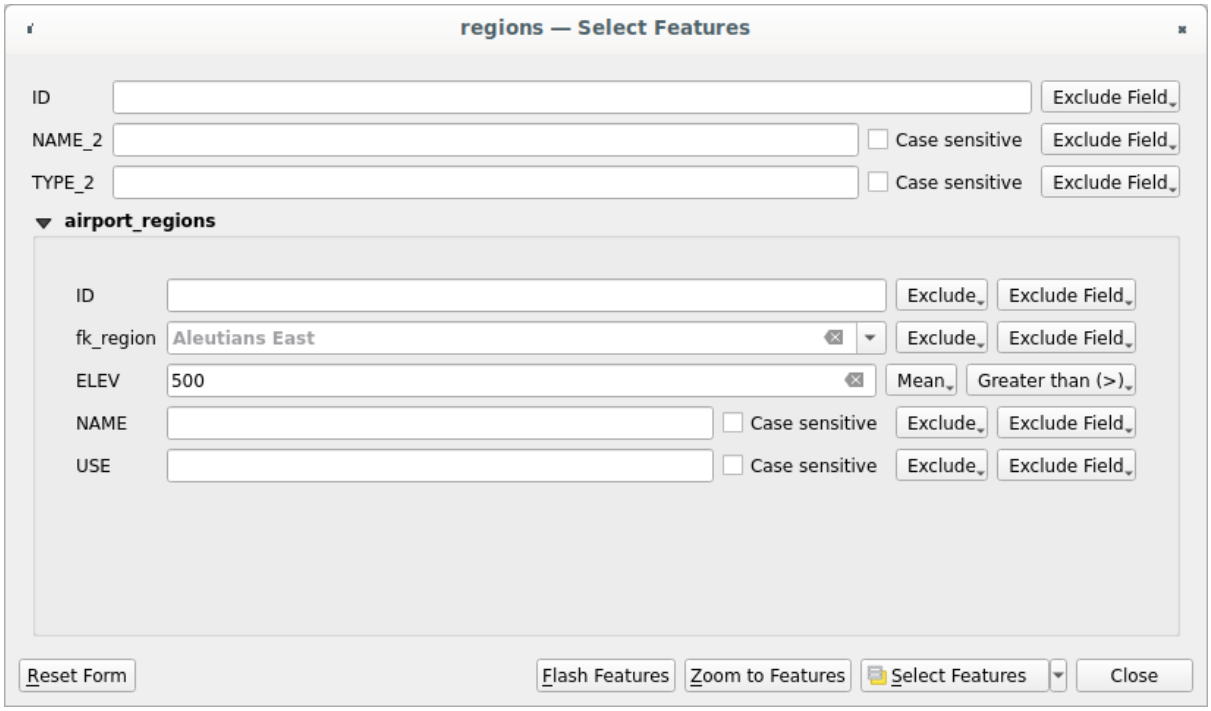


Figura 12.111: Seleccionar entidades principales con valores secundarios

Relaciones de muchos a muchos (N-M)

Las relaciones N-M son relaciones de muchos a muchos entre dos tablas. Por ejemplo, las capas de `airports` y `airlines`: un aeropuerto recibe varias compañías aéreas y una compañía aérea vuela a varios aeropuertos.

This SQL code creates the three tables we need for an N-M relationship in a PostgreSQL schema named *locations*. You can run the code using the *Database* ► *DB Manager*… for PostgreSQL or external tools such as pgAdmin. The airports table stores the `airports` layer and the airline table stores the `airlines` layer. In both tables few fields are used for clarity. The *tricky* part is the `airports_airlines` table. We need it to list all airlines for all airports (or vice versa). This kind of table is known as a *pivot table*. The *constraints* in this table force that an airport can be associated with an airline only if both already exist in their layers.

```
CREATE SCHEMA locations;

CREATE TABLE locations.airports
(
   id serial NOT NULL,
   geom geometry(Point, 4326) NOT NULL,
   airport_name text NOT NULL,
   CONSTRAINT airports_pkey PRIMARY KEY (id)
);

CREATE INDEX airports_geom_idx ON locations.airports USING gist (geom);

CREATE TABLE locations.airlines
(
   id serial NOT NULL,
   geom geometry(Point, 4326) NOT NULL,
   airline_name text NOT NULL,
   CONSTRAINT airlines_pkey PRIMARY KEY (id)
);

CREATE INDEX airlines_geom_idx ON locations.airlines USING gist (geom);
```

(continúe en la próxima página)

(proviene de la página anterior)

```
CREATE TABLE locations.airports_airlines
(
   id serial NOT NULL,
   airport_fk integer NOT NULL,
   airline_fk integer NOT NULL,
   CONSTRAINT airports_airlines_pkey PRIMARY KEY (id),
   CONSTRAINT airports_airlines_airport_fk_fkey FOREIGN KEY (airport_fk)
      REFERENCES locations.airports (id)
      ON DELETE CASCADE
      ON UPDATE CASCADE
      DEFERRABLE INITIALLY DEFERRED,
   CONSTRAINT airports_airlines_airline_fk_fkey FOREIGN KEY (airline_fk)
      REFERENCES locations.airlines (id)
      ON DELETE CASCADE
      ON UPDATE CASCADE
      DEFERRABLE INITIALLY DEFERRED
 );
```

En lugar de PostgreSQL, también puedes utilizar GeoPackage. En este caso, las tres tablas se pueden crear manualmente usando *Base de datos* ► *Administrador de Bases de datos…*. En GeoPackage no hay esquemas, por lo que el prefijo *locations* no es necesario.

Las restricciones de clave externa en la tabla `` Airports_airlines`` no se pueden crear usando *Tabla* ► *Crear Tabla…* o *Tabla* ► *Editar Tabla…* por lo que deben crearse usando *Base de datos* ► *Ventana SQL…*. GeoPackage no admite declaraciones *ADD CONSTRAINT*, por lo que la tabla `airports_airlines` debe crearse en dos pasos:

1. Configura la tabla solo con el campo `id` usando *Tabla* ► *Crear tabla …*
2. Usando *Base de Datos* ► *Ventana SQL…*, escribe y ejecuta este código SQL:

```
ALTER TABLE airports_airlines
   ADD COLUMN airport_fk INTEGER
   REFERENCES airports (id)
   ON DELETE CASCADE
   ON UPDATE CASCADE
   DEFERRABLE INITIALLY DEFERRED;

ALTER TABLE airports_airlines
   ADD COLUMN airline_fk INTEGER
   REFERENCES airlines (id)
   ON DELETE CASCADE
   ON UPDATE CASCADE
   DEFERRABLE INITIALLY DEFERRED;
```

Luego, en QGIS, debe configurar dos *una-a-muchas relaciones* como se explicó anteriormente:

- una relación entre la tabla de `airlines` y la tabla dinámica;
- y una segunda entre la tabla `airports` y la tabla dinámica.

Una forma más fácil de hacerlo (solo para PostgreSQL) es usando *Descubrir Relaciones* en *Proyecto* ► *Propiedades* ► *Relacione*. QGIS leerá automáticamente todas las relaciones en su base de datos y solo tiene que seleccionar las dos que necesita. Recuerde cargar primero las tres tablas en el proyecto QGIS.

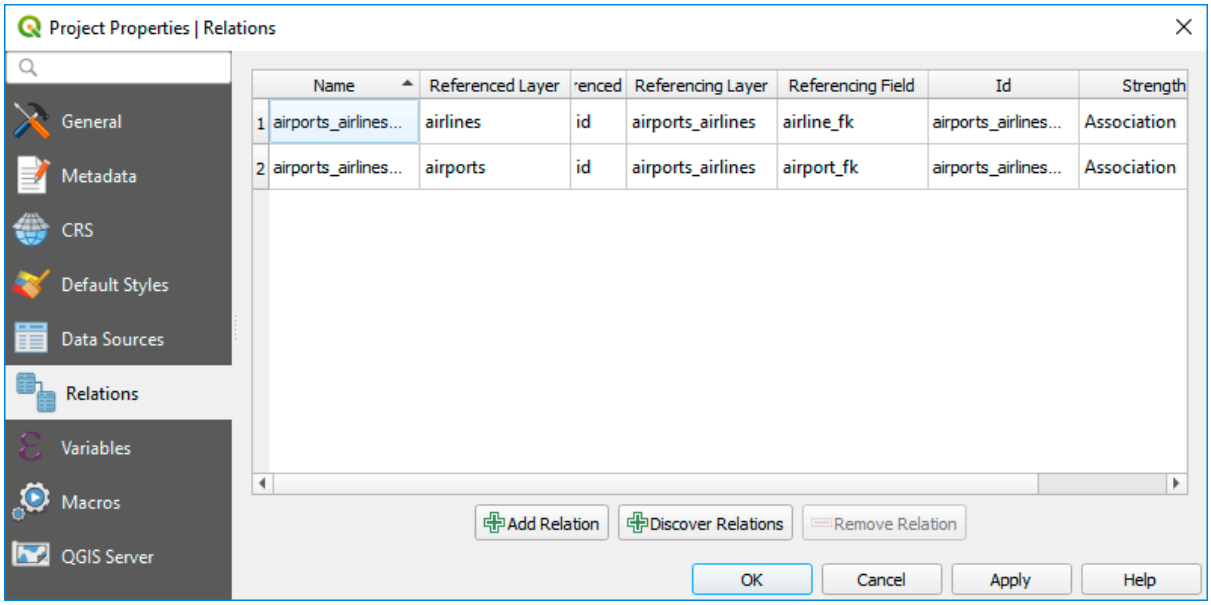


Figura 12.112: Relaciones y descubrimiento automático

En caso de que desee eliminar un `` airport`` o una `airline`, QGIS no eliminará los registros asociados en la tabla `Airports_airlines`. Esta tarea la realizará la base de datos si especificamos las * restricciones * correctas en la creación de la tabla dinámica como en el ejemplo actual.

Nota

** Combinando relación N-M con grupo de transacciones automáticas **

Debes habilitar el modo de transacción en *Propiedades del proyecto* ► *Fuentes de datos* ► cuando trabajes en dicho contexto. QGIS debería poder agregar o actualizar fila(s) en todas las tablas (airlines, airports y las tablas dinámicas).

Finalmente tenemos que seleccionar la cardinalidad correcta en *Layer Properties* ► *Attributes Form* para las capas `airports` y `airlines`. Para el primero debemos elegir la opción **aerolíneas (id)** y para el segundo la opción **aeropuertos (id)**.

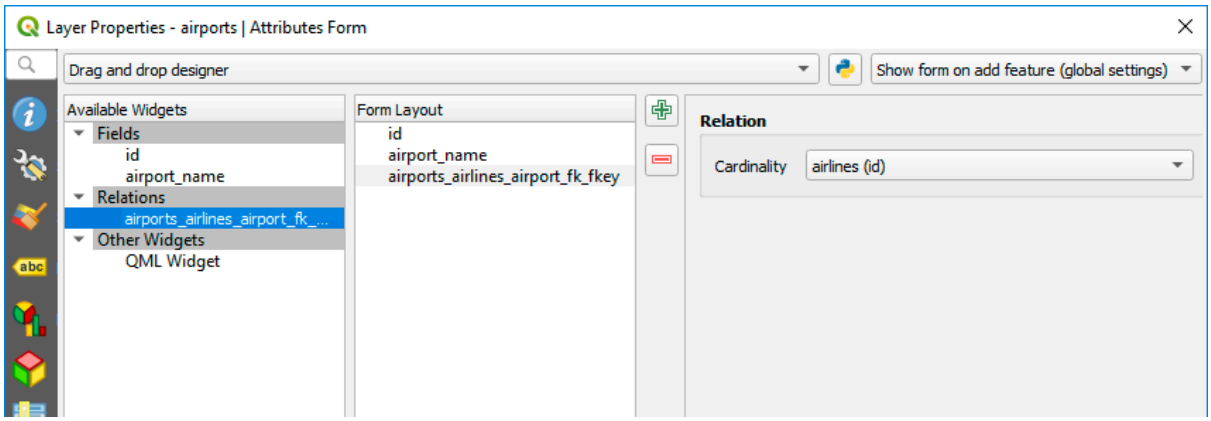


Figura 12.113: Establecer cardinalidad de relación

Ahora puede asociar un airport con una airline (o una airline con un airport) usando *Agregar función secundaria* o *Vincular función secundaria existente* en los subformularios. Se insertará automáticamente un registro en la tabla `Airports_airlines`.

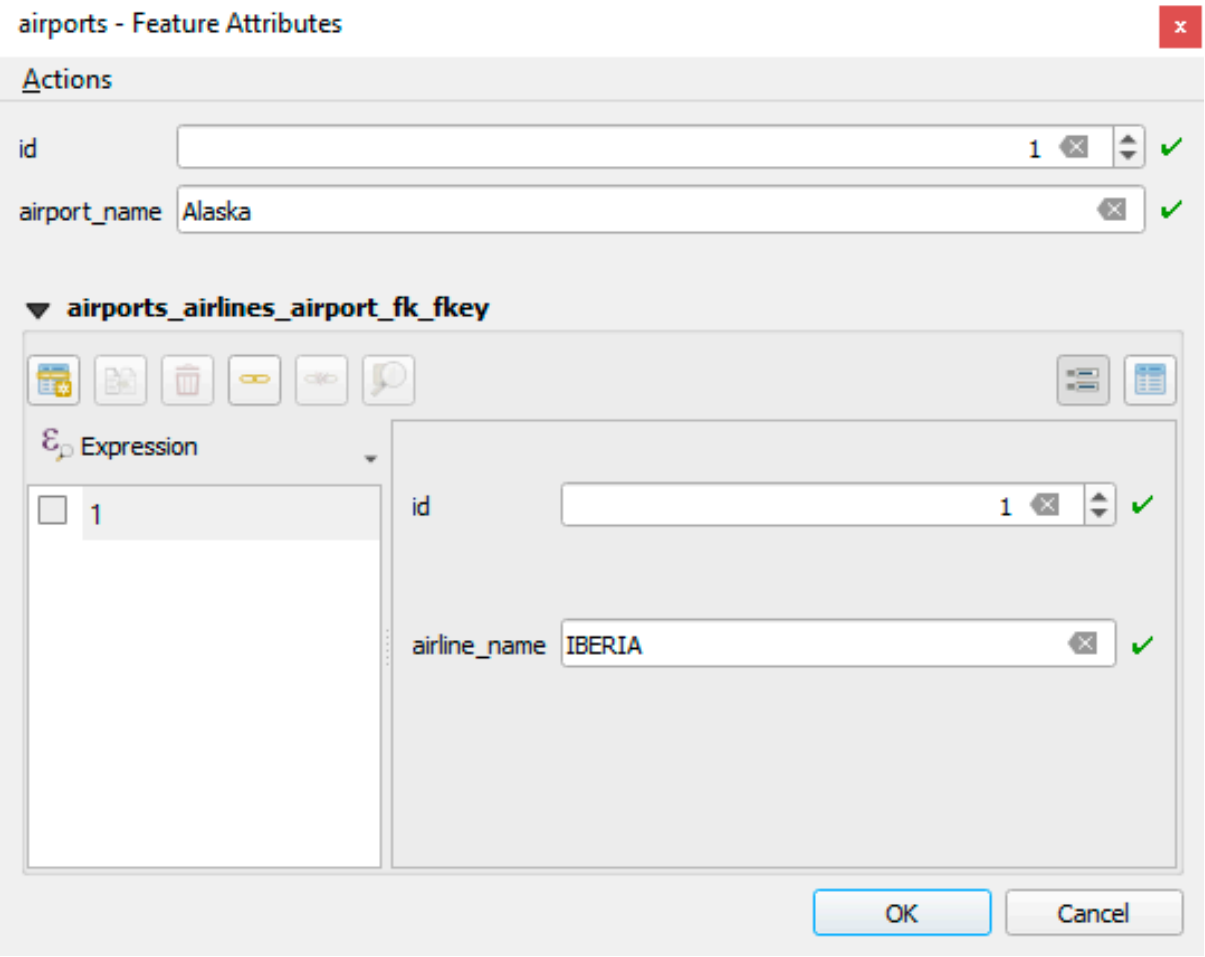


Figura 12.114: Relación N-M entre aeropuertos y aerolíneas

Nota

Uso de cardinalidad ** relación muchos a uno **

A veces, ocultar la tabla dinámica en una relación N-M no es deseable. Principalmente porque hay atributos en la relación que sólo pueden tener valores cuando se establece una relación. Si sus tablas tienen un campo de geometría, podría ser interesante activar la opción *Identificación en mapa* (*Propiedades de capa* ► *Formulario de atributos* ► *Widgets disponibles* ► *Campos*) para los campos de clave externa en la tabla pivote.

Nota

** Clave principal de la tabla dinámica **

Evite utilizar múltiples campos en la clave primaria de una tabla pivotante. QGIS asume una única clave primaria por lo que una restricción como `constraint airports_airlines_pkey primary key (airport_fk, airline_fk)` no funcionará.

Relaciones polimórficas

El propósito

Las relaciones polimórficas son un caso especial de las relaciones 1-N, donde una única capa de referencia (documento) contiene las características de varias capas de referencia. Esto difiere de las relaciones normales que requieren una capa de referencia diferente para cada capa referenciada. Se logra una sola capa de referencia (documento) agregando una columna `layer_field` adicional en la capa de referencia (documento) que almacena información para identificar la capa referenciada. En su forma más simple, la capa de referencia (documento) simplemente insertará el nombre de la capa a la que se hace referencia en este campo.

Para ser más precisos, una relación polimórfica es un conjunto de relaciones normales que tienen la misma capa de referencia pero que tienen la capa de referencia definida dinámicamente. La configuración polimórfica de la capa se resuelve mediante el uso de una expresión que debe coincidir con algunas propiedades de la capa a la que se hace referencia, como el nombre de la tabla, la identificación de la capa, el nombre de la capa.

Imagina que vamos al parque y queremos hacer fotos de diferentes especies de `plantas` y `animales` que vemos allí. Cada planta o animal tiene varias imágenes asociadas, por lo que si usamos las relaciones 1:N normales para almacenar imágenes, necesitaríamos dos tablas separadas, `animal_images` y `plant_images`. Esto podría no ser un problema para 2 mesas, pero imagina si queremos tomar fotos separadas para hongos, pájaros, etc.

Las relaciones polimórficas resuelven este problema, ya que todos los objetos espaciales referenciados se almacenan en la misma tabla `documents`. Para cada objeto espacial, la capa referenciada se almacena en el campo `referenced_layer` y el id del objeto espacial referenciado en el campo `referenced_id`.

Definición de relaciones polimórficas

Primero, informe a QGIS sobre las relaciones polimórficas entre las capas. Esto se hace en *Proyecto* ► *Propiedades…*. Abra la pestaña *Relaciones* y haga clic en la pequeña flecha hacia abajo junto a symbologyAdd| *Agregar relación*, para que pueda seleccionar la opción *Agregar relación polimórfica* del menú desplegable recién aparecido.

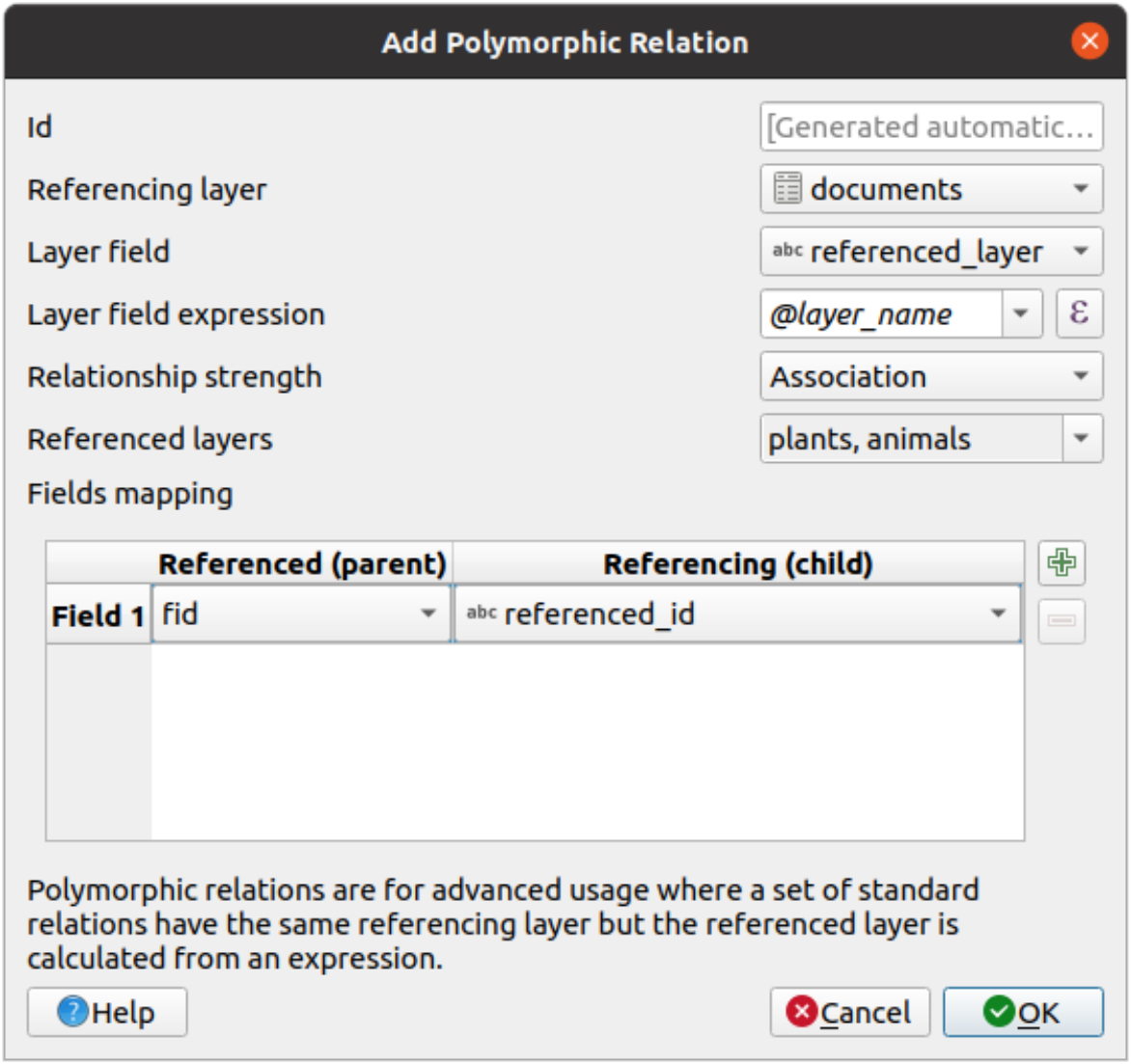


Figura 12.115: Agregar una relación polimórfica usando la capa `documents` como referencia y `animals` y `plantas` como capas referenciadas.

- **Id** se utilizará para fines internos y debe ser único. Puede que lo necesite para construir *formularios personalizados*. Si lo deja vacío, se generará uno para usted, pero puede asignar uno usted mismo para obtener uno que sea más fácil de manejar.
- **Capa de referencia (hija)** también considerada como capa hija, es la que tiene el campo de clave foránea en ella. En nuestro caso, es la capa `documents`. Para esta capa es necesario añadir un campo de referencia que apunta a la otra capa, por lo que este es `referenced_id`.

  Nota

  A veces se necesita más de un campo para identificar de forma exclusiva los objetos espaciales de una capa. La creación de una relación con una capa de este tipo requiere una **clave compuesta**, es decir, más de un único par de campos coincidentes. Utilice el botón Añadir un nuevo par de campos como parte de una clave externa compuesta para añadir tantos pares como sean necesarios.

- **Campo de capa** es el campo de la tabla de referencia que almacena el resultado de la expresión de capa evaluada, que es la tabla de referencia a la que pertenece esta entidad. En nuestro ejemplo, este sería el campo `referenced_layer`.

- **Expresión de capa** se evalúa como un identificador único de la capa. Este puede ser el nombre de la capa `@layer_name`, la identificación de la capa `@layer_id`, el nombre de la tabla de la capa `decode_uri(@layer, 'table')` o cualquier cosa que pueda identificar de manera única una capa.
- **Intensidad de la relación** establece la intensidad de las relaciones generadas entre la capa principal y la secundaria. El tipo *Asociación* por defecto significa que la capa principal está *simplemente* vinculada a la secundaria, mientras que el tipo *Composición* le permite duplicar también las funciones secundarias al duplicar las principales y al eliminar una función los hijos también se eliminan, lo que da como resultado una cascada en todos los niveles (significa que los hijos de los hijos de… también se eliminan).
- **Capas referenciadas** también consideradas como capas padre, son aquellas que tienen la clave primaria apuntada, por lo que aquí serían capas `plantas` y `animales`. Debe definir la clave principal de las capas a las que se hace referencia desde el menú desplegable, por lo que es `fid`. Tenga en cuenta que la definición de una clave primaria válida requiere que todas las capas referenciadas tengan un campo con ese nombre. Si no existe tal campo, no puede guardar una relación polimórfica.

Una vez añadida, la relación polimórfica se puede editar a través de la entrada de menú *Editar relación polimórfica*.

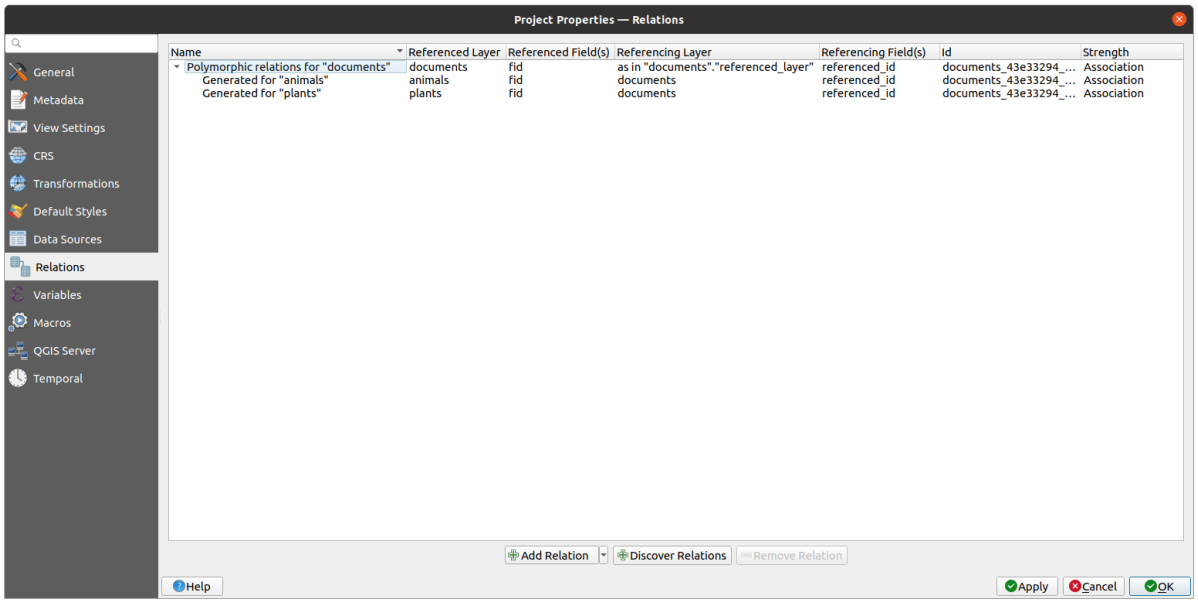


Figura 12.116: Vista previa de la relación polimórfica recién creada y sus relaciones hijas para animales y plantas.

El ejemplo anterior utiliza el siguiente esquema de base de datos:

```
CREATE SCHEMA park;

CREATE TABLE park.animals
(
   fid serial NOT NULL,
   geom geometry(Point, 4326) NOT NULL,
   animal_species text NOT NULL,
   CONSTRAINT animals_pkey PRIMARY KEY (fid)
);

CREATE INDEX animals_geom_idx ON park.animals USING gist (geom);

CREATE TABLE park.plants
(
   fid serial NOT NULL,
   geom geometry(Point, 4326) NOT NULL,
   plant_species text NOT NULL,
   CONSTRAINT plants_pkey PRIMARY KEY (fid)
```

(continúe en la próxima página)

(proviene de la página anterior)

```
);

CREATE INDEX plants_geom_idx ON park.plants USING gist (geom);

CREATE TABLE park.documents
(
   fid serial NOT NULL,
   referenced_layer text NOT NULL,
   referenced_id integer NOT NULL,
   image_filename text NOT NULL,
   CONSTRAINT documents_pkey PRIMARY KEY (fid)
);
```

Once relations are established, the *Identify Features* result panel will display related features, allowing you to view associated data directly.

Dataset-stored Relationships

Other than the above mentioned relationships that are created in the project properties and apply only in the active project, QGIS can also read database-level relations, i.e., connections defined between layers in the dataset. This is supported for database providers such as PostgreSQL, SpatiaLite, GeoPackage, ESRI FileGeodatabase. When such parent and child layers are loaded in a project, QGIS can automatically detect the relation settings and fill appropriately the *Project* ► *Project properties* ► *Relations* tab. This can also be triggered using the *Discover relations* button.

For some data types (e.g., GeoPackage, SpatiaLite or ESRI FileGeodatabase), QGIS also allows to create and edit the relations between the tables in the database. This can be managed through the *Browser Panel*. From the *Browser panel*, right-click on the database entry, select *New relationship…*, and in the new dialog, fill the following options:

- *Name*: set the name of the new relationship.
- *Cardinality*: define the cardinality of the relationship.
- *Strength*: specify the strength of the relationship.
- Under the *Tables* group, set your *Parent* and *Child* tables.
- Under the *Fields* group, set your *Parent field* and *Child field*.
- Under the *Advanced* section, depending on the database type, you can set:
  - *Related table type*:
    - *features*: for standard feature tables.
    - *media*: for tables storing media files.
    - *simple_attributes*: for tables with simple attribute data.
    - *attributes*: for tables with complex attribute data.
    - *tiles*: for tables storing tile data.
  - *Forward label*: a name to uniquely identify the relationship when navigating from the parent table to the child table.
  - *Reverse label*: a name to uniquely identify the relationship when navigating from the child table to the parent table.

Applying the dialog will add a *Relationships* node under the database entry, with the new relationship as child item.

Editing or deleting existing relationships can be done by right-clicking on the relationship in the Browser Panel.

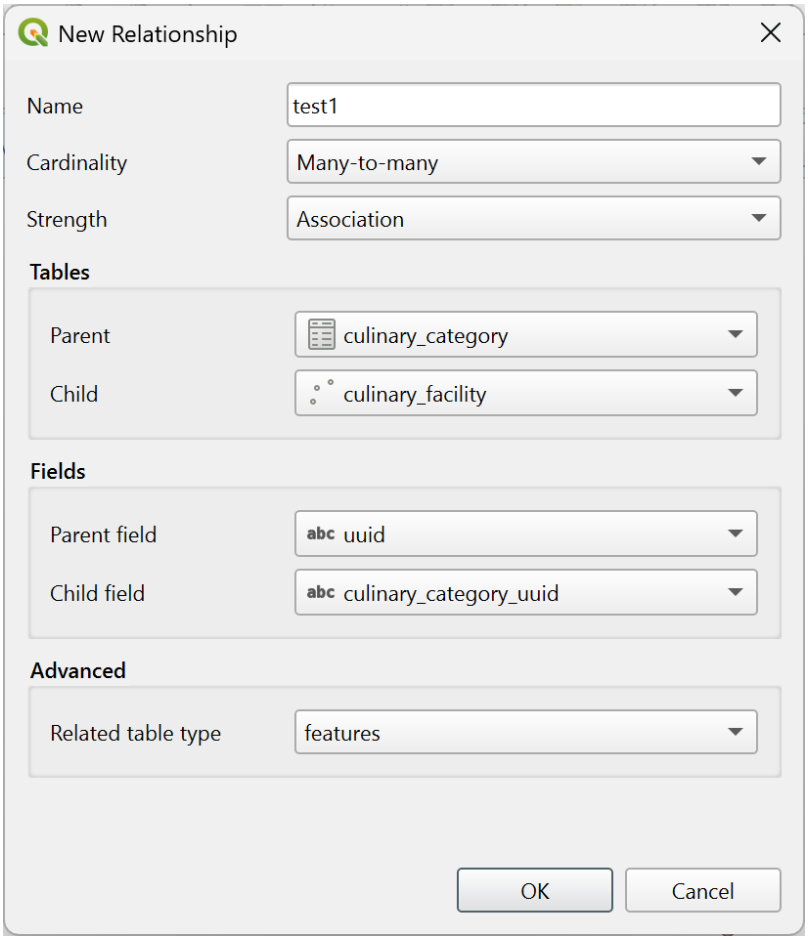


Figura 12.117: Managing dataset-stored relationships in the Browser Panel

12.4.3 Storing and fetching an external resource

A field may target a resource stored on an external storage system. Attribute forms can be configured so they act as a client to an external storage system in order to store and fetch those resources, on users demand, directly from the forms.

Configuring an external storage

In order to setup an external storage, you have to first configure it from the vector *attribute form properties* and select the *Attachment* widget.



Figura 12.118: Editing a WebDAV external storage for a given field

From the *Attachment* widget, you have to first select the *Storage type*:

- *Select Existing File*: The target URL already exists. When you select a resource, no store operation is achieved, the attribute is simply updated with the URL.
- *Simple Copy*: Stores a copy of the resource on a file disk destination (which could be a local or network shared file system) and the attribute is updated with the path to the copy.
- *WebDAV Storage*: The resource is pushed to a HTTP server supporting the WebDAV protocol and the attribute is updated with its URL. Nextcloud, Pydio or other file hosting software support this protocol.
- *AWS S3*: The resource is pushed to a server supporting AWS Simple Storage Service protocol and the attribute is updated with its URL. Amazon Web Service and MinIO hosting software support this protocol.

Then, you have to set up the *Store URL* parameter, which provides the URL to be used when a new resource needs to be stored. It's possible to set up an expression using the *data defined override widget* in order to have specific values according to feature attributes.

The variable **@selected_file_path** could be used in that expression and represent the absolute file path of the user selected file (using the file selector or drag'n drop).

Nota

Using the **WebDAV** or **AWS S3** external storage, if the URL ends with a «/», it is considered as a folder and the selected file name will be appended to get the final URL.

If the external storage system needs to, it's possible to configure an *authentication*.

Nota

To use the **AWS S3** external storage, you must use an **AWS S3** authentication type.

Using an external storage

Once configured, you can select a local file using the button … when editing a feature's attribute. Depending on the configured *storage type*, the file will be stored on the external storage system (except if *Select existing file* has been selected) and the field will be updated with the new resource URL.

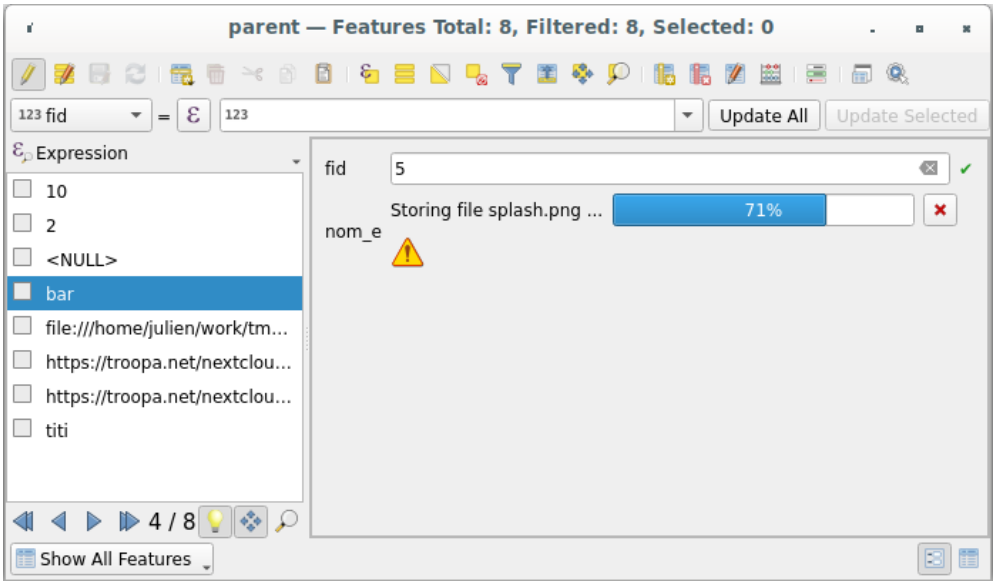


Figura 12.119: Storing a file to a WebDAV external storage

Nota

User can also achieve the same result if he drags and drops a file on the whole attachment widget.

Use the Cancel button to abort the storing process. It's possible to configure a viewer using the *Integrated document viewer* so the resource will be automatically fetched from the external storage system and displayed directly below the URL. The above icon indicates that the resource cannot be fetched from the external storage system. In that case, more details might appear in the *Panel de mensajes de registro*.

CAPÍTULO 13

Trabajar con Datos Raster

13.1 Dialogo de Propiedades Ráster

Los datos ráster están formados por píxeles (o celdas), y cada píxel tiene un valor. Se suelen utilizar para almacenar varios tipos de datos, entre ellos:

- Imágenes, como imágenes por satélite, fotografías aéreas digitales, mapas escaneados
- Datos de elevación, como modelos digitales de elevación (MDE), modelos digitales del terreno (MDT)
- Otros tipos de datos, como la cubierta terrestre, los tipos de suelo, las precipitaciones y muchos otros.

Raster data can be stored in several supported formats, including GeoTIFF, ERDAS Imagine, ArcInfo ASCII GRID, PostgreSQL Raster and others. See more at *Abriendo Datos*.

Para ver y establecer las propiedades de una capa ráster, haga doble clic en el nombre de la capa en la leyenda del mapa, o haga clic con el botón derecho en el nombre de la capa y elija *Propiedades* en el menú contextual. Esto abrirá el diálogo *Propiedades de la capa ráster*.

Hay varias pestañas en el cuadro de diálogo:

Información	*Fuente*	*Symbology* ([1])
Transparency ([1])	*Labels* ([1])	*Histogram* ([1])
Representación	*Temporal*	*Piramides*
Elevation	*Metadatos*	*Legend*
Mostrar	*Attribute Tables*	*Servidor de QGIS*
External plugins ([2]) tabs		

[1] Also available in the *Layer styling panel*
[2] *External plugins* you install can optionally add tabs to this dialog. Those are not presented in this document. Refer to their documentation.

> **Truco**
>
> **Renderizado de actualización en vivo**
>
> El *Panel de Estilizado de Capa* le proporciona algunas de las prestaciones comunes del diálogo de propiedades de la capa y es un buen widget sin modo que puede usar para acelerar la configuración de los estilos de capa y ver sus cambios en el lienzo del mapa.

> **Nota**
>
> Debido a que las propiedades (simbología, etiqueta, acciones, valores predeterminados, formas …) de las capas incrustadas (ver *Incorporación de capas de proyectos externos*) se extraen del archivo del proyecto original, y para evitar cambios que puedan romper este comportamiento, las propiedades de la capa El diálogo no está disponible para estas capas.

13.1.1 Propiedades de información

La pestaña *Información* es de solo lectura y representa un lugar interesante para obtener rápidamente información resumida y metadatos para la capa actual. La información proporcionada es:

- general, como el nombre en el proyecto, la ruta de origen, la lista de archivos auxiliares, el tiempo y el tamaño del último guardado, el proveedor utilizado
- propiedades personalizadas, utilizadas para almacenar en el proyecto activo información adicional sobre la capa. Las propiedades personalizadas por defecto incluyen *Identify/format*, que influye en la forma en que los resultados del uso de la herramienta *Identificar objetos espaciales* sobre una capa raster se formatean. Se pueden crear y gestionar más propiedades utilizando PyQGIS, concretamente a través del método setCustomProperty().
- según el proveedor de la capa: extensión, ancho y alto, tipo de datos, controlador GDAL, estadísticas de bandas
- el sistema de referencia de coordenadas: nombre, unidades, método, precisión, referencia (es decir, si es estático o dinámico)
- leer de las propiedades de la capa: tipo de datos, extensión, ancho/alto, compresión, tamaño de píxel, estadísticas sobre bandas, número de columnas, filas y valores sin datos del ráster…
- seleccionado de los *filled metadata*: acceso, extensiones, enlaces, contactos, historial…

13.1.2 Propiedades de fuente

La pestaña *Fuente* muestra información básica sobre el ráster seleccionado, que incluye:

- el *Nombre de la capa* a mostrar en el *Panel de capas*;
- el *Sistema de referencia de coordenadas*: Muestra el :ref:` Sistema de referencia de coordenadas (CRS) ` de la capa. Puede cambiar el CRS de la capa, seleccionando uno usado recientemente en la lista desplegable o haciendo clic en el botón Seleccionar CRS (ver :ref:` crs_selector`). Utilice este proceso solo si la capa CRS es incorrecta o no está especificada. Si desea reproyectar sus datos, utilice un algoritmo de reproyección de Processing o Guárdelo como nuevo conjunto de datos.
- Depending on the data provider, a *Layer source* group indicates the path to the source of the dataset and allows for replacing the loaded layer:
  - When the layer is stored as file on disk, edit the path shown in the text box or press … Browse to select another file on the disk

- When the layer is provided by an ArcGIS MapServer service, it is possible to modify its *authentication settings*, while keeping unchanged the details for connecting to the service.
- For an XYZ remote layer, it is possible to modify at the layer level any of its *connection details* (URL, authentication settings, zoom levels, resolution, …). Modifications are applied to the layer without altering the original connection settings.

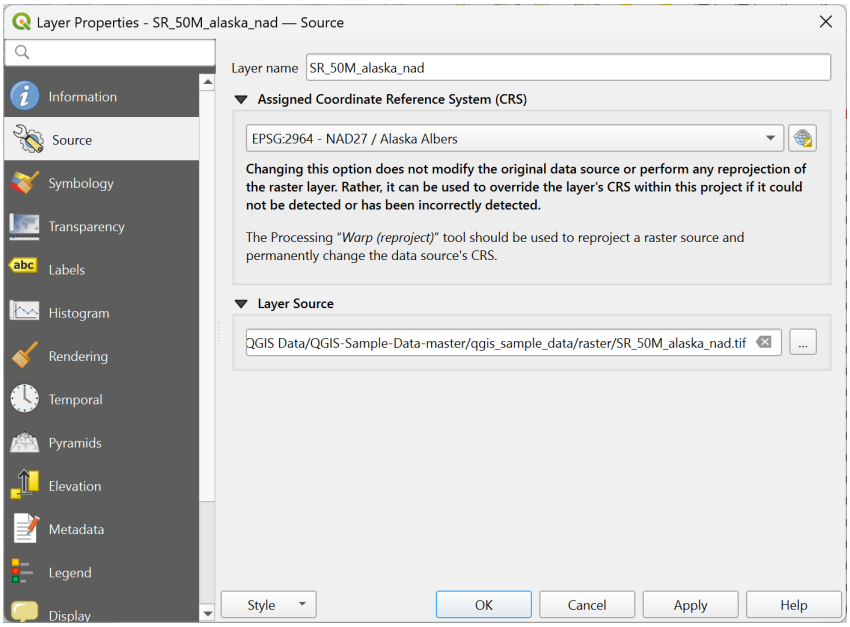


Figura 13.1: Propiedades de capa Ráster - Diálogo Fuente

13.1.3 Propiedades de simbología

La pestaña de simbología de la capa ráster se compone de tres secciones diferentes:

- El *Renderizado de banda* donde puedes controlar el tipo de renderizador a usar
- El *Layer rendering* para aplicar efectos sobre los datos renderizados
- Los métodos *remuestreo* para optimizar el renderizado en el mapa

Representación de banda

QGIS ofrece muchos *tipos de renderizado* diferentes. La elección del renderizador depende del tipo de datos y la información que quiera resaltar.

1. *Color Multibanda* - si el archivo viene con varias bandas (por ejemplo, una imagen de satélite con varias bandas).
2. *Valores de Paleta/Valores únicos* - para archivos de banda única que vienen con una paleta indexada (por ejemplo, un mapa topográfico digital) o para el uso general de paletas para renderizar capas ráster.
3. *Gris Monobanda* - (una banda de) la imagen se renderizará en gris. QGIS elegirá este renderizador si el archivo no es multibanda ni paletizado (por ejemplo, un mapa de relieve sombreado).
4. *Pseudocolor Monobanda* - este renderizador se puede utilizar para archivos con una paleta continua o un mapa de colores (por ejemplo, un mapa de elevación).
5. *Color único* - la capa de trama se renderizará con un solo color.
6. *Sombreado* - Crea sombreado a partir de una banda.
7. *Contornos* - Genera contornos al vuelo para una banda de fuente ráster.

Color Multibanda

Con el renderizador de color multibanda, tres bandas seleccionadas de la imagen se utilizarán como componente rojo, verde o azul de la imagen en color. QGIS obtiene automáticamente los valores *Min* y :guilabel:` Max` para cada banda del ráster y escala el color en consecuencia. Puede controlar los rangos de valores en la sección *Configuración de valores Min/Max*.

Se puede aplicar un método *mejora de contraste* a los valores: “Sin realce”, “Estirar a MinMax”, “Estirar y cortar a MinMax” y “Cortar a min max”.

Nota

Mejorar contraste

Al agregar rásters de GRASS, la opción *Mejora de contraste* siempre se establecerá automáticamente en *Estirar a MinMax*, incluso si se establece en otro valor en las opciones generales de QGIS.

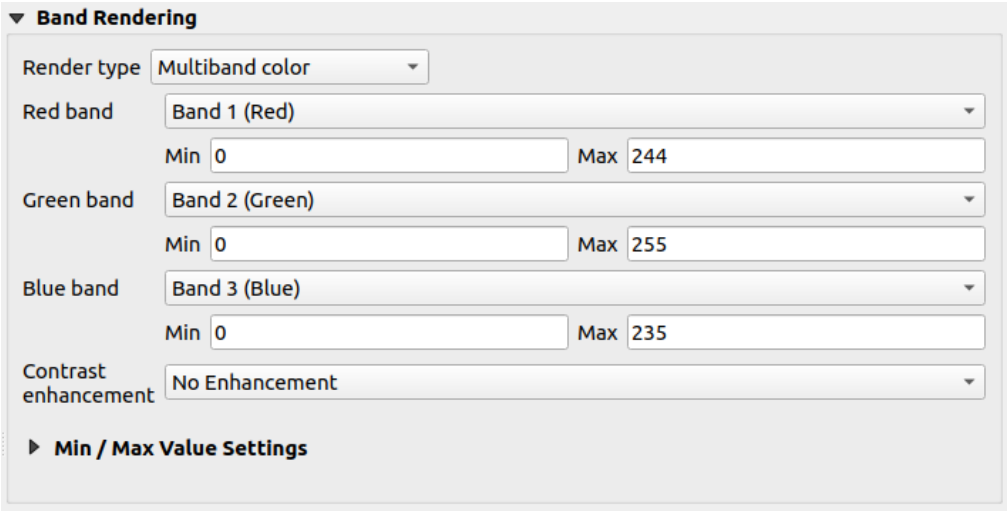


Figura 13.2: Simbología Ráster - renderizado de color Multibanda

Truco

Visualización de una sola banda de un ráster multibanda

Si desea ver una sola banda de una imagen multibanda (por ejemplo, roja), podría pensar que establecería las bandas verde y azul en *No configurado*. Pero la forma preferida de hacer esto es establecer el tipo de imagen en *Gris Monobanda*, y entonces seleccione Rojo como la *Banda Gris* a usar.

Valores en Paleta/valores únicos

Esta es la opción de procesamiento estándar para archivos de banda única que incluyen una tabla de colores, donde se asigna un color determinado a cada valor de píxel. En ese caso, la paleta se renderiza automáticamente.

Se puede utilizar para todo tipo de bandas de ráster, asignando un color a cada valor de ráster único.

Si desea cambiar un color, simplemente haga doble clic en el color y aparecerá el cuadro de diálogo *Seleccionar color*.

También es posible asignar etiquetas a los colores. La etiqueta aparecerá en la leyenda de la capa ráster.

Al hacer clic con el botón derecho sobre las filas seleccionadas en la tabla de colores, se muestra un menú contextual para:

- *Cambiar Color…* para la selección
- *Cambiar opacidad…* para la selección
- *Cambiar Etiqueta…* para la selección

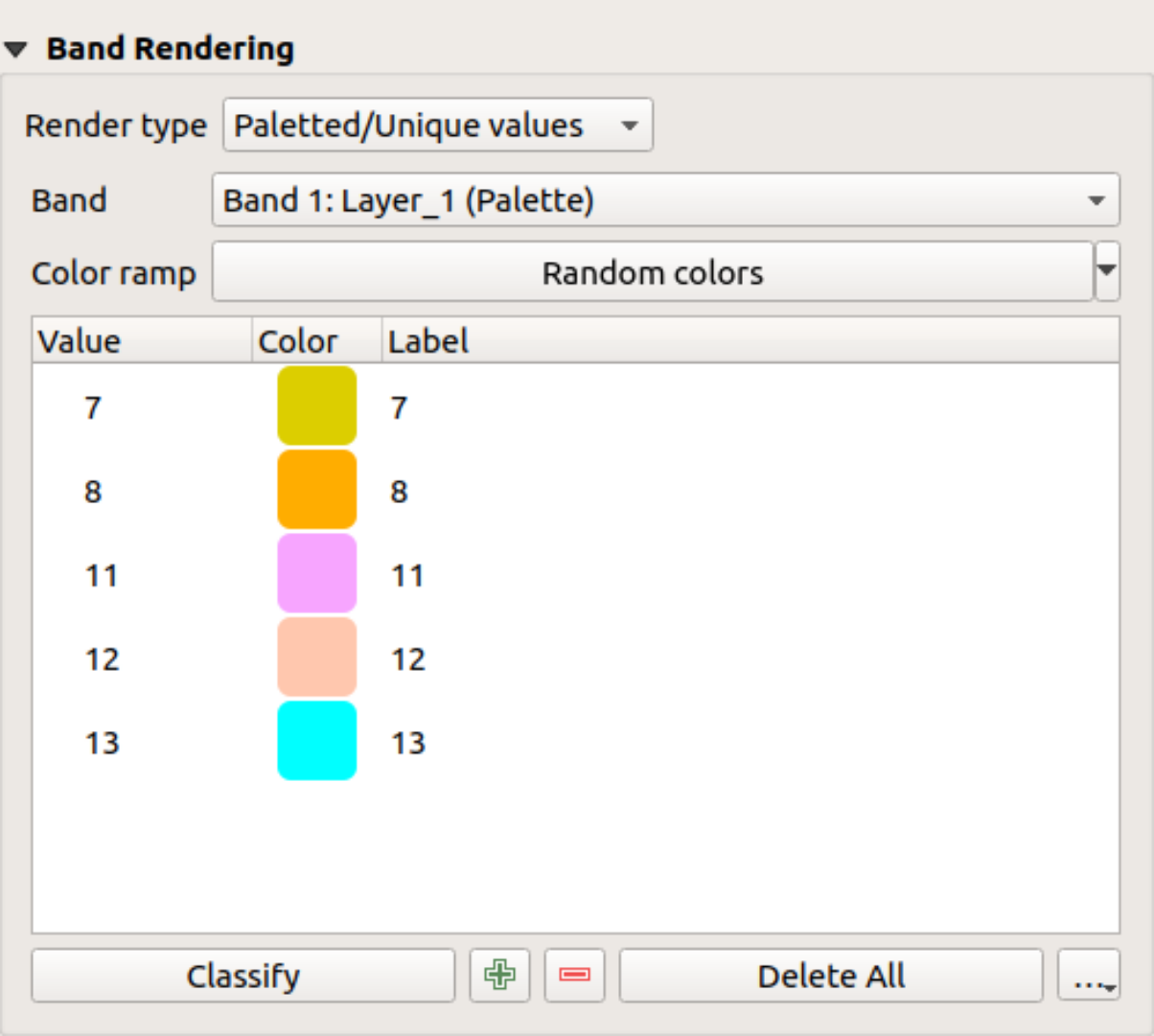


Figura 13.3: Simbología Ráster - Representación paletizada de valor único

El menú desplegable, que se abre al hacer clic en el botón … (Opciones avanzadas) debajo del mapa de color a la derecha, ofrece la carga del mapa de color (*Cargar mapa de color desde archivo. ..*) y exportación (*Exportar mapa de colores a archivo…*), y carga de clases (*Cargar clases desde capa*).

Gris monobanda

Este renderizador te permite renderizar una capa usando solo una banda con un *Degradado de color*: “Negro a blanco” o “Blanco a negro”. Puede cambiar el rango de valores a color (*Min* y *Max*)in the *Min/Max Value Settings*.

Se puede aplicar un método *mejora de contraste* a los valores: “Sin realce”, “Estirar a MinMax”, “Estirar y cortar a MinMax” y “Cortar a min max”.

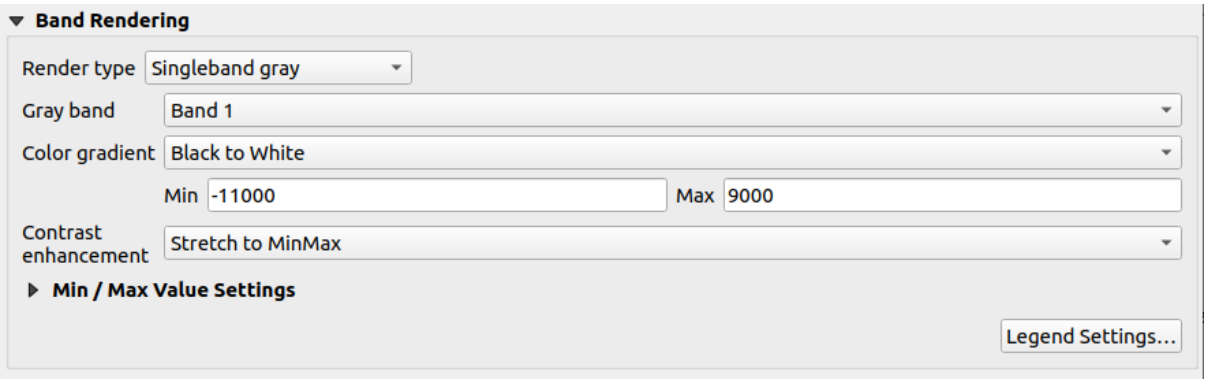


Figura 13.4: Simbología Ráster - representación de gris monobanda

A los píxeles se les asigna un color según el degradado de color seleccionado y la leyenda de la capa (en el panel *Capas* y el diseño t *legend item*) se muestra usando una rampa de color continua. Pulse *Configuración de leyenda…* si desea modificar la configuración. Más detalles en *Personalizar leyenda ráster*.

Pseudocolor Monobanda

Esta es una opción de procesamiento para archivos de una sola banda que incluyen una paleta continua. También puede crear mapas de color para una banda de un ráster multibanda.

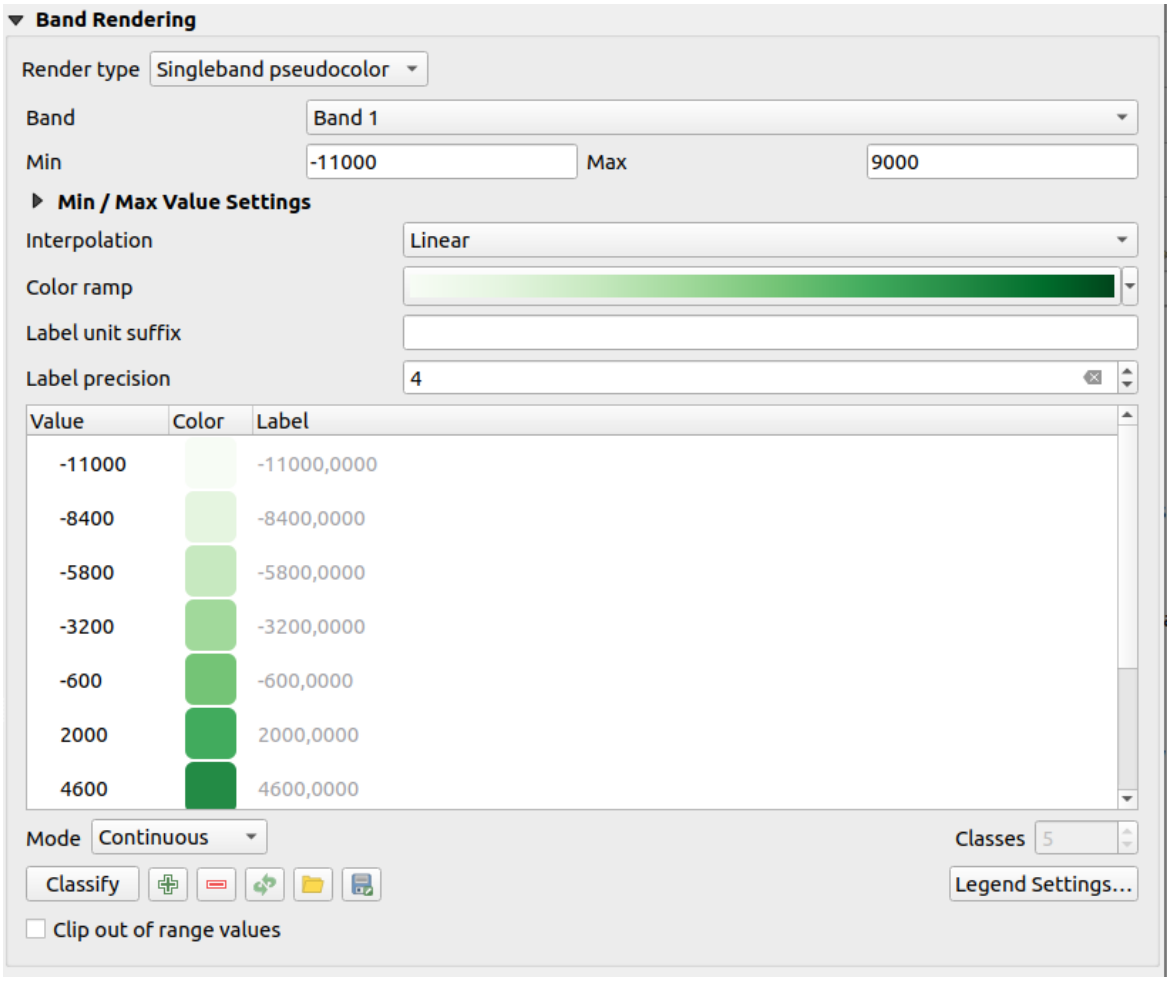


Figura 13.5: Simbología Ráster - representación Pseudocolor Monobanda

Usando una *Bands* de la capa y una *values range*, puede interpolar y asignar colores de representación a los píxeles dentro de las clases. Mas en : ref:*color_ramp_shader*.

A los píxeles se les asigna un color según la rampa de color seleccionada y la leyenda de la capa (en el panel *Capas* y el diseño *legend item*) iSe muestra usando una rampa de color continua. Pulse *Configuración de leyenda…* si desea ajustar la configuración o, en su lugar, utilizar una leyenda con clases separadas (y colores). Más detalles en *Personalizar leyenda ráster*.

Color único

Este renderizador permite representar una capa de trama utilizando *Color único*. Este tipo de renderizador es útil cuando se desea mostrar una capa rasterizada uniformemente, sin ninguna variación de color basada en los valores de los píxeles.

El renderizador de color único puede utilizarse tanto con capas ráster de banda única como multibanda. Cuando se utiliza con rásters multibanda, puede seleccionar a qué banda aplicar el color único, mostrando de forma eficaz esa banda específica de manera uniforme en toda la capa.

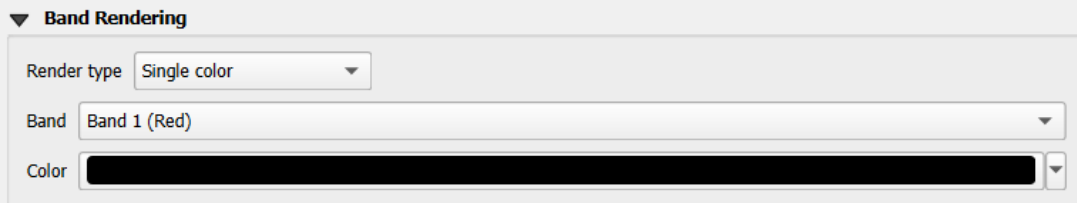


Figura 13.6: Simbología de ráster - Representación monocolor

Sombreado

Representar una banda del ráster usando sombreado.

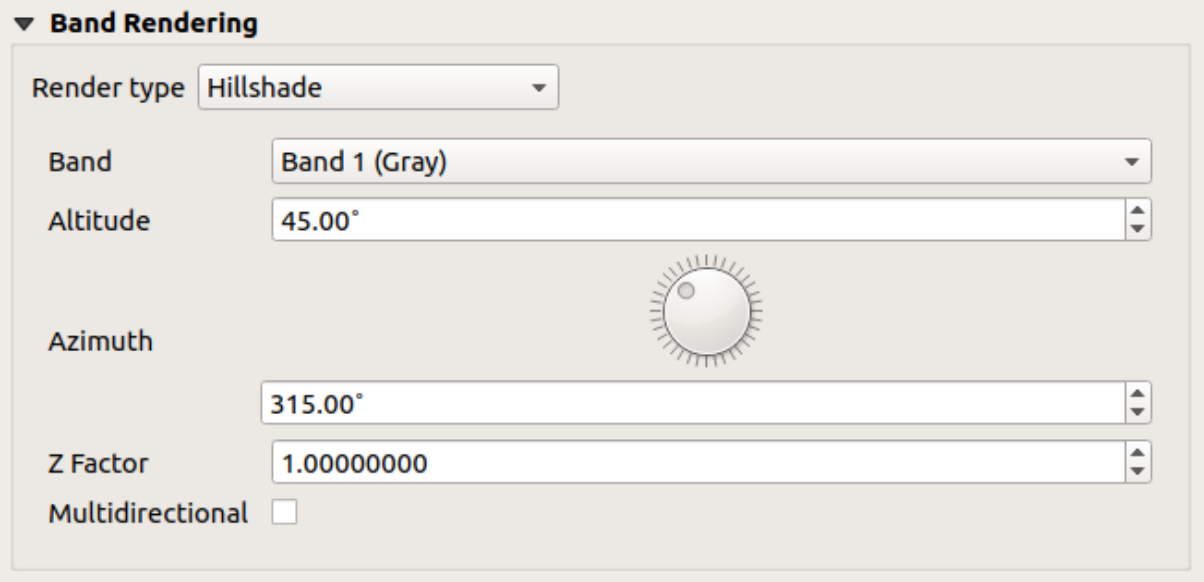


Figura 13.7: Simbología Ráster - representación sombreada

Opciones:

- *Banda*: La banda ráster a usar.

- *Altitud*: El ángulo de elevación de la fuente luminosa (predeterminado es `45°`).
- *Azimuth*: El azimut de la fuente de luz (predeterminado es `315°`).
- *Factor Z*: Factor de Escala para los valores de la banda del ráster (predeterminado es `1`).
- *Multidireccional*: Especifica si se utilizará sombreado multidireccional (el valor predeterminado es `` desactivado “”).

Contornos

Este renderizador traza las líneas de contorno que son calculadas al vuelo desde la banda de fuente ráster.

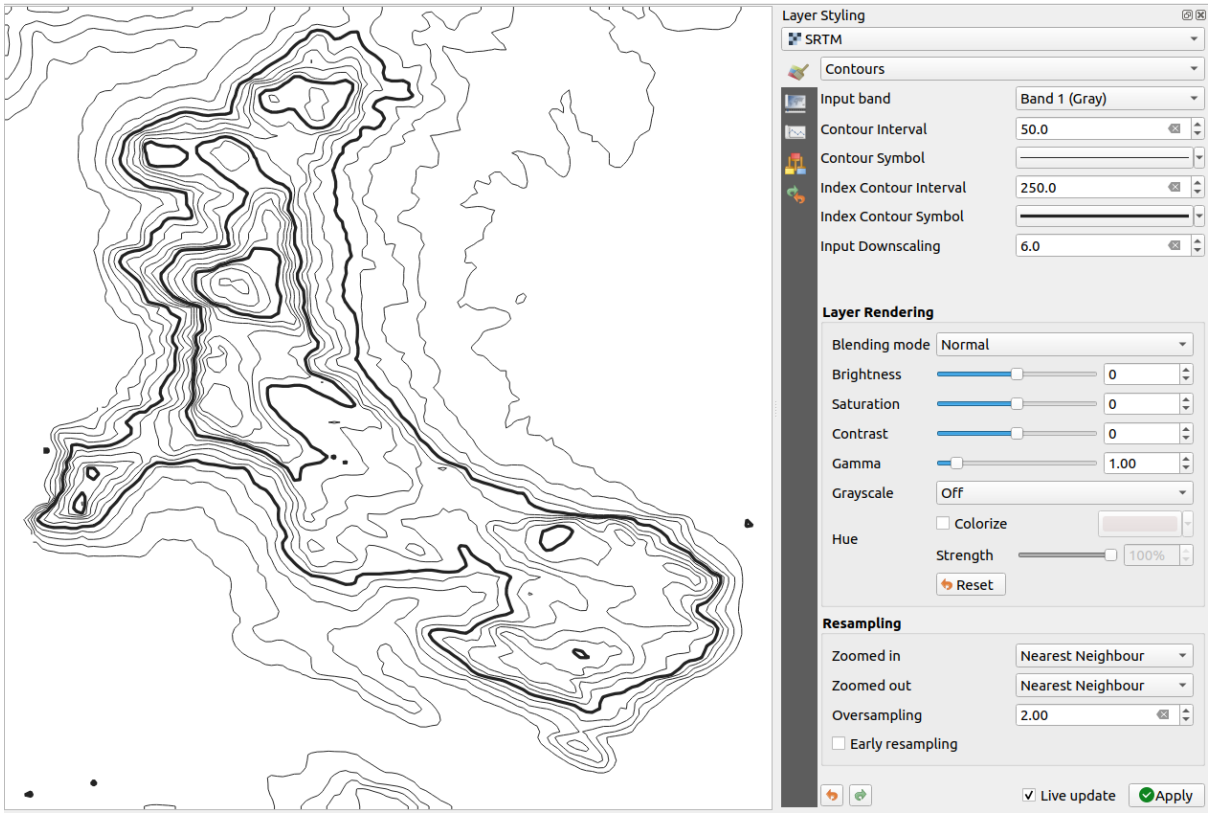


Figura 13.8: Simbología Ráster - renderizado de Contornos

Opciones:

- *banda de Entrada*: la banda ráster a usar.
- *intervalo de Contorno*: la distancia entre dos líneas de contorno consecutivas
- *Símbolo de Contorno*: el *símbolo* a aplicar a las líneas de contorno vulgares.
- *Indice de Intervalo de contorno*: la distancia entre dos **índices de contornos** consecutivos, es decir, las líneas que se muestran de manera distintiva para facilitar la identificación, que se imprimen comúnmente con mayor densidad que otras líneas de contorno y generalmente se etiquetan con un valor a lo largo de su curso.
- *símbolo de índice de contorno*: el símbolo a aplicar al índice de las líneas de contorno
- *Input downscaling*: Indica cuanto reducirá el renderizador la solicitud al proveedor de datos (el valor predeterminado es `4.0`).

  Por ejemplo, si genera líneas de contorno en un bloque ráster de entrada con el mismo tamaño que el bloque ráster de salida, las líneas generadas contendrían demasiados detalles. Este detalle se puede reducir mediante el factor de «reducción de escala», que solicita una resolución más baja del ráster de origen. Para un bloque

ráster 1000x500 con escala descendente 10, el renderizador solicitará ráster 100x50 al proveedor. Una escala más alta hace que las curvas de nivel sean más simplificadas (a costa de perder algunos detalles).

Configurando los valores min y max

De forma predeterminada, QGIS informa los valores *Min* y :guilabel:` Max` de la banda(s) del ráster. Unos pocos valores muy bajos y/o altos pueden tener un impacto negativo en la representación del ráster. El marco *Configuración de valores mín/máx* le ayuda a controlar la representación.

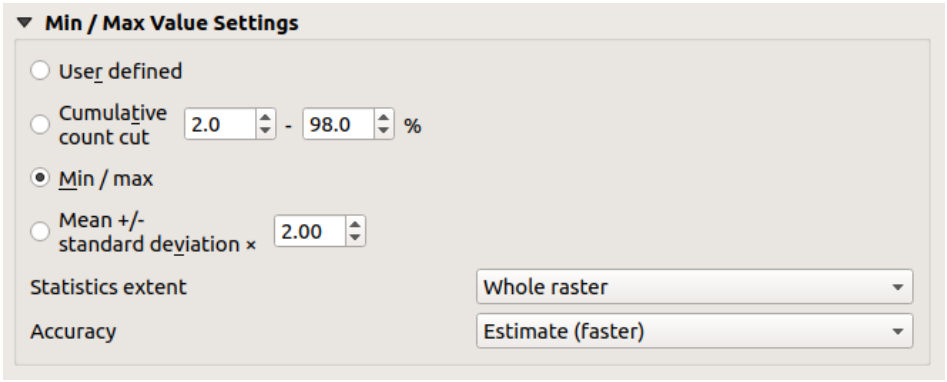


Figura 13.9: Simbología ráster: configuración de valores mínimo y máximo

Las opciones disponibles son:

- *Definido por el usuario*: Los valores predeterminados *Min* y *Max* de la banda(s) pueden ser invalidados
- *Acumulativo corte del conteo*: Elimina valores atípicos. El rango estándar de valores es de `2%` a `98%`, pero se puede adaptar manualmente.
- *Min / max*: Utiliza todo el rango de valores en la banda de la imagen.
- *Media +/- desviación estándar x*: Crea una tabla de colores que solo considera valores dentro de la desviación estándar o dentro de múltiples desviaciones estándar. Esto es útil cuando tiene una o dos celdas con valores anormalmente altos en una capa ráster que impactan negativamente en la representación del ráster.

Los cálculos de los valores mínimo y máximo de las bandas se realizan en base a:

- *Extensión de estadísticas*: puede ser *Ráster completo*, *Lienzo actual* o :guilabel:` Lienzo actualizado`. *Lienzo actualizado* significa que los valores mínimos / máximos utilizados para la representación cambiarán con la extensión del lienzo (estiramiento dinámico).
- *Precisión*, que puede ser *Estimar (rápida)* o *Actual (lenta)*.

Nota

Para algunas configuraciones, es posible que deba presionar el botón *Aplicar* del diálogo de propiedades de la capa para mostrar los valores mínimos y máximos reales en los widgets.

Clasificación de sombreado de rampa de color

Este método se puede utilizar para clasificar y representar conjuntos de datos escalares (ráster o contorno de malla) en función de sus valores. Dado un *color ramp* y una serie de clases, genera entradas intermedias de mapas de colores para los límites de clase. Cada color se mapea con un valor interpolado a partir de un rango de valores y según un modo de clasificación. Luego, a los elementos del conjunto de datos escalares se les asigna su color en función de su clase.

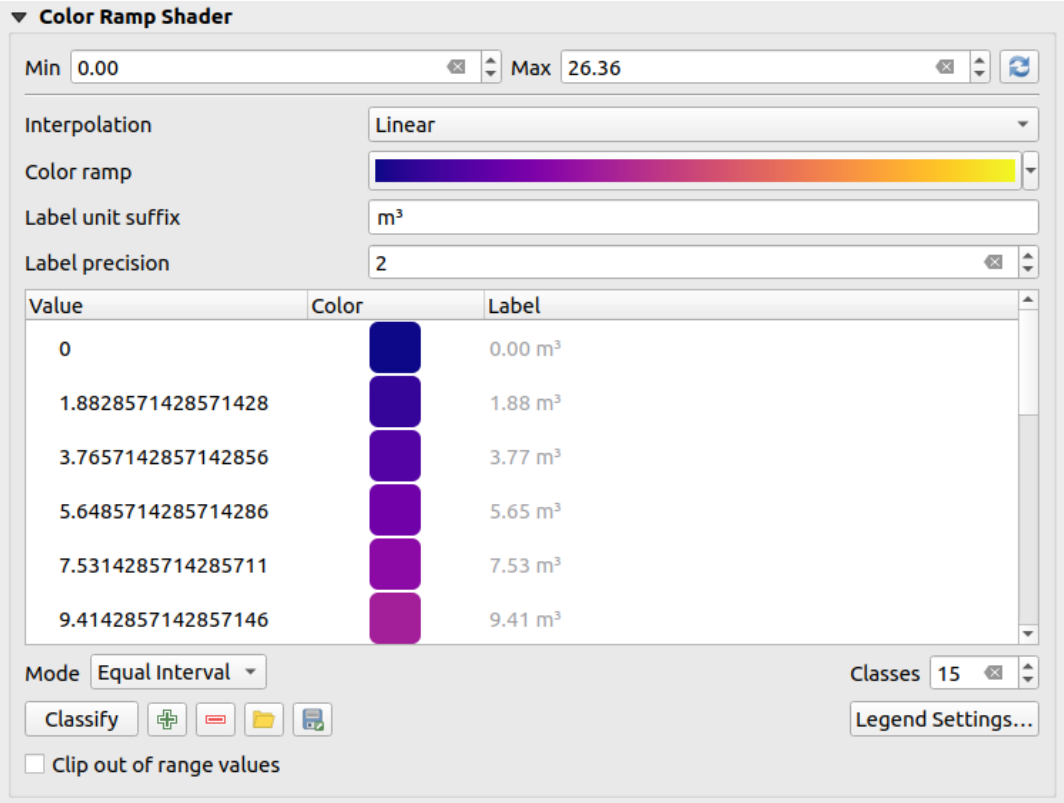


Figura 13.10: Clasificación de un conjunto de datos con un sombreador de rampa de color

1. Se deben definir y utilizar valores de *Min* y *Max* para interpolar los límites de las clases. Por defecto, QGIS los detecta desde el conjunto de datos, pero se pueden modificar.
2. La entrada *Interpolación* define cómo se asigna el color a los elementos escalares:
   - *Discreto* (aparece un símbolo <= en la cabecera de la columna *Valor*): El color se toma de la entrada del mapa de colores más cercana con un valor igual o superior
   - *Linear*: El color se interpola linealmente a partir de las entradas del mapa de colores por encima y por debajo del valor del píxel, lo que significa que a cada valor del conjunto de datos le corresponde un color único
   - *Exacto* (aparece un símbolo = en la cabecera de la columna *Valor*): Sólo a los píxeles con valor igual a una entrada del mapa de color se les aplica un color; los demás no se renderizan.
3. El widget *Rampa de color* te ayuda a seleccionar la rampa de color para asignar al conjunto de datos. Como de costumbre con *this widget*, puede crear uno nuevo y editar o guardar el seleccionado actualmente. El nombre de la rampa de color se guardará en la configuración.
4. *Sufijo unidad etiqueta* añade una etiqueta después del valor en la leyenda, y *Precisión etiqueta* controla el número de decimales a mostrar.
5. La clasificación *Modo* le ayuda a definir cómo se distribuyen los valores entre las clases:
   - *Intervalo igual*: A partir del *Número de clases*, se definen valores límite para que todas las clases tengan la misma magnitud.

- *Continuo*: El número de clase y el color se obtienen de las paradas de la rampa de color; Los valores límites se establecen siguiendo la distribución de paradas en la rampa de color.
- *Quantile*: A partir del *Número de clases*, se definen valores límite para que las clases tengan el mismo número de elementos. No disponible con *mesh layers*.

6. Luego puede *Clasificar* o ajustar las clases:

   - El botón  Add values manually agrega un valor a la tabla.
   - El botón  Remove selected row elimina los valores seleccionados de la tabla.
   - Haciendo doble clic en la columna *Valor* se puede modificar el valor de la clase.
   - Al hacer doble clic en la columna *Color*, se abre el cuadro de diálogo *Cambiar color*, donde puede seleccionar un color para aplicar a ese valor.
   - Haga doble clic en la columna *Etiqueta* para modificar la etiqueta de la clase, pero este valor no se mostrará cuando use la herramienta de identificación de características.
   - Al hacer clic derecho sobre las filas seleccionadas en la tabla de colores, se muestra un menú contextual para *Cambiar color…* y *Cambiar opacidad…* para la selección.

   Puede usar los botones  Cargar mapa de color de archivo o  Exportar mapa de color a archivo para cargar una tabla de colores existente o para guardar la tabla de colores para su uso posterior.

7. Con *Interpolación* lineal, también puedes configurar:

   -  *Recortar valores fuera de rango*: por defecto, el método lineal asigna el color de la primera clase (respectivamente la última clase) a los valores en el conjunto de datos que son más bajos que el conjunto *Min* (respectivamente mayores que el conjunto :guilabel: Valor *máximo*). Marque esta configuración si no desea representar esos valores.
   - *Ajustes de leyenda*, para mostrar en el panel *Capas* y el diseño *legend item*. Más detalles en *Personalizar leyenda ráster*.

Personalizar leyenda ráster

Al aplicar una rampa de color a un ráster o una capa de malla, es posible que desee mostrar una leyenda que muestre la clasificación. De forma predeterminada, QGIS muestra una rampa de color continua con valores mínimos y máximos en el panel *Capas* y el diseño *legend item*. TEsto se puede personalizar usando el botón *Configuración de leyenda* en el widget de clasificación.

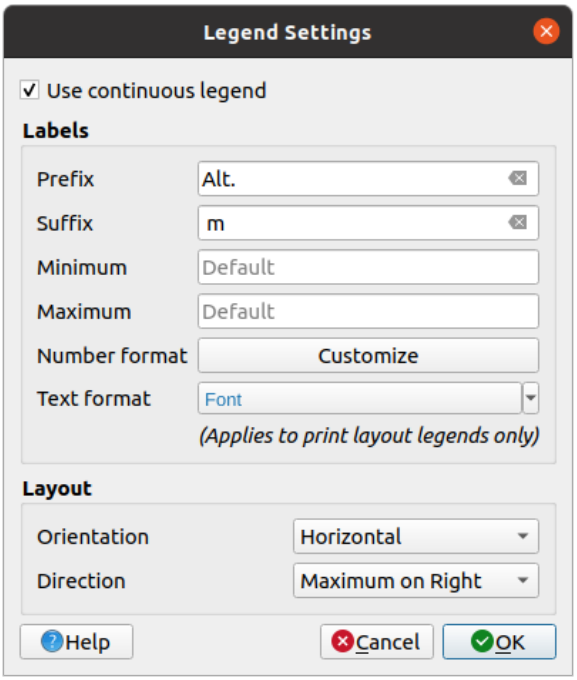


Figura 13.11: Modificar una leyenda de ráster

En este cuadro de diálogo, puede establecer si :guilabel:` Usar leyenda continua`: si no está marcada, la leyenda muestra colores separados correspondientes a las diferentes clases aplicadas. Esta opción no está disponible para la simbología ráster *singleband gray* .

Marcando *Usar leyenda continua* te permite configurar tanto las etiquetas como las propiedades de diseño de la leyenda.

Etiquetas

- Agregue un *Prefijo* y un *Sufijo* a las etiquetas
- Modificar los valores *Mínimo* y *Máximo* para mostrar en la leyenda
- *Customize* el *Formato de número*
- *Customize* El:guilabel:*Formato de texto* para usar en la leyenda del diseño de impresión.

Layout

- Controla la *Orientación* de la rampa de color de la leyenda; puede ser **Vertical** u **Horizontal**
- Controla la *Dirección* de los valores en función de la orientación:
  - Si es vertical, puede mostrar el **Máximo en la parte superior** o el **Mínimo en la parte superior**
  - Si es horizontal, puede mostrar el **Máximo a la derecha** o el **Mínimo a la derecha**

Reproducción de capas

Sobre el tipo de simbología aplicado a la(s) banda(s) de la capa, puede lograr efectos de representación especiales para todo el(los) archivo(s) ráster:

- Usa uno de los modos de fusión (ver *Modos de Mezcla*)
- Establezca *Brillo*, *Saturación*, *Gamma* y *Contraste* personalizados para los colores.
- Con el *Invert colors*, La capa se renderiza con colores opuestos. Útil, por ejemplo, para cambiar los mosaicos de OpenStreetMap listos para usar al modo oscuro.
- Gire la capa a la opción *Escala de grises* ya sea “Por luminosidad”, “Por luminosidad” o “Por promedio”.

- *colorear* y ajusta el *Fuerza* de *Matiz* en la tabla de colores

Presione *Reiniciar* para eliminar cualquier cambio personalizado en la representación de la capa.

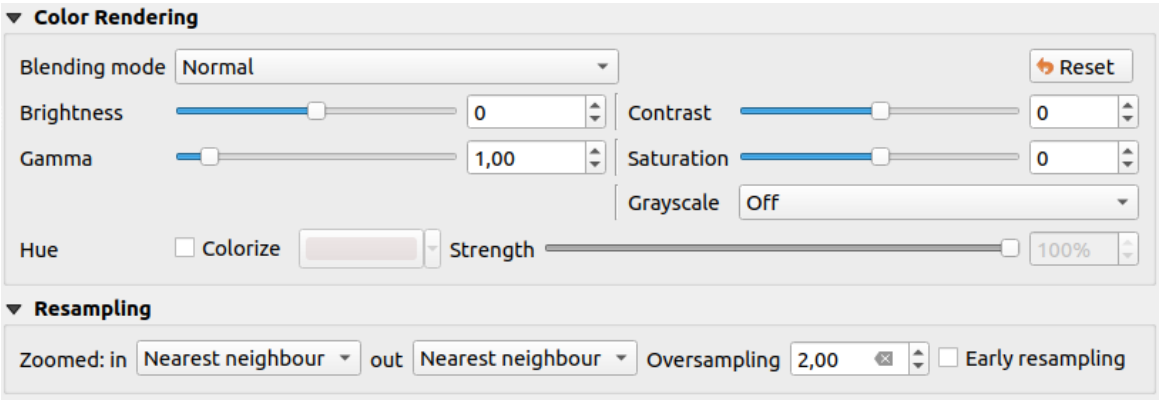


Figura 13.12: Simbología de ráster: configuración de renderizado de capas y remuestreo

Remuestreo

La opción *Remuestreo* tiene efecto al acercar y alejar una imagen. Los modos de remuestreo pueden optimizar la apariencia del mapa. Calculan una nueva matriz de valores grises mediante una transformación geométrica.

Al aplicar el método "Vecino más próximo", el mapa puede adquirir una estructura pixelada al ampliarlo. Este aspecto puede mejorarse utilizando el método «Bilineal (núcleo 2x2)» o «Cúbico (núcleo 4x4)», que difuminan los bordes afilados. El efecto es una imagen más suave. Este método puede aplicarse, por ejemplo, a mapas ráster topográficos digitales.

Remuestreo temprano: permite calcular la resolución ráster a nivel de proveedor donde se conoce la resolución de la fuente y asegura un mejor zoom en el renderizado con el estilo personalizado de QGIS. Realmente conveniente para rásteres de teselas cargados usando un *método9 de interpretación*.

13.1.4 Propiedades de Transparencia

QGIS proporciona capacidades para establecer el nivel de |**transparencia**| *Transparencia* de una capa ráster.

Use el control deslizante *Opacidad global* para establecer hasta qué punto las capas subyacentes (si las hay) deben ser visibles a través de la capa ráster actual. Esto es muy útil si superpone capas ráster (por ejemplo, un mapa de relieve sombreado superpuesto por un mapa ráster clasificado). Esto hará que el aspecto del mapa sea más tridimensional. La opacidad del ráster se puede definir según los datos y variar, p. en función de la visibilidad de otra capa, por variables temporales, en distintas páginas de un atlas,…

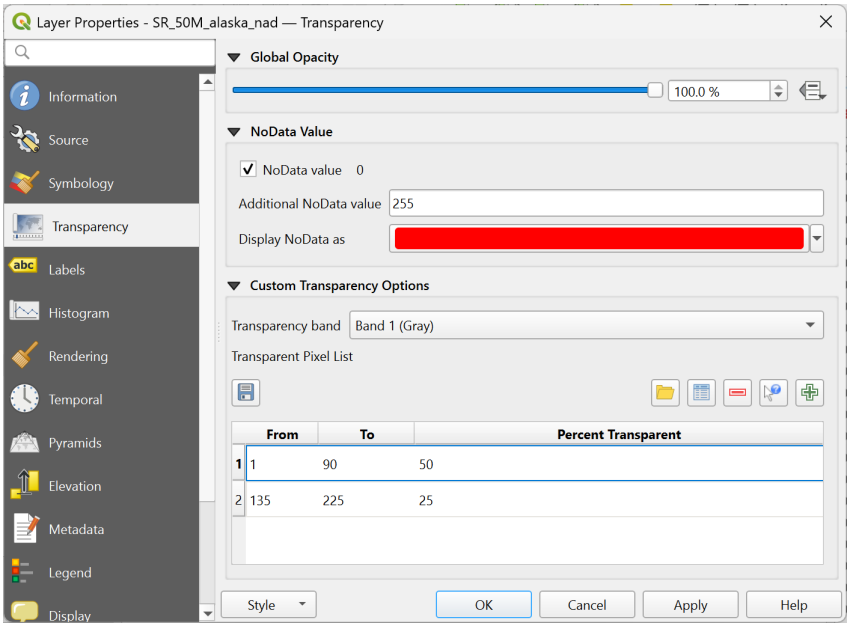


Figura 13.13: Transparencia ráster

Con *No data value* QGIS informa la fuente original sin valor de datos (si está definido) que puede considerar tal como está en la representación. Además, puede ingresar un valor ráster que debe tratarse como un *Valor adicional sin datos*. El selector de color *Mostrar sin datos como* le permite aplicar un color personalizado a los píxeles sin datos, en lugar de la representación transparente predeterminada.

Una forma aún más flexible de personalizar la transparencia está disponible en la sección *Opciones de transparencia personalizadas*:

- Utilice *Banda de transparencia* para aplicar transparencia a toda una banda.
- Proporcione una lista de píxeles para hacer transparentes con los niveles correspondientes de transparencia:

  1. Haga clic en el botón Agregar valores manualmente. Aparecerá una nueva fila en la lista de píxeles.
  2. Para simbología basada en una sola banda (por ejemplo, MDE), introduzca los valores **Desde** y **Hasta** y ajuste el **Porcentaje de transparencia** a aplicar.
  3. Para simbología multibanda (por ejemplo, imágenes RGB) introduzca los valores **Rojo**, **Verde** y **Azul** del píxel y ajuste el **Porcentaje de Transparencia** a aplicar. QGIS soporta **Tolerancia** para los valores de los píxeles, cuando se define la transparencia. Esto significa que los píxeles con colores cercanos a los valores RGB especificados también pueden hacerse transparentes. Tenga en cuenta que esta característica sólo se aplica a rásters multibanda.
  4. Alternativamente, puede obtener los valores de píxeles directamente desde el ráster utilizando el botón Añadir valores de la visualización. Luego ingrese el valor de transparencia.
  5. Repita los pasos para ajustar más valores con transparencia personalizada.
  6. Presiona el botón *Aplicar* y echa un vistazo al mapa.

  Como puede ver, es bastante fácil configurar transparencias personalizadas, pero puede suponer mucho trabajo. Por lo tanto, puede utilizar el botón Exporta a archivo para guardar su lista de transparencias en un archivo. El botón Importar de archivo carga la configuración de transparencia y la aplica a la capa ráster actual.

13.1.5 Labels Properties

The *Labels* properties provides you with all the needed and appropriate capabilities to configure smart labeling on raster layers. This dialog can also be accessed from the *Layer Styling* panel.

At the top of the dialog, you have:

- a combobox for selecting the appropriate labeling method for the active layer
- the Configure project labeling rules button: helps you control interactions between labels and features across the layers in the project. More details at *Configuración de las reglas de etiquetado de proyectos*.
- the Automated placement settings (applies to all layers) button: configure general properties on label placement and conflicts resolution. More details at *Configurar el motor de colocación automatizado*.

The first step is to choose the labeling method from the drop-down list. Available methods are:

- *No labels*: the default value, showing no labels from the layer
- *Label with pixel values*: Show labels on the map using a band.

Using the *Label with pixel values* option, the following dialog opens.

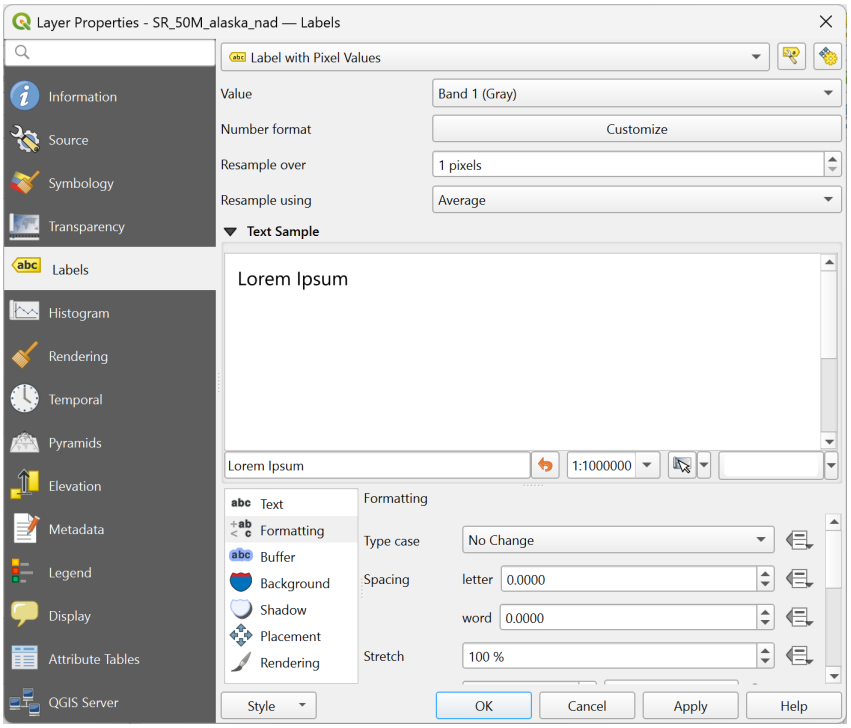


Figura 13.14: Raster layer labeling settings

At the top of the dialog:

- A *Value* drop-down list allows you to select the band to take the values from
- Press *Customize* to configure a *proper number formatting*
- By default, the displayed value represents individual pixel band value. With *Resample over*, you can compute the value from the neighbouring pixels (setting `2` means `2*2=4` pixels) using a statistical method set in the *Resample using* widget.

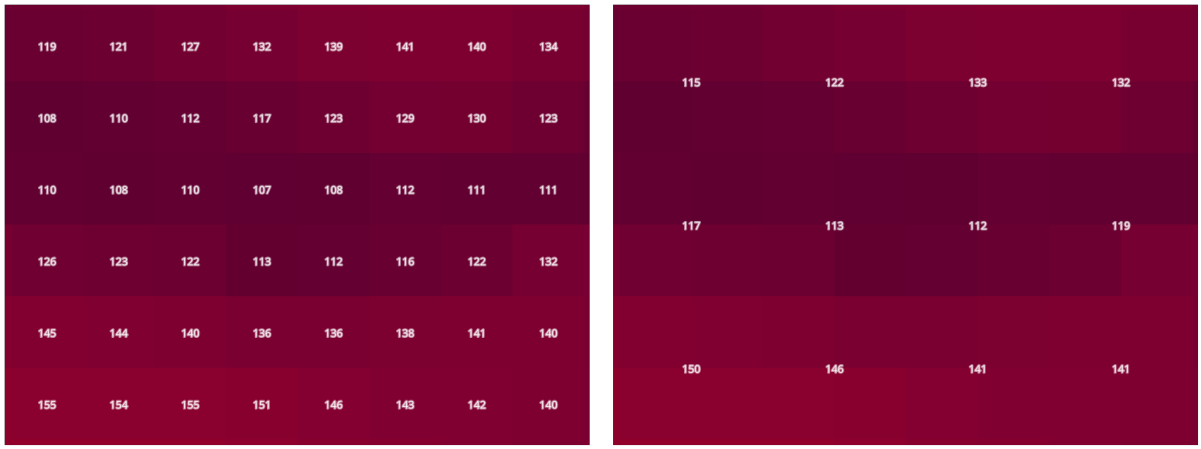


Figura 13.15: Pixels labeled using various resampling options

Below are displayed options to customize the labels, under various tabs:

- *Text*
- *Formatting*
- *Buffer*
- *Background*
- *Shadow*
- *Placement*
- *Rendering*

Description of how to set each property is exposed at *Ajustando una etiqueta*.

13.1.6 Propiedades de Histograma

La pestaña *Histograma* le permite ver la distribución de los valores en su ráster. El histograma se genera cuando presiona el botón *Calcular histograma*. Todas las bandas existentes se mostrarán juntas. Puede guardar el histograma como una imagen con el botón .

En la parte inferior del histograma, puede seleccionar una banda ráster en el menú desplegable y *Establecer estilo mín/máx para*. El menú desplegable *Prefs/Acciones* le ofrece opciones avanzadas para personalizar el histograma:

- Con la opción *Visibilidad*, puede mostrar histogramas para bandas individuales. Deberá seleccionar la opción *Mostrar banda seleccionada*.
- Las *Opciones mín/máx* te permiten “Mostrar siempre los marcadores mín/máx”, “Zoom a mín/máx” Y “Actualizar estilo a mín/máx”.
- La opción *Acciones* le permite “Restablecer” o “Recalcular el histograma” después de haber cambiado los valores mínimo o máximo de la banda(s).

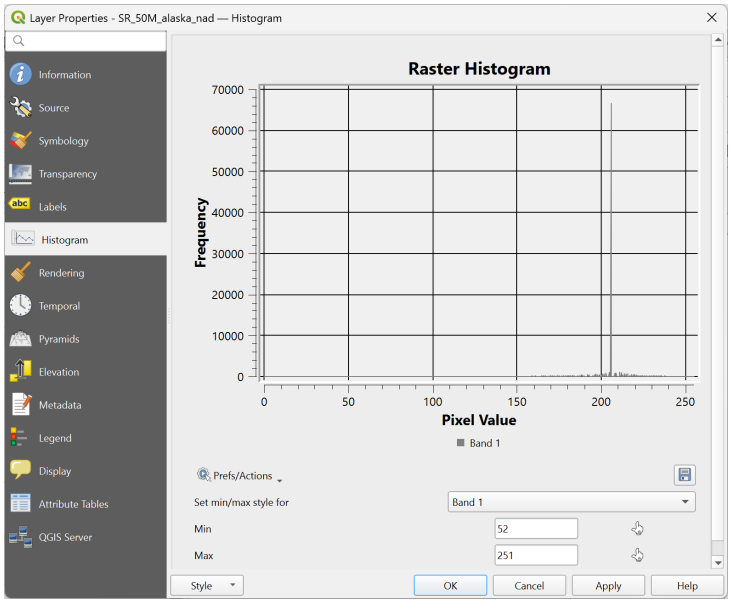


Figura 13.16: Histograma de Ráster

13.1.7 Propiedades de representación

En la pestaña *Rendering*, es posible:

- establece *Visibilidad dependiente de la escala* para la capa: Puede establecer las escalas *Máxima (inclusive)* y *Mínima (exclusive)*, definiendo un rango de escalas en el que la capa será visible. Se ocultará fuera de este rango. El botón Ajustar a la escala actual del lienzo le ayuda a utilizar la escala actual del lienzo del mapa como límite. Véase *Selector de escala de visibilidad* para más información.

  > **Nota**
  >
  > También puede activar la visibilidad dependiente de la escala en una capa desde el panel *Layers*: haga clic con el botón derecho del ratón en la capa y, en el menú contextual, seleccione *Definir la visibilidad de la escala de capas*.

- *Refrescar capa a intervalos*: controla si se puede refrescar una capa y con qué periodicidad. Las opciones *Configuración* disponibles son:

  – *Recargar datos*: la capa se actualizará por completo. Cualquier dato almacenado en caché será descartado y recuperado del proveedor. Este modo puede ralentizar la actualización del mapa.

  – *Redibujar sólo capa*: este modo es útil para animaciones o cuando el estilo de la capa se actualizará a intervalos regulares. Las actualizaciones del lienzo se aplazan para evitar que se actualice varias veces si más de una capa tiene establecido un intervalo de actualización automática.

  También es posible establecer el *Intervalo (segundos)* entre refrescos consecutivos.

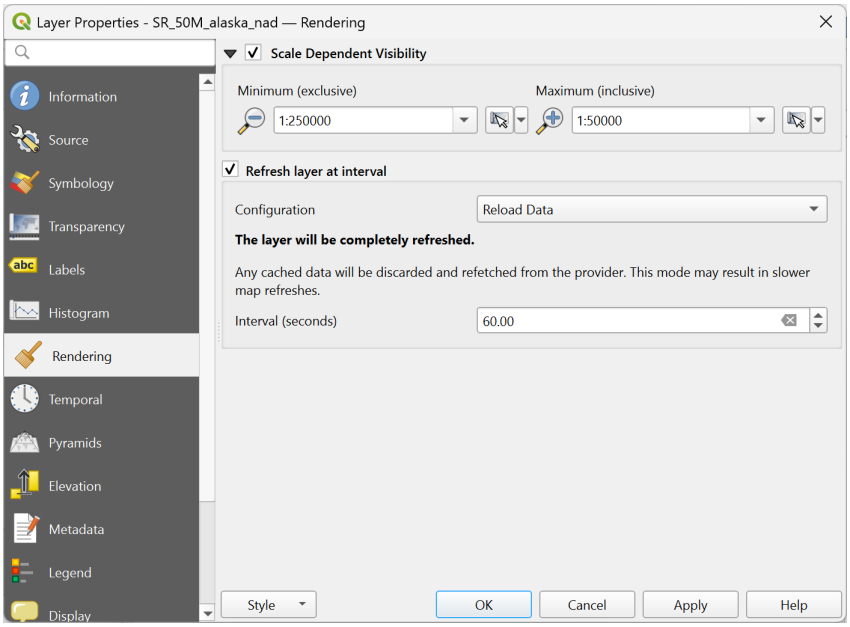


Figura 13.17: Propiedades de Renderizado del Ráster

13.1.8 Propiedades Temporales

El *Temporal* proporciona opciones para controlar el renderizado de la capa a lo largo del tiempo. Esta representación dinámica requiere que la *temporal navigation* esté habilitada en el lienzo del mapa.

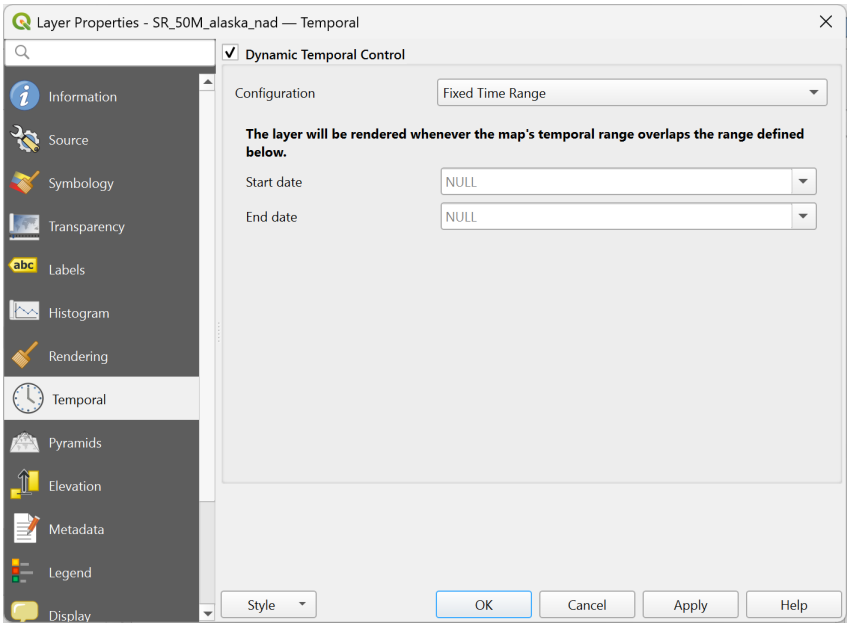


Figura 13.18: Propiedades temporales del ráster

Comprobar la opción *Dynamic Temporal Control* y establezca si el redibujado de la capa debe ser:

- *Automatic*: the rendering is controlled by the underlying data provider if it suppports temporal data handling. E.g. this can be used with WMS-T layers or PostgreSQL rasters.

- *Fixed Date/Time*: only show the raster layer at a single, specific date/time. This avoids having to enter the same value for both the start and end of the temporal range.
- *Fixed time range*: only show the raster layer if the animation time is within a *Start date* and *End date* range.
- *Rango de tiempo fijo por banda*: sólo muestra una banda cuando el tiempo de animación actual se encuentra entre su rango de fechas *Inicio* y *Fin*. Esta opción le permite establecer manualmente estos intervalos de tiempo para cada banda o utilizar el botón  para generar automáticamente valores de fecha-hora, permitiendo un análisis y visualización temporal detallados. Este modo es especialmente útil para trabajar con capas ráster en las que cada banda corresponde a un periodo de tiempo específico, como los archivos NetCDF.
- *Representa Valores Temporales*: interpreta cada píxel de la capa ráster como un valor fecha-hora. Cuando este modo temporal está activo, los píxeles que no caen dentro del rango temporal especificado en el contexto de renderizado se ocultarán, asegurando que sólo se muestren los datos temporalmente relevantes. Este modo es efectivo para:
  - Analizar los cambios en el uso del suelo, como la observación de los patrones de deforestación.
  - Estudiar las inundaciones comparando la cobertura de agua en distintas épocas.
  - Evaluación de los costes de desplazamiento en el análisis del terreno, por ejemplo, utilizando la herramienta r.walk de GRASS GIS para calcular los costes de desplazamiento a través de un paisaje.
- *Redibujar capa solamente*: la capa se redibuja en cada nuevo cuadro de animación. Es útil cuando la capa utiliza valores de expresión basados en el tiempo para la configuración del renderizador (p. ej., la opacidad del renderizador definida por datos, para aparecer o desaparecer gradualmente en una capa ráster).

13.1.9 Propiedades de Pirámides

Las capas ráster de alta resolución pueden ralentizar la navegación en QGIS. Al crear copias de los datos de menor resolución (pirámides), el rendimiento se puede mejorar considerablemente, ya que QGIS selecciona la resolución más adecuada para usar según el nivel de zoom.

Debe tener acceso de escritura en el directorio donde se almacenan los datos originales para construir pirámides.

En la lista *Resoluciones*, seleccione las resoluciones en las que desea crear niveles piramidales haciendo clic en ellos.

Si elige **Interno (Si es posible)** del menú desplegable *Vista general del formato*, QGIS intenta construir pirámides internamente.

> **Nota**
>
> Tenga en cuenta que la construcción de pirámides puede alterar el archivo de datos original y, una vez creadas, no se pueden eliminar. Si desea conservar una versión «no piramidal» de su ráster, haga una copia de seguridad antes de construir la pirámide.

Si elige **Externo** y **Externo (Erdas Imagine) **, las pirámides se crearán en un archivo junto al ráster original con el mismo nombre y una extensión `.ovr`.

Varios *Métodos de remuestreo* se pueden utilizar para el cálculo de la pirámide:

- Vecino más próximo
- Media
- Gauss
- Cúbico
- Spline cúbica
- Laczos
- Modo

- Ninguno

Finalmente, haga clic en *Construir Prirámides* para iniciar el proceso.

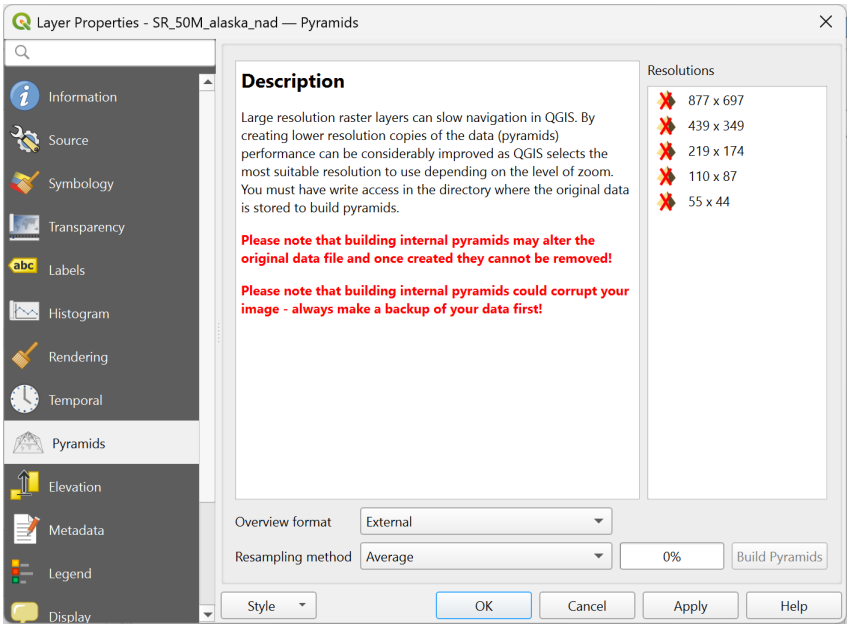


Figura 13.19: Pirámides Ráster

13.1.10 Propiedades de elevación

La pestaña *Elevación* proporciona opciones para controlar las propiedades de elevación de la capa dentro de una *vista de mapa 3D* y su aparición en los *gráficos de la herramienta de perfil*. En concreto, puede elegir *Desactivar* esta configuración si la capa no contiene datos de elevación o puede establecer:

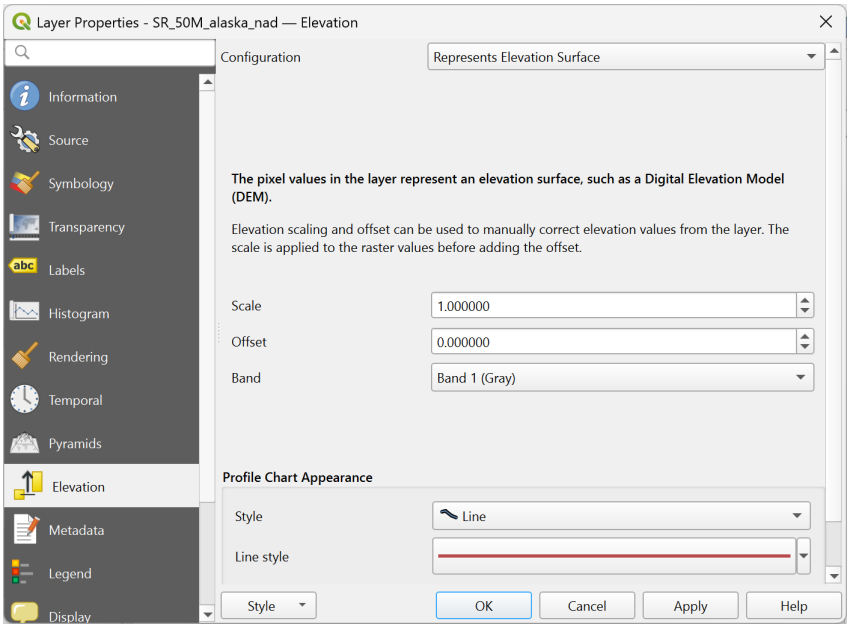


Figura 13.20: Propiedades de elevación raster

- *Representa superficie de elevación*: si la capa ráster representa una superficie de altura (por ejemplo, DEM) y

los valores de píxel deben interpretarse como elevaciones. Elija esta opción si desea mostrar un ráster en una vista *perfil de elevación*. También tendrá que rellenar la *Banda* para elegir los valores y puede aplicar un factor *Escala* y un *Desplazamiento*.

- *Gama de elevación fija*: La capa ráster (o banda ráster seleccionada) está asociada a un rango de elevación fijo. Este modo puede utilizarse cuando una capa tiene una única elevación fija o un rango (rebanada) de valores de elevación. Si se especifica un rango, los píxeles se extruirán sobre este rango. Puede establecer los valores de rango de elevación *Inferior* y *Superior* para la capa, y especificar si los *Límites* inferior o superior son inclusivos o exclusivos.
- *Rango de elevación fijo por banda*: Cada banda del ráster puede tener asociado un rango de elevación fijo. Está diseñado para fuentes de datos que exponen datos relacionados con la elevación en bandas, como los archivos NetCDF. Por ejemplo, un raster con datos de temperatura a diferentes profundidades oceánicas. Al renderizar, se seleccionará la banda superior y se utilizará para los datos de la capa. Esta función se presenta como una tabla editable por el usuario para bandas ráster con valores inferiores y superiores. Los usuarios pueden rellenar los valores inferior y superior manualmente o utilizar una expresión *Expression* para rellenar automáticamente todos los valores de banda basándose en la expresión. El relleno basado en expresiones permite diseñar expresiones que extraigan información útil de los nombres de las bandas. Por ejemplo, extraer el valor de profundidad de un nombre de banda como «Banda 001: profundidad=-5500 (metros)».
- *Dynamic Elevation Range Per Band*: This mode calculates elevation ranges for raster bands dynamically using QGIS expressions. It's ideal for datasets where elevation values follow a consistent pattern across bands (like equally spaced vertical layers), eliminating the need to manually assign fixed elevations. Instead of entering individual values for each band, you define expressions for the *Lower* and *Upper* elevation bounds. These expressions can use variables like `@band`, `@band_name`, or `@band_description` automatically computing the elevation range based on each band's properties.
- *Apariencia del gráfico de perfil*: controla la representación de los datos de elevación ráster en el gráfico de perfil. El *Estilo* del perfil puede establecerse como:
  - una *Línea* con un estilo *Línea*
  - una superficie de elevación representada utilizando un símbolo de relleno por encima (*Rellenar por encima*) o por debajo (*Rellenar por debajo*) de la línea curva de elevación. La simbología de la superficie se representa utilizando:
    - un *Estilo de relleno*
    - y una :guilabel:"Límite": la altitud máxima (o mínima) que determina la altura de la superficie de relleno.

13.1.11 Propiedades de metadatos

El  *Metadata* le brinda opciones para crear y editar un informe de metadatos en su capa. Consulte *Metadatos* para obtener más información.

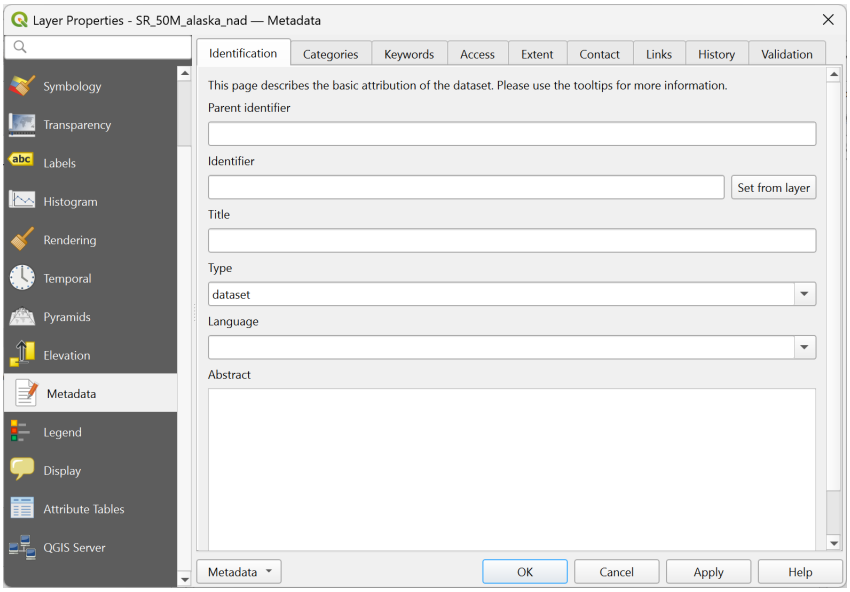


Figura 13.21: Metadatos Ráster

13.1.12 Propiedades de la leyenda

La pestaña de propiedades de *Leyenda* le provee con ajustes avanzados para el *Layers panel* y/o el *print layout legend*. Estas opciones incluyen:

- Dependiendo de la simbología aplicada a la capa, puede terminar con varias entradas en la leyenda, no necesariamente legibles/útiles para mostrar. La *Legend placeholder image* le ayuda a *select an image* para reemplazarla, que se muestra tanto en el panel *Layers* como en la leyenda del diseño de impresión.
- una lista de widgets que puede incrustar dentro del árbol de capas en el panel Capas. La idea es tener una forma de acceder rápidamente a algunas acciones que a menudo se usan con la capa (configuración de transparencia, filtrado, selección, estilo u otras cosas…).

  De forma predeterminada, QGIS proporciona un widget de transparencia, pero esto se puede ampliar mediante complementos que registran sus propios widgets y asignan acciones personalizadas a las capas que administran.

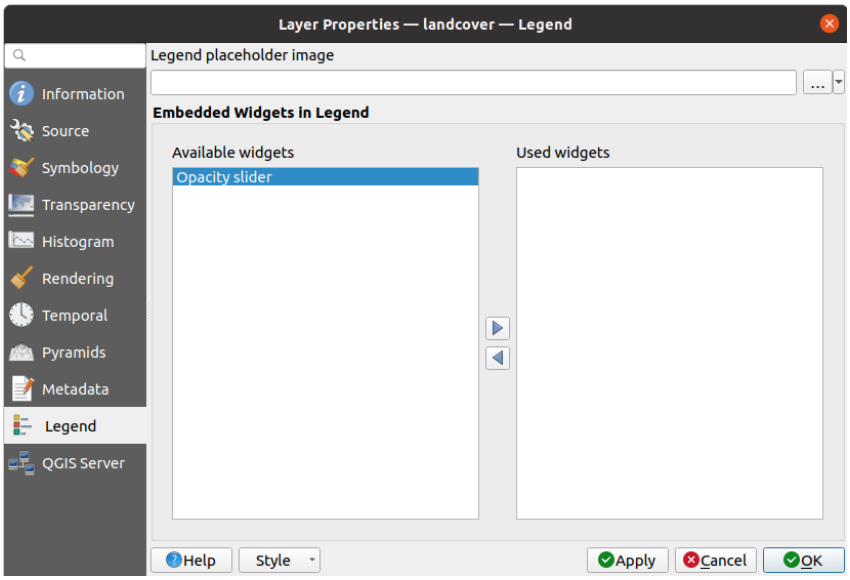


Figura 13.22: Leyenda Ráster

13.1.13 Propiedades a mostrar

La pestaña *Mostrar* le ayuda a configurar los consejos de mapa HTML a utilizar para la identificación de píxeles:

- *Habilitar consejos de mapa* controla si se muestran consejos de mapa para la capa
- El *Consejo de mapa HTML* proporciona un complejo y completo editor de texto HTML para consejos de mapa, mezclando expresiones QGIS y estilos y etiquetas html (multilínea, fuentes, imágenes, hipervínculo, tablas, …). Puede comprobar el resultado de su muestra de código en el marco *Vista previa*. También puede seleccionar y editar expresiones existentes utilizando el botón *Insertar/Editar expresión*.

  Puede buscar expresiones situadas en el grupo *Rasters* o en la variable `@layer_cursor_point` del diálogo *Expresiones*.

> **Nota**
>
> **Entender el comportamiento del motor** *Insertar/Editar Expresión*
>
> Si selecciona algún texto dentro de una expresión (entre «[%» y «%]»), o si no hay ningún texto seleccionado pero el cursor está dentro de una expresión, se seleccionará automáticamente toda la expresión para editarla. Si el cursor o un texto seleccionado está fuera de una expresión, el diálogo se abre con la selección.

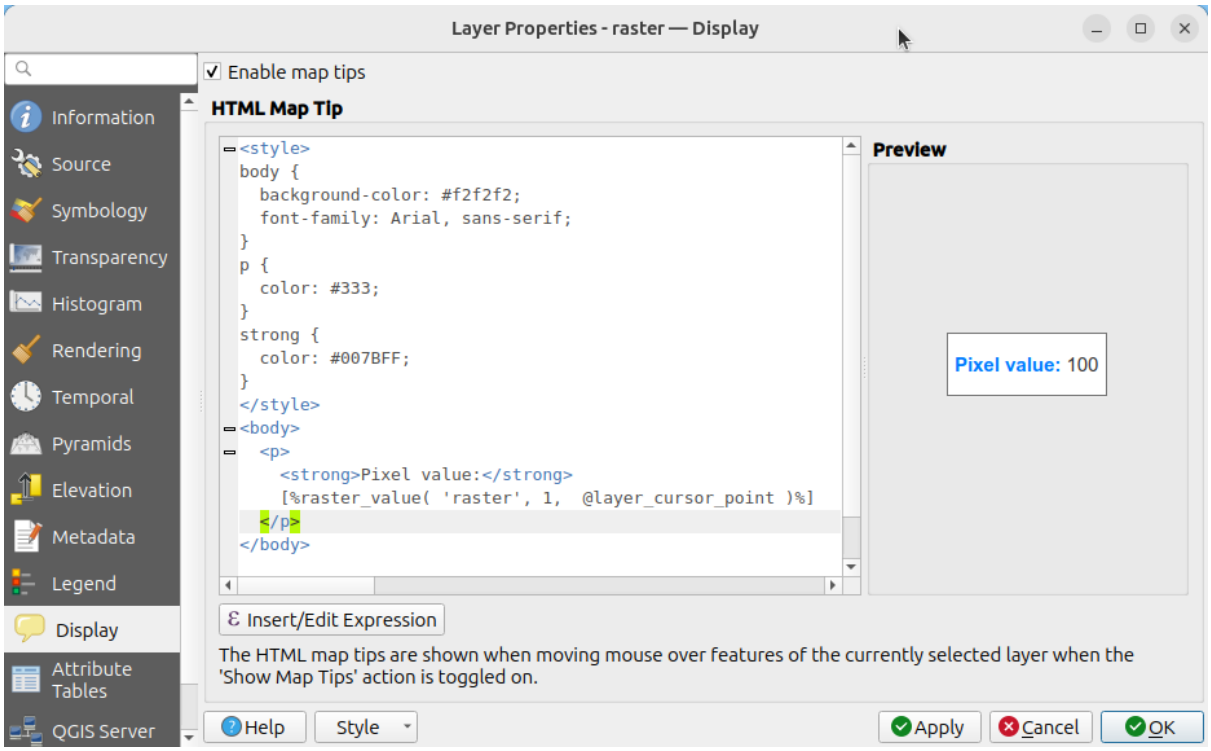


Figura 13.23: Consejos para mapas con capa ráster

Para mostrar consejos sobre el mapa:

1. Seleccione la opción de menú *Ver* ► *Mostrar consejos sobre el mapa* o haga clic en el icono Mostrar consejos sobre el mapa de la barra de herramientas *Atributos*.
2. Asegúrese de que la capa de destino está activa y tiene marcada la propiedad *Habilitar consejos de mapa*.
3. Desplázate sobre un píxel y aparecerá la información correspondiente.

La punta de mapa es un objeto espacial multicapa, lo que significa que, una vez activada, permanece activada y se aplica a cualquier capa habilitada para la punta de mapa en el proyecto hasta que se desactive.

13.1.14 Propiedades de servidor QGIS

La pestaña |cubierta| *Servidor QGIS* le ayuda a configurar los ajustes de los datos cuando son publicados por el Servidor QGIS. La configuración se ocupa de:

- *Description*: provides information to describe the data, such as *Short name*, *Title*, *Abstract*, a *Keyword list*, and a *Data URL* whose *Format* can be in `text/html`, `text/plain` or `application/pdf`.
- *Atribución*: un *Título* y *URL* para identificar quién proporciona los datos.
- *URL de metadatos*: una lista de *URL* para los metadatos que pueden ser de `FGDC` o `TC211` *Tipo*, y en `text/plain` o `text/xml` *Format*
- *URL de la leyenda*: una *URL* para la leyenda, en formato `image/png` o `image/jpeg` *Formato*.

Nota

Cuando la capa ráster que desea publicar ya está proporcionada por un servicio web, están disponibles para ajuste más *propiedades*.

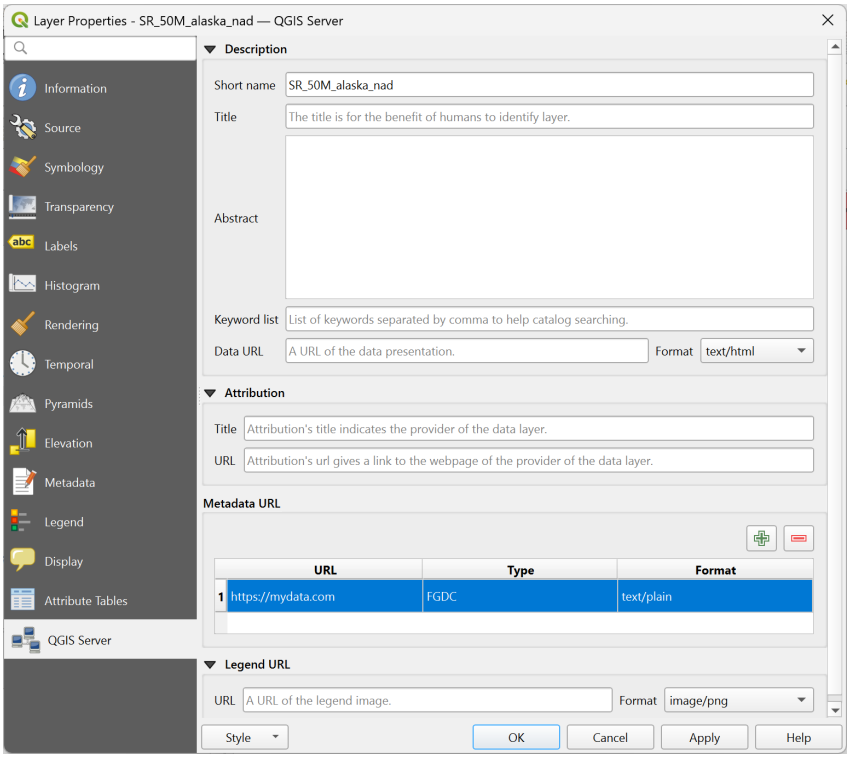


Figura 13.24: Servidor de QGIS en Propiedades de Ráster

13.1.15 Attribute Tables

Similar to vector layers, raster layers in QGIS can have associated Raster Attribute Tables (RATs) that store additional information about raster pixel values.

A Raster Attribute Table links raster values (or value ranges) to descriptive attributes such as class names, pixel counts, and colors. RATs are commonly used for classified rasters, color tables, and histogram information, helping QGIS interpret how raster values should be displayed and classified.

The *Attribute Tables* tab allows you to view and edit raster attribute data. To access the attribute table of a raster layer, right-click the layer in the layers panel, open *Properties…* and select *Attribute Tables*.

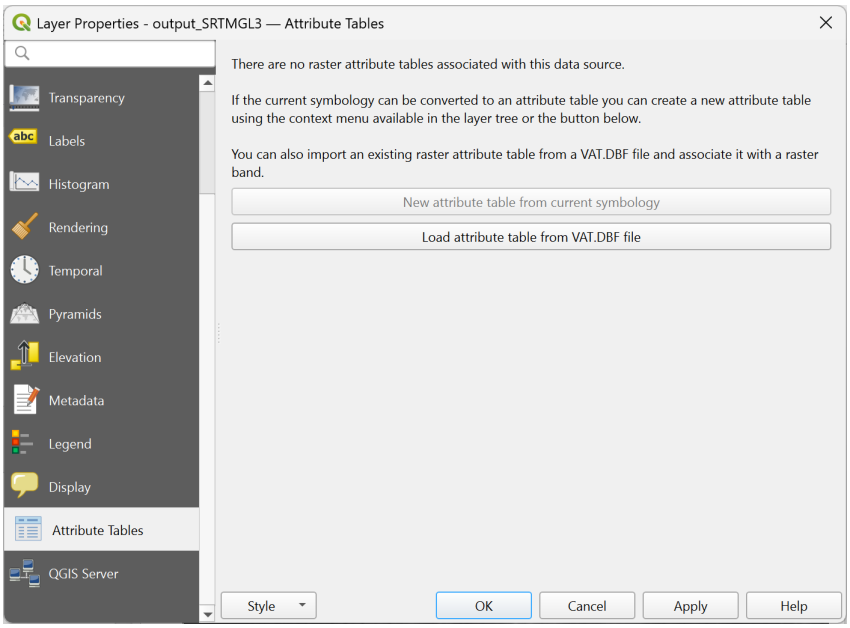


Figura 13.25: Accessing the raster attribute table

When opening the *Attribute Tables* for a raster layer that does not have an associated Raster Attribute Table, QGIS displays a notice informing that no attribute table is currently linked to the data source. In this state, the interface provides two options to define and associate a new attribute table:

- *New attribute table from current symbology*: will appear if your raster layer has a classified symbology applied. Otherwise, this option is disabled. Applying classified symbology also triggers an option in the layer context menu to *Create Raster Attribute Table*. Click this button and set the following options in the *New Raster Attribute Table* dialog.
  - *Managed by the data provider*: the attribute table will be saved and managed by the data provider (if supported), overwriting any existing table for the raster band used by the current style. Depending on the data provider, the attribute table will be embedded in the main raster file or saved into a sidecar file managed by the data provider.
  - *Sidecar VAT.DBF file*: saves the attribute table into a sidecar VAT.DBF file. The resulting file will not be associated with any particular band.

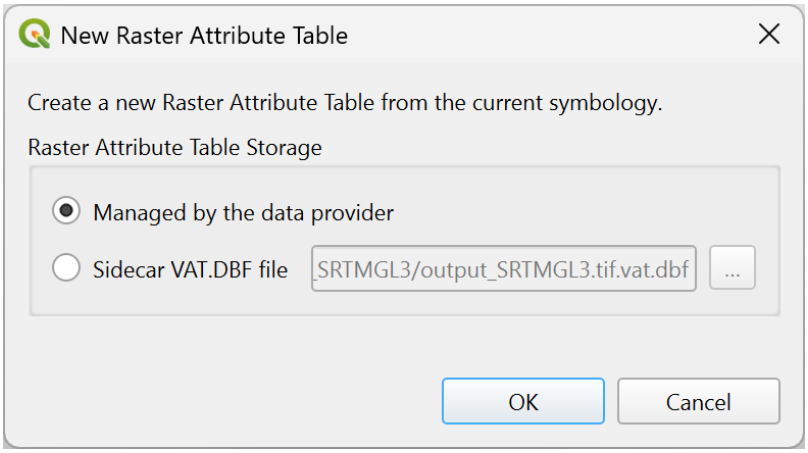


Figura 13.26: Create new attribute table from current symbology

- *Load attribute table from VAT.DBF file*: retrieves the attribute table from an external DBF file. *Load Raster Attribute Table* dialog will appear to let you browse and select the DBF file and associate it with the raster band. The action also can be done by right-clicking the raster layer in the layers panel and selecting *Load Raster Attribute Table from VAT.DBF*. *Open the newly loaded attribute table* allows you to directly open the attribute table editor after loading the table.

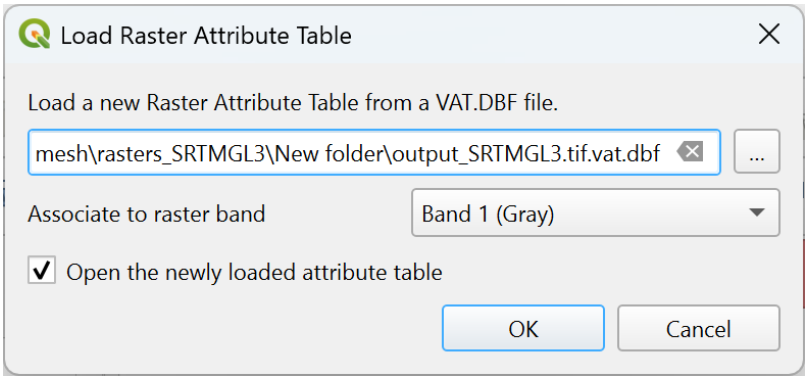


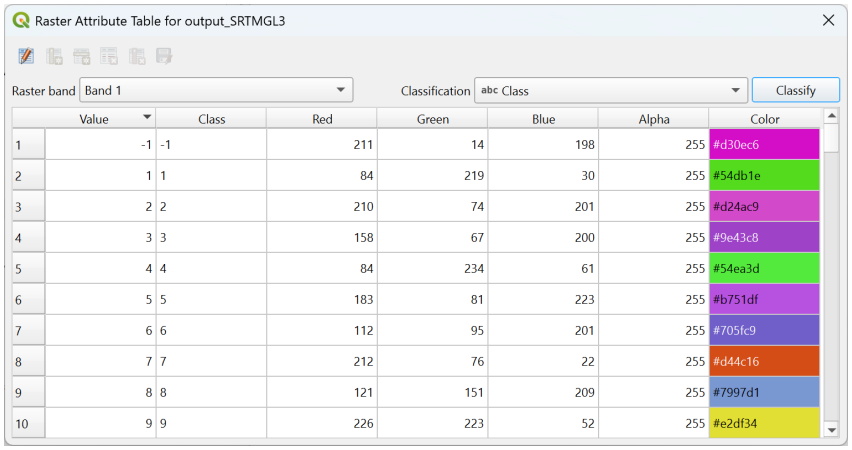
Figura 13.27: Load attribute table from VAT.DBF file

Editing raster attribute tables

When set, a raster attribute table can be accessed from the *Attribute tables* tab of the layer properties dialog or by selecting *Open raster attribute table* from the layer contextual menu in *Layers* panel. This dialog shows a table whose rows correspond to the different pixel values or value ranges classified in a given band of the raster layer. Depending on the type of classification, a row is described by:

- for an exact values classification (e.g., paletted/unique values): the band name and the applied *Color*. Additional fields of the *Red*, *Green*, *Blue* and *Opacity* of the color are displayed.
- for range values classification (e.g., singleband pseudocolor): the range of values defined by their *Min*, *Max*, the associated label (*Class*) and color ramp interpolated over the values (*Color*). Additional fields of the minimum and maximum of *Red*, *Green*, *Blue* and *Opacity* of the color ramp are displayed.

You can also customize the raster attribute table by adding new fields for classification and group different value ranges accordingly, modifying value ranges and their associated colors or labels, adding new rows, … The new attribute table can be used to update the current layer symbology.



	Value	Class	Red	Green	Blue	Alpha	Color
1	-1	-1	211	14	198	255	#d30ec6
2	1	1	84	219	30	255	#54db1e
3	2	2	210	74	201	255	#d24ac9
4	3	3	158	67	200	255	#9e43c8
5	4	4	84	234	61	255	#54ea3d
6	5	5	183	81	223	255	#b751df
7	6	6	112	95	201	255	#705fc9
8	7	7	212	76	22	255	#d44c16
9	8	8	121	151	209	255	#7997d1
10	9	9	226	223	52	255	#e2df34

Figura 13.28: Raster Attribute Table

At the top of the dialog, a set of tools allows to edit and save changes to the attribute table:

- Edit Attribute Table turns the table into edit mode. If no specific row or field is selected, only the Add Column… option is available.
- Add Column… opens the *Add Column* dialog to add a new column to the raster attribute table. The dialog lets you define:
  - *Column Type*, which can be:
    - *Standard column*
    - *Color RGBA*
    - *Color ramp (RGBA minimum and RGBA maximum)*
  - *Column Definition*, including:
    - *Name* of the column (mandatory)
    - *Usage*, which can be:
      - *Pixel Count*: stores the number of pixels belonging to a given value or value range.
      - *Generic*: general-purpose column with no predefined meaning.
      - *Name*: stores the class name or label associated with a raster value or range.
    - *Data type*, which can be:
      - *String*
      - *Integer*
      - *Long Integer*
      - *Double*
  - *Insertion point*, defining whether the new column is inserted *Before* or *After* a selected existing column.

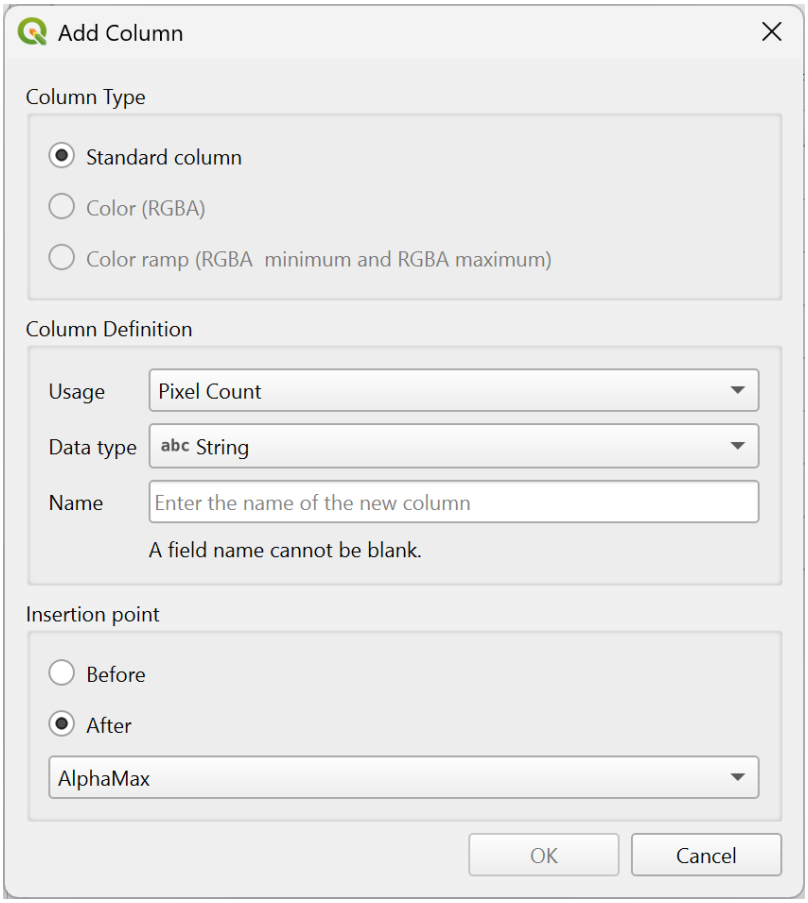


Figura 13.29: Add Column dialog

- *Add Row…* opens the *Add Row* dialog to add a new row to the raster attribute table. Choose the *Insertion point* as *Before current row* or *After current row*.
- Remove Row removes the selected row from the raster attribute table.
- Remove Column removes the selected column from the raster attribute table.
- Save Changes saves any modifications made to the raster attribute table.

The next row provides options for controlling the displayed information in the table:

- *Raster band*: allows selecting which raster band the attribute table applies to. Raster layers may contain multiple bands (e.g., multispectral data). Each band can have its own Raster Attribute Table.
- *Classification*: allows selecting the fields used for classifying the raster layer. Keep in mind that the existing symbology for the raster will be replaced by a new symbology from the attribute table and any unsaved changes to the current symbology will be lost. Make sure to save your symbology style if needed before applying a classification.

13.1.16 Identificar celdas ráster

La herramienta  *identificar objetos espaciales* permite obtener información sobre puntos concretos de una capa ráster.

Para utilizar la herramienta  *Identificar objetos espaciales*:

1. Seleccione la capa ráster en el panel Capas.
2. Haga clic en la herramienta *Identificar objetos espaciales* de la barra de herramientas o pulse `Ctrl+Mayús+I`.
3. Click on the point in the raster layer that you want to identify. The pixel will get highlighted.

El panel Identificar resultados se abrirá en su vista predeterminada `Árbol` y mostrará información sobre el punto seleccionado. El formato de los resultados varía en función del proveedor de la capa. Por ejemplo:

- Para una capa raster local: debajo del nombre de la capa, tiene a la izquierda la(s) banda(s) del píxel pulsado, y a la derecha su valor respectivo.
- Para una capa remota como WMS, un menú *Formato* permite seleccionar si la información debe mostrarse como *HTML*, *Objeto espacial* o *Texto*.

Estos valores también se pueden representar (desde el menú *Vista* situado en la parte inferior del panel) en:

- una vista de `Tabla`: organiza la información sobre los objetos espaciales identificados y sus valores en una tabla.
- una vista «Gráfico»: organiza la información sobre los objetos espaciales identificados y sus valores en un gráfico.

Debajo de los atributos de píxel, encontrará la información *Derivado*, como:

- Valores de las coordenadas `X` e `Y` del punto clickado
- Columna y fila del punto pulsado (píxel) cuando es compatible

13.2 Análisis raster

13.2.1 Calculadora Ráster

La *Calculadora ráster* en *Ráster* le permite realizar cálculos sobre la base de los valores de píxeles ráster existentes (ver Figura 13.30). Los resultados se escriben en una nueva capa ráster en un formato compatible con GDAL.

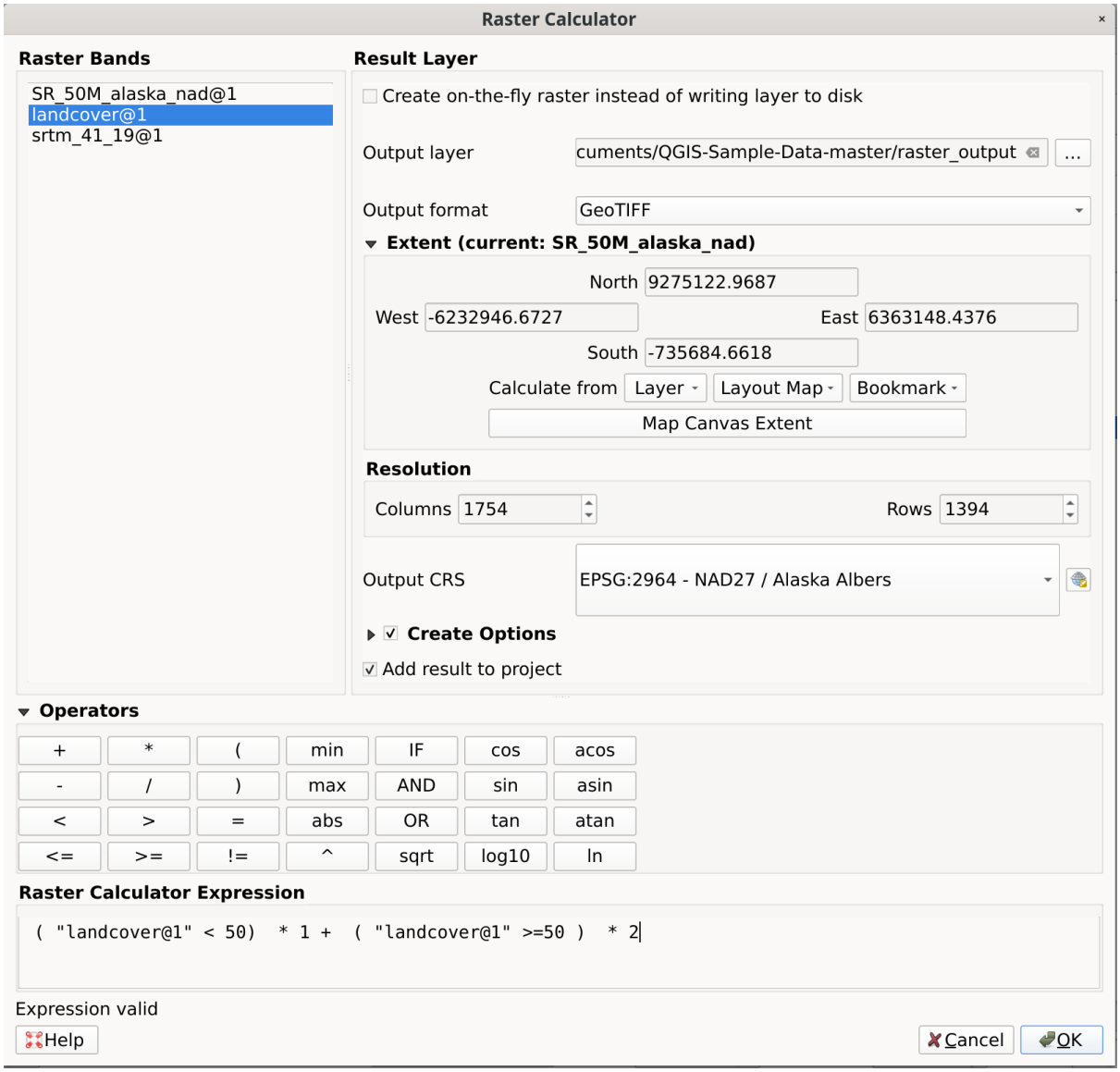


Figura 13.30: Calculadora Ráster

La lista de *Bandas de ráster* contiene todas las capas de ráster cargadas que se pueden usar. Para agregar un ráster al campo de expresión de la calculadora ráster, haga doble clic en su nombre en la lista Campos. Luego puede usar los operadores para construir expresiones de cálculo, o simplemente puede escribirlas en el cuadro.

En la sección *Capa de resultado*, deberás definir una capa de salida. Puede:

- *Crea un ráster sobre la marcha en lugar de escribir una capa en el disco*:
  - Si no está marcada, la salida se almacena en el disco como un nuevo archivo sin formato. Se requiere una ruta *Capa de salida* y un *Formato de salida*.
  - Si está marcada, se crea una capa ráster virtual, es decir, una capa ráster definida por su URI y cuyos píxeles se calculan sobre la marcha. No es un archivo nuevo en el disco; la capa virtual todavía está conectada a los rásteres utilizados en el cálculo, lo que significa que eliminar o mover estos rásteres la rompería. Se puede proporcionar un *Nombre de la capa*, en caso contrario se utiliza la expresión de cálculo como tal. Al eliminar la capa virtual del proyecto, se elimina y se puede hacer persistente en el archivo mediante el menú contextual de la capa *Exportar* ► *Guardar como…*.
- Define the *Spatial extent* of the calculation using the *extent selector* widget.
- Establece la *Resolución* de la capa usando el número de columnas y filas. Si la capa de entrada tiene una

resolución diferente, los valores se volverán a muestrear con el algoritmo vecino más cercano.

- Enable the *Create Options* to define additional raster creation options. These options let you control how the output file is structured and compressed, including format, compression method, and other driver-specific settings. See more at *Raster driver options*.
- Con la **|casilla|** :guilabel:casilla de verificación *Agregar resultado al proyecto*, la capa de resultado se agregará automáticamente al área de la leyenda y se podrá visualizar. Marcado de forma predeterminada para rásteres virtuales.

La sección *Operadores* contiene todos los operadores disponibles. Para agregar un operador al cuadro de expresión de la calculadora ráster, haga clic en el botón correspondiente. Cálculos matemáticos (+, −, *, …) y funciones trigonométricas (`sin`, `cos`, `tan`, …) están disponibles. Las expresiones condicionales (=, !=, <, >=, …) devuelven 0 para falso o 1 para verdadero, y por lo tanto se pueden usar con otras Operadores y funciones.

Ver también

algoritmos *Calculadora ráster* y :ref:`qgisvirtualrastercalc

Expresión de la calculadora de ráster

El diálogo

El cuadro de diálogo *Expresión de calculadora ráster* proporciona medios para escribir expresiones para cálculos de píxeles entre un conjunto de capas ráster.

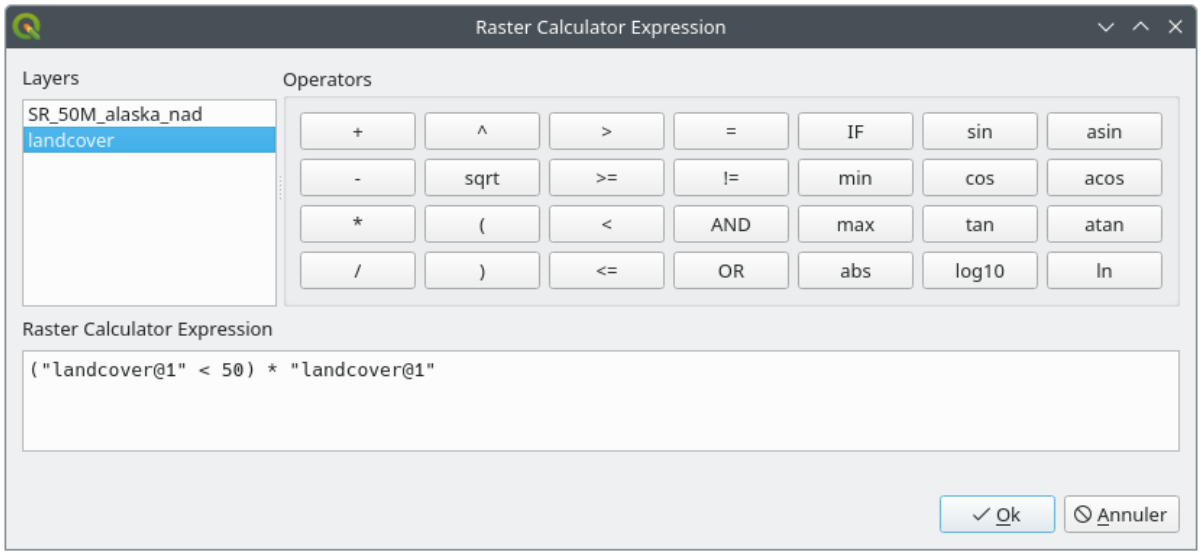


Figura 13.31: Calculadora de expresiones ráster

- *Capas*: Muestra la lista de todas las capas ráster cargadas en la leyenda. Pueden utilizarse para rellenar el cuadro de expresión (doble clic para añadir). Las capas ráster se referencian por su nombre y el número de la banda: `nombre_capa@número_banda`. Por ejemplo, la primera banda de una capa llamada `DEM` se denominará `DEM@1`.
- *Operadores*: contiene una serie de operadores de cálculo para la manipulación de píxeles:
  - Aritméticas: +, −, *, `sqrt`, `abs`, `ln`, …
  - Trigonométricas: `sin`, `cos`, `tan`, …
  - Comparación: =, !=, <, >=, …

  - Lógicas: `IF`, `AND`, `OR`, `(`, `)`
  - Estadística: `min`, `max`

  Para añadir un operador al cuadro de expresión de la calculadora raster, haga clic en el botón correspondiente.

- *Expresión de la calculadora ráster* es el área en la que se compone la expresión

Ejemplos

Convierta los valores de elevación de metros a pies

Al crear un ráster de elevación en pies a partir de un ráster en metros, debe usar el factor de conversión de metros a pies: 3.28. La expresión es:

```
"elevation@1" * 3.28
```

Usando una máscara

Si desea enmascarar partes de un ráster, digamos, por ejemplo, porque solo está interesado en elevaciones superiores a 0 metros, puede utilizar la siguiente expresión para crear una máscara y aplicar el resultado a un ráster en un solo paso.

```
("elevation@1" >= 0) * "elevation@1"
```

En otras palabras, para cada celda mayor o igual a 0, la expresión condicional se evalúa en 1, lo que mantiene el valor original multiplicándolo por 1. De lo contrario, la expresión condicional se evalúa a 0, lo que establece el valor ráster a 0. Esto crea el máscara sobre la marcha.

Clasificar un ráster

Si desea clasificar un ráster –digamos, por ejemplo en dos clases de elevación, puede utilizar la siguiente expresión para crear un ráster con dos valores 1 y 2 en un solo paso.

```
("elevation@1" < 50) * 1 + ("elevation@1" >= 50) * 2
```

En otras palabras, para cada celda menor que 50, establezca su valor en 1. Para cada celda mayor o igual que 50, establezca su valor en 2.

O puede usar el operador `` SI “”.

```
if ( elevation@1 < 50 , 1 , 2 )
```

CAPÍTULO 14

Trabajando con Malla de Datos

14.1 Qué es una malla?

Una malla es una cuadrícula no estructurada que suele tener componentes temporales y de otro tipo. El componente espacial contiene una colección de vértices, aristas y/o caras, en un espacio 2D o 3D:

- **vértices** - puntos XY(Z) (en el sistema de referencia de coordenadas de la capa)
- **bordes** - conecta pares de vértices
- **faceta** - una faceta es un conjunto de segmentos que producen una forma cerrada - tipicamente un triángulo o un cuadrilátero (cuadros), raramente polígonos con mas vértices

Basándose en lo anterior, las capas de malla pueden tener diferentes tipos de estructura:

- Mallas 1D: constan de vértices y aristas. Un borde conecta dos vértices y puede tener datos asignados (escalares o vectores). La red de malla 1D se puede utilizar, por ejemplo, para modelar un sistema de drenaje urbano.
- Mallas 2D: consisten en caras con triángulos, cuadriláteros regulares o no estructurados.
- Mallas en capas 3D: consisten en múltiples mallas 2D no estructuradas apiladas, cada una extruida en la dirección vertical (niveles) por medio de una coordenada vertical. Los vértices y las caras tienen la misma topología en cada nivel vertical. La definición de malla (extrusión de nivel vertical) podría, en general, cambiar con el tiempo. Los datos suelen estar definidos en centros de volumen o por alguna función paramétrica.

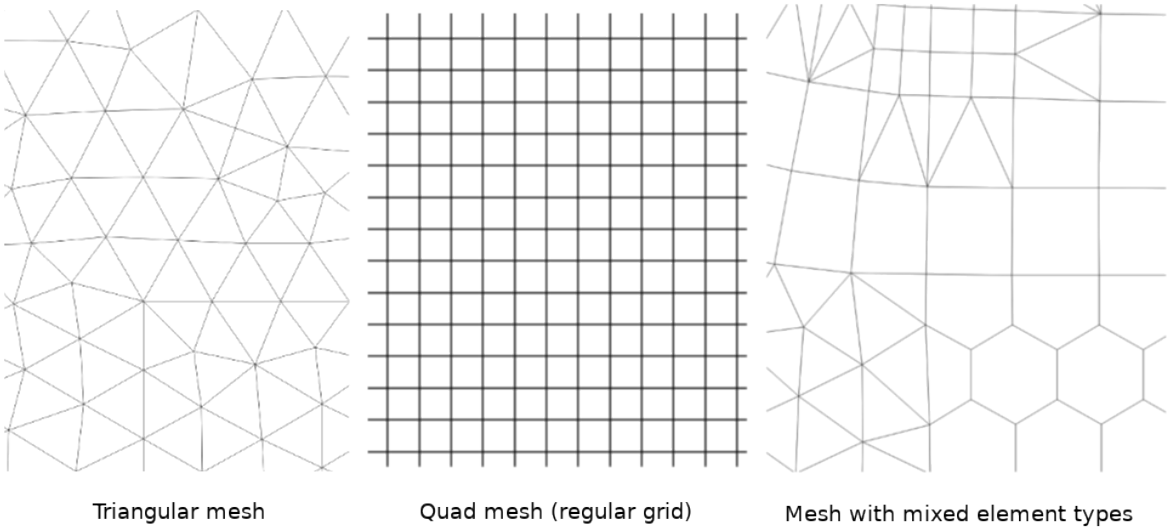


Figura 14.1: Diferentes tipos de Malla

Una malla provee información acerca de su estructura espacial. Además, la malla puede tener conjuntos de datos (grupos) asociados que asignan un valor a cada vértice. Por ejemplo, se puede tener una malla triangular con vértices numerados como se muestra en la imagen abajo:

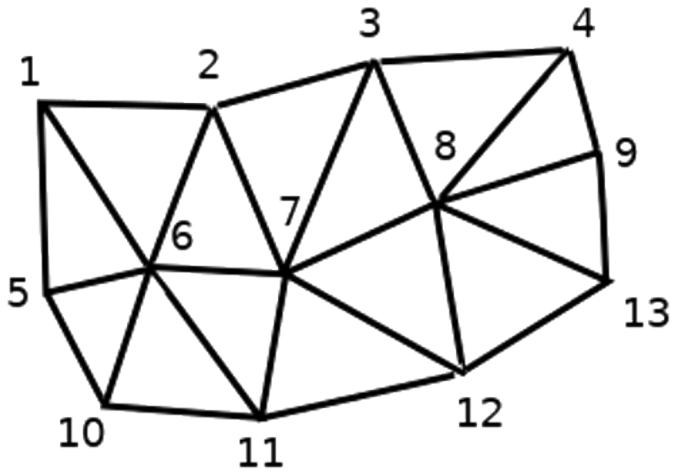


Figura 14.2: Malla triangular con vértices numerados

Cada vértice puede tener asociado differentes conjuntos de datos (típicamente múltiples cantidades), y esos conjuntos de datos pueden tener tambien una dimension temporal. Así, un solo achivo puede contener múltiples conjuntos de datos.

La siguiente tabla brinda una idea acerca de la información que puede ser almacenada en un conjunto de datos de malla. Las columnas de la tabla representan índices de los vértices de la malla, cada fila representa un conjunto de datos. Cada conjunto de datos puede ser de un tipo de datos diferente. En este caso, se almacena velocidad del viento a 10 metros en momentos determinados (t1,t2,t3).

De manera similar, el conjunto de datos de malla puede almacenar valores de vectores para cada vértice. Por ejemplo, vectores de dirección del viento en tiempos dados:

Tabla 14.1: Ejemplo de conjunto de datos de malla

Viento a 10 metros	1	2	3	...
Velocidad a 10 metros en tiempo t=t1	17251	24918	32858	…
Velocidad a 10 metros en tiempo t=t2	19168	23001	36418	…
Velocidad a 10 metros en tiempo t=t3	21085	30668	17251	…
…	…	…	…	…
Dirección de viento a 10 metros en tiempo t=t1	[20,2]	[20,3]	[20,4.5]	…
dirección del viento a 10m en tiempo=t2	[21,3]	[21,4]	[21,5.5]	…
dirección del viento a 10 metros en tiempo=t3	[22,4]	[22,5]	[22,6.5]	…
…	…	…	…	…

Se puede visualizar datos al asignar colores a los valores (de manera similar a como se renderiza un raster *Singleband pseudocolor* raster rendering) e interpolar datos entre vértices de acuerdo con la topología de la malla. Es común que algunas cantidades sean vectores 2D en lugar de un solo valor escalar (por ejemplo, dirección del viento). Para tales cantidades es deseable desplegar flechas indicando las direcciones.

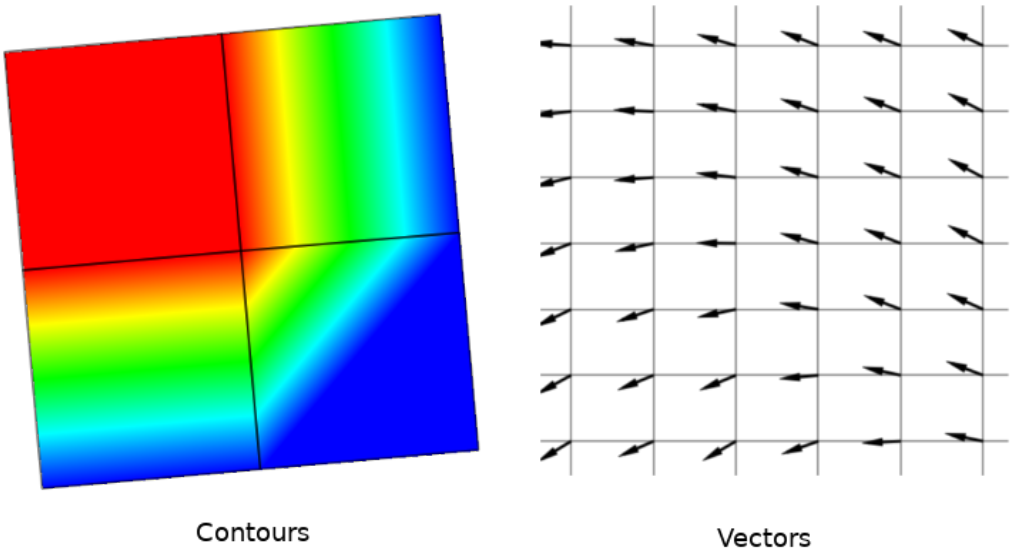


Figura 14.3: Posible visualización de Datos de Malla

14.2 Formatos soportados

QGIS accede a los datos de malla utilizando los controladores MDAL, y soporta de forma nativa una variedad de formatos. Que QGIS pueda editar una capa de malla depende del formato y del tipo de estructura de la malla.

Para cargar un conjunto de datos de malla en QGIS, use la pestaña *Mesh* en el diálogo *Data Source Manager*. Lea *Cargando una capa de malla* para mas detalles.

14.3 Propiedades del Conjunto de Datos tipo Malla

El cuadro de diálogo *Layer Properties* para una capa de malla proporciona configuraciones generales para administrar grupos de conjuntos de datos de la capa y su representación (grupos de conjuntos de datos activos, simbología, representación 2D y 3D). También proporciona información sobre la capa.

Para acceder al diálogo:guilabel:*Propiedades de la capa*:

- En el panel *Capas* , doble-click en la capa o click-derecho y selecciona *Propiedades…* en el menú emergente;
- Vaya al menú *Capa* ► *Propiedades de capa…* cuando la capa este seleccionada.

El diálogo de malla *Layer Properties* proporciona las siguientes secciones:

Tabla 14.2: Pestañas de las propiedades de la capa de malla

Information	*Source*	*Symbology* ([1])
Labels ([1])	*3D View* ([1])	*Rendering*
Temporal	*Elevation*	*Metadata*

Nota

La mayoría de las propiedades de una capa de malla se pueden guardar o cargar desde un `.qml` usando el menú *Style* en la parte inferior del cuadro de diálogo. Más detalles en *Administrando Estilos Personalizados*.

[1] Also available in the *Layer styling panel*

14.3.1 Propiedades de información

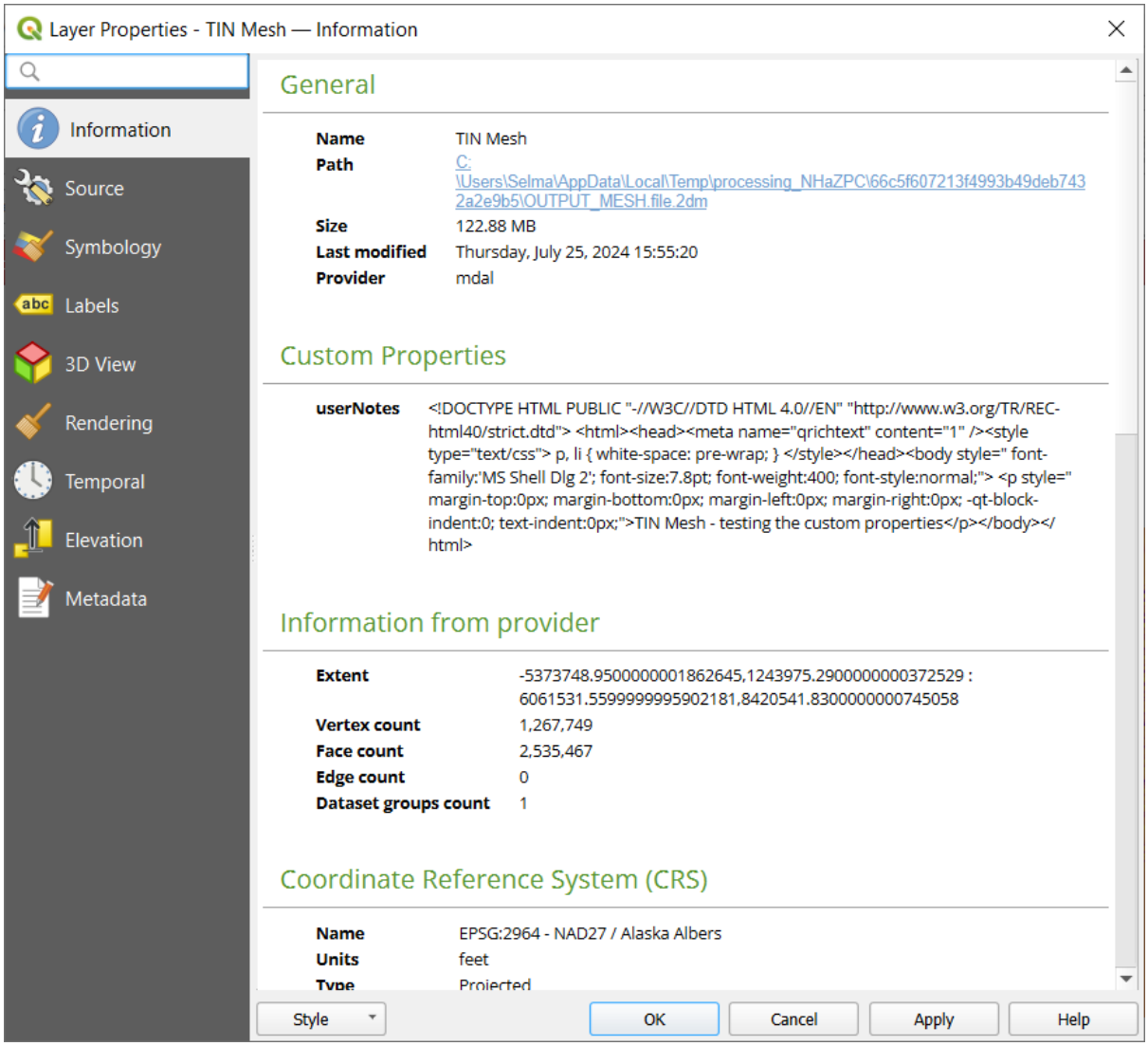


Figura 14.4: Propiedades de información de la capa de malla:

La pestaña *Information* es de solo lectura y representa un lugar interesante para obtener rápidamente información resumida y metadatos para la capa actual. La información proporcionada es:

- general, como el nombre en el proyecto, la ruta de origen, la lista de archivos auxiliares, el tiempo y el tamaño del último guardado, el proveedor utilizado
- propiedades personalizadas, utilizadas para almacenar en el proyecto activo información adicional sobre la capa. Las propiedades personalizadas por defecto pueden incluir *notas de capa*. Se pueden crear y gestionar más propiedades utilizando PyQGIS, concretamente a través del método setCustomProperty().
- según el proveedor de la capa: cuentan los grupos de extensiones, vértices, caras, aristas y/o conjuntos de datos
- el sistema de referencia de coordenadas: nombre, unidades, método, precisión, referencia (es decir, si es estático o dinámico)
- extraído de relleno *metadata*: acceso, extensiones, enlaces, contactos, historial…

14.3.2 Propiedades de fuente

La *pestaña Fuente* muestra información básica sobre la malla seleccionada, que incluye:

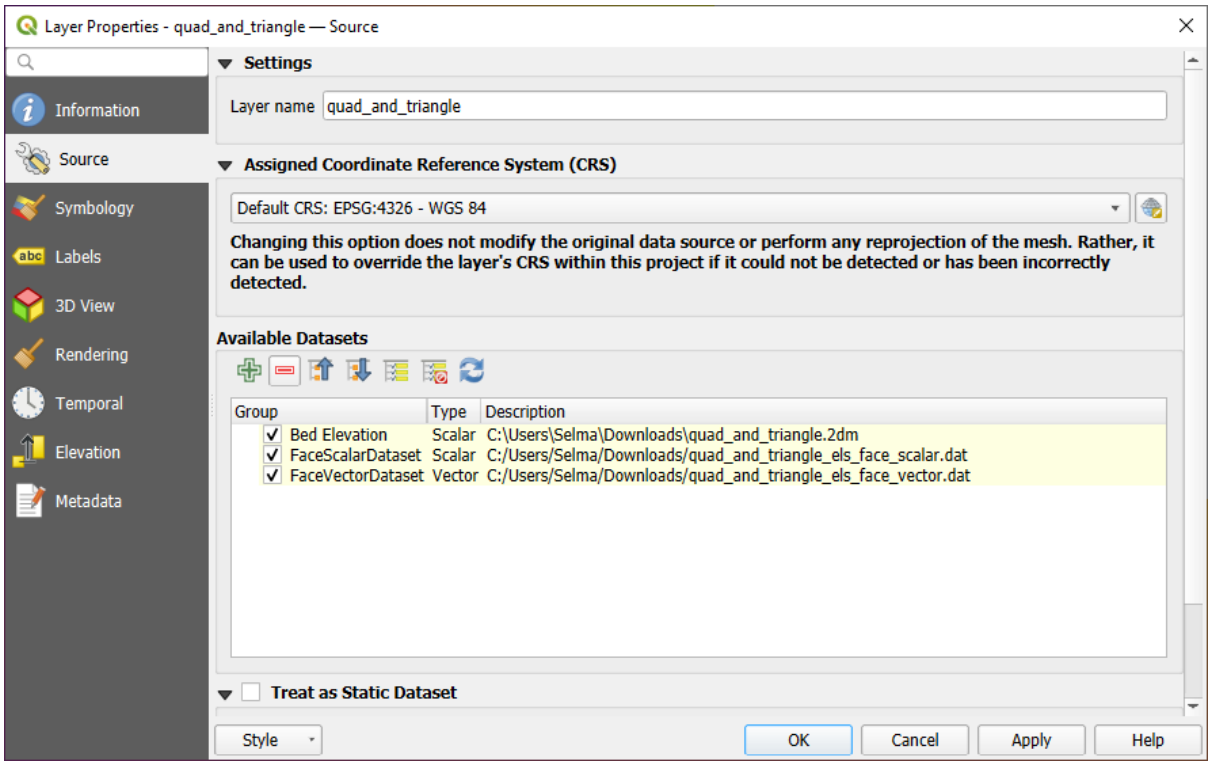


Figura 14.5: Propiedades de origen de la capa de malla

- el nombre de la capa para mostrar en el *Layers* panel
- establecer el sistema de referencia de coordenadas: muestra la capa:ref:*Assigned Coordinate Reference System (CRS) <layer_crs>*. Puede cambiar el CRS de la capa seleccionando uno usado recientemente en la lista desplegable o haciendo clic en el botón *Seleccionar CRS* (ver *Selector del Sistema de Coordenadas de Referencia*). Use este proceso solo si el CRS aplicado a la capa es incorrecto o si no se aplicó ninguno.
- El cuadro *Conjuntos de datos disponibles* enumera todos los grupos (y subgrupos) de conjuntos de datos en la capa de malla, con su tipo y descripción en una vista de árbol. Tanto conjuntos de datos regulares (es decir, sus datos se almacenan en el archivo) como conjuntos de datos virtuales (which are *calculated on the fly*) listados.
  - Use the Assign Extra Dataset to Mesh button to add more groups to the current mesh layer. You can add dataset group to a mesh layer with the same name, but not from the same URI. Dataset group names are automatically renamed to «Original Name_Number».
  - Use Remove Extra Dataset from Mesh to remove additional datasets groups from the mesh layer. Note that only dataset groups not associated with the mesh source file can be removed.
  - Collapse all and Expand all the dataset tree, in case of embedded groups
  - Si está interesado en algunos conjuntos de datos, puede desmarcar los demás y hacer que no estén disponibles en el proyecto
  - Haga doble clic sobre un nombre y podrá cambiar el nombre del conjunto de datos.
  - Reset to defaults: checks all the groups and renames them back to their original name in the provider.
  - Haga clic con el botón derecho en un grupo de conjuntos de datos virtuales y podrá:

    * *Eliminar grupo de conjuntos de datos* desde el proyecto
    * *Guardar grupo de conjuntos de datos como…* un archivo en el disco, a cualquier formato compatible. El nuevo archivo se mantiene asignado a la capa de malla actual en el proyecto.

- Comprobar el grupo |no comprobado| *Tratar como un conjunto de datos estáticos* le permite ignorar las propiedades de *mapa de navegación temporal* mientras renderiiza la capa de malla. Por cada grupo de conjunto de datos activo (como el seleccionado en la pestaña |simbología| *Simbología* ►  *Conjuntos de Datos*), puede:
  - set to *Ninguno*: el grupo de conjuntos de datos no se muestra en absoluto
  - *Mostrar conjunto de datos*: por ejemplo, para el conjunto de datos «bed elevation» que no tiene en cuenta el tiempo
  - extract a particular date time: the dataset matching the provided time is rendered and stays fixed during map navigation.

14.3.3 Propiedades de simbología

Click en el botón  *Simbología* para activar el diálogo. Las propiedades de simbología se dividen en varias pestañas:

- *Datasets*
- *Datasets*
- *Vectors*
- *Rendering*
- *Stacked mesh averaging method*

Conjuntos de datos

The tab  Datasets ¡Es el lugar principal para controlar y establecer qué conjuntos de datos se utilizarán para la capa. Presenta los siguientes elementos:

- *Grupos* disponibles en el conjunto de datos de malla, con si proporcionan:
  -  conjunto de datos escalares
  - o  Conjunto de datos vectoriales: de forma predeterminada, cada conjunto de datos vectoriales tiene un conjunto de datos escalar que representa su magnitud generado automáticamente.

  Haga clic en el icono junto al nombre del conjunto de datos para seleccionar el grupo y el tipo de datos que desea representar.
- *Metadatos de grupo(s) de conjuntos de datos seleccionados*, con detalles en:
  - el tipo de malla: bordes o caras
  - el tipo de datos: vértices, aristas, caras o volumen
  - si es de tipo vectorial o no
  - el nombre original en la capa de malla
  - la unidad, en su caso
- *blending mode* disponibles para los conjuntos de datos seleccionados.

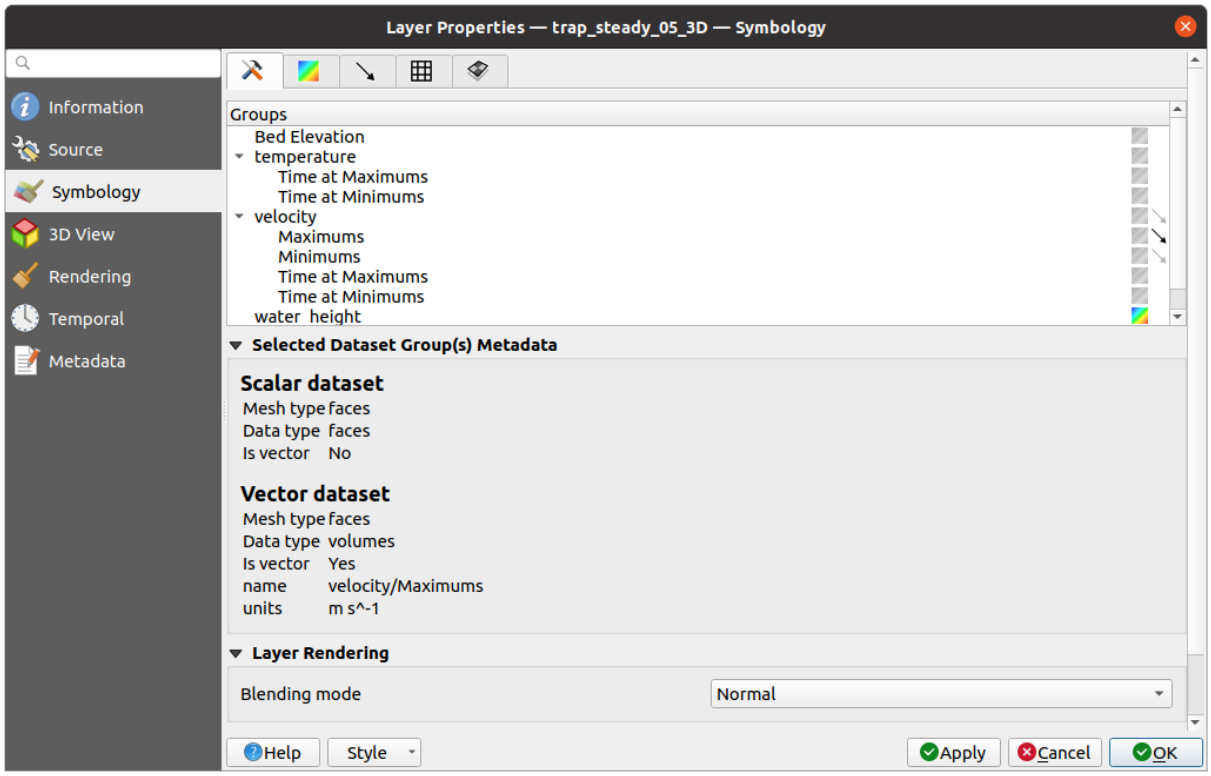


Figura 14.6: Conjuntos de datos de capa de malla

Puede aplicar la simbología al vector y/o grupo escalar seleccionado usando las siguientes pestañas.

Simbología de Contornos

Nota

The Contours La pestaña se puede activar solo si se ha seleccionado un conjunto de datos escalares en :guilabel:pestaña *Conjuntos de datos*.

En la pestaña Contours puede ver y cambiar las opciones de visualización actuales de contornos para el grupo seleccionado, como se muestra en Figura 14.7 below:

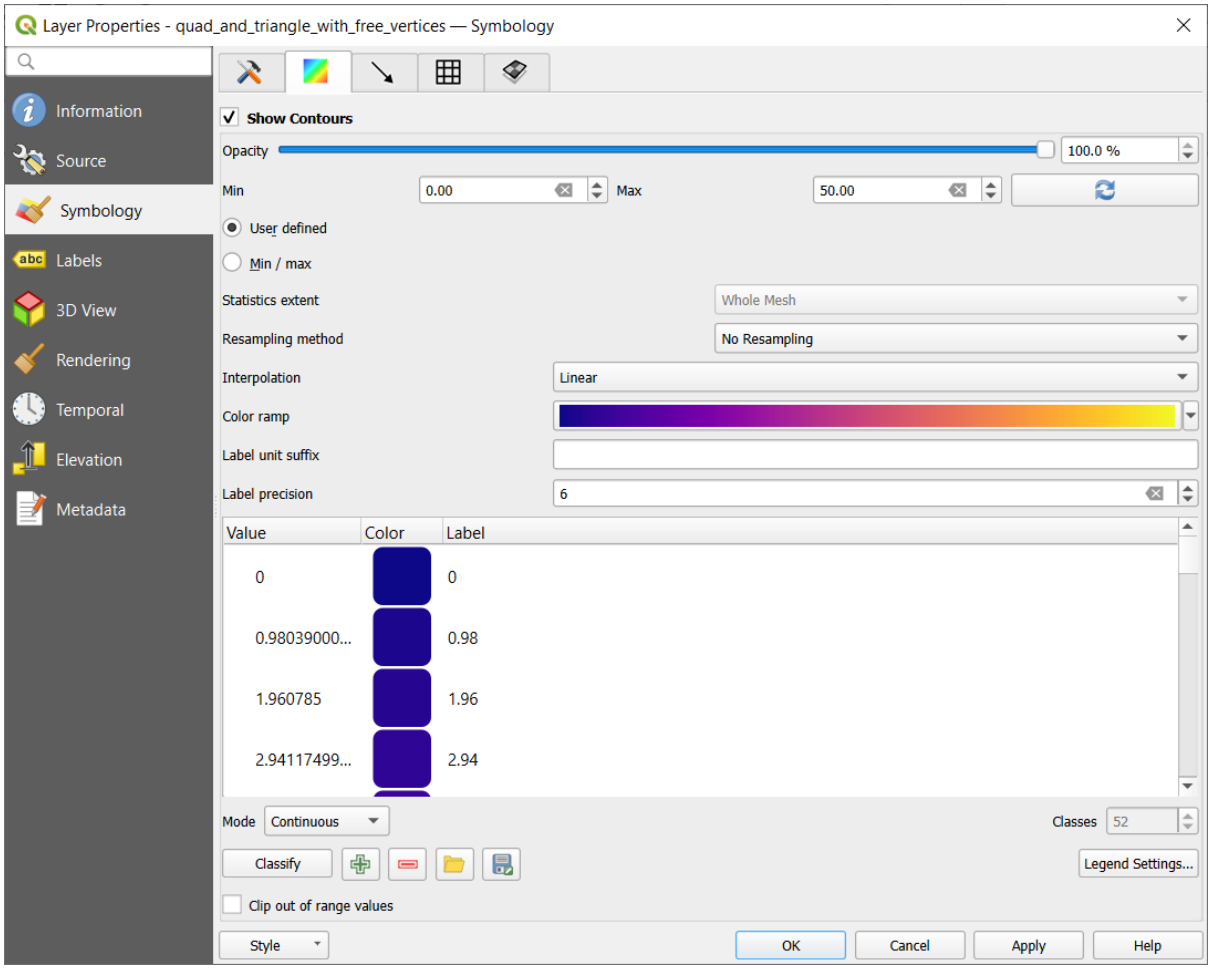


Figura 14.7: Aplicando Estilo a Contornos en una Capa de Malla de Datos

- Para la malla 1D, establezca el *Ancho del trazo* de los bordes. Esto puede ser un tamaño fijo para todo el conjunto de datos, o variar a lo largo de la geometría (más detalles con el f:*interpolated line renderer <interpolated_line_symbol>*)
- Use el control deslizante o el cuadro giratorio para establecer la *Opacidad* del grupo actual, si es un tipo de malla 2D.
- *User Defined* allows you to enter the range of values you want to represent on the current group: use Load to fetch the min and max values of the current group or enter custom values if you want to exclude some.
- Select *Min/Max* to set the renderer's minimum and maximum values based on the chosen extent.
- *Statictics extent* can be *Whole mesh*, *Current canvas* or *Updated canvas*. *Updated canvas* means that min/max values used for the rendering will change with the canvas extent (dynamic stretching).
- Para mallas 2D/3D, seleccione el *Método de remuestreo* para interpolar los valores de los vértices circundantes a las caras (o de las caras circundantes a los vértices) utilizando el método *Promedio de vecinos*. Dependiendo de si el conjunto de datos está definido en los vértices (respectivamente en las caras), QGIS establece esta configuración por defecto en el método *Ninguno* (respectivamente *Promedio de vecinos*) para usar valores en los vértices y mantener el método predeterminado prestación suave.
- Clasifique el conjunto de datos utilizando la clasificación ref:*color ramp shader <color_ramp_shader>*

Simbología de vectores

Nota

La pestaña Vectores sólo puede activarse si se ha seleccionado un conjunto de datos vectoriales en la pestaña *Conjuntos de datos*. Haga clic en el icono Vectores situado a la derecha de la pestaña *Conjuntos de datos*.

En la pestaña Vectores puede ver y cambiar las opciones de visualización actuales de los vectores para el grupo seleccionado, como se muestra en Figura 14.8:

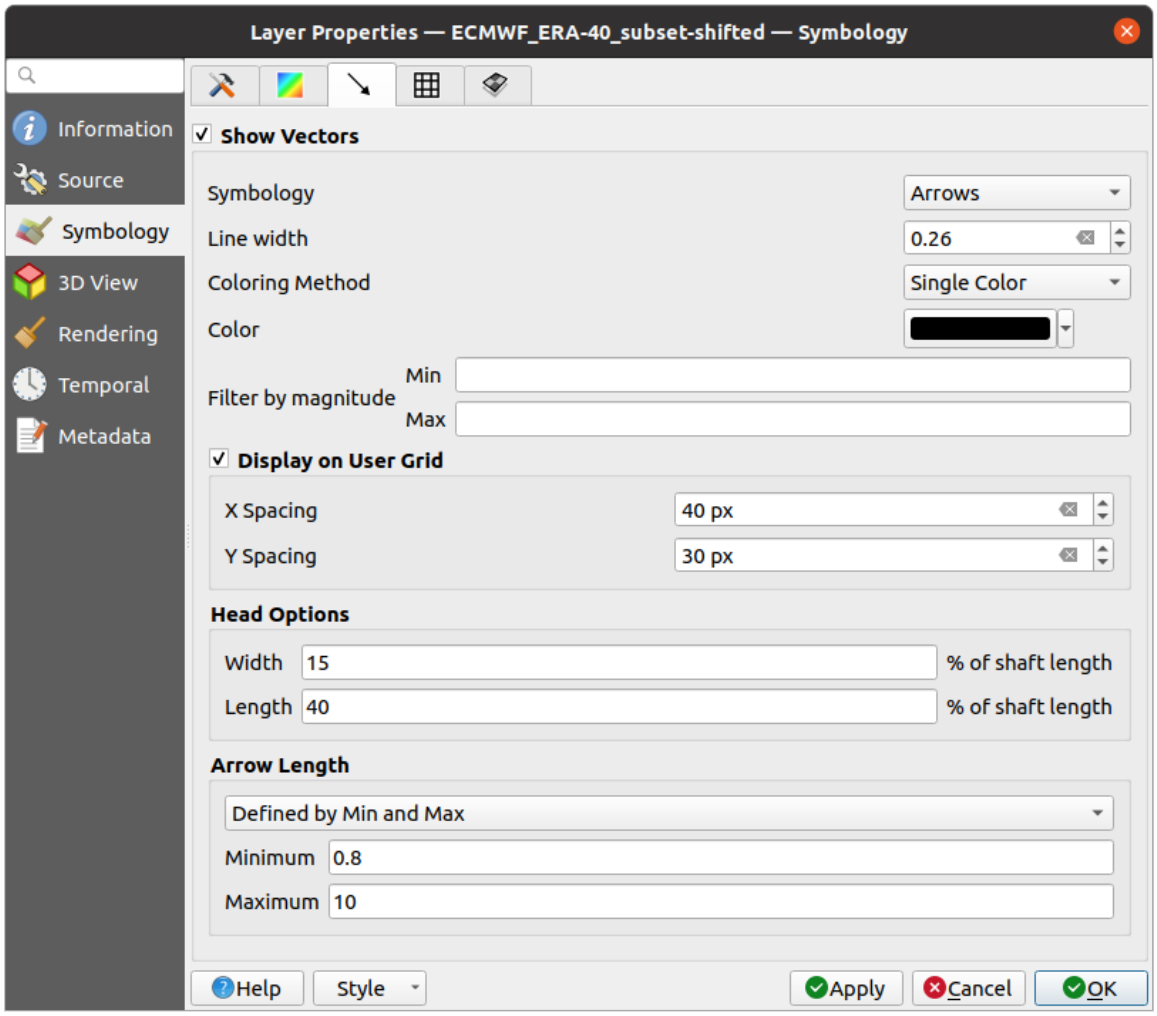


Figura 14.8: Dar estilo a los vectores en una capa de malla con flechas

El conjunto de datos de vector de malla se puede diseñar usando varios tipos de *Simbología*:

- **Flechas**: los vectores se representan con flechas en el mismo lugar en el que están definidos en el conjunto de datos brutos (es decir, en los nodos o en el centro de los elementos) o en una cuadrícula definida por el usuario (por tanto, están distribuidos uniformemente). La longitud de la flecha es proporcional a la magnitud de la flecha tal y como se define en los datos brutos, pero se puede escalar por varios métodos.
- **Líneas de corriente**: los vectores se representan con líneas de corriente sembradas desde los puntos de inicio. Los puntos de siembra pueden partir de los vértices de la malla, de una grilla de usuario o aleatoriamente.

- **Trazas**: una animación más agradable de las líneas de corriente, el tipo de efecto que obtienes cuando arrojas arena al azar al agua y ves dónde fluyen los elementos de arena.
- **Barras de viento**: los vectores se representan con barras de viento, una forma habitual de representar la velocidad y dirección del viento.

Las propiedades disponibles dependen de la simbología seleccionada, como se muestra en la siguiente tabla.

Tabla 14.3: Disponibilidad y significado de las propiedades de la simbología de vectores

Label, Description and Properties	Flecha	Streamlines	Trazos	Barbas de viento
Line width: Width of the vector representation	☑	☑	☑	☑
Coloring method: • a *Color único* asignado a todos los vectores • o un color variable basado en la magnitud de los vectores, usando un:ref:*Color ramp shader <color_ramp_shader>*	☑	☑	☑	☑
Filter by magnitude: Only vectors whose length for the selected dataset falls between a *Min* and *Max* range are displayed	☑	☑		☑
Display on user grid: Places the vector on a grid with custom *X spacing* and *Y spacing* and interpolates their length based on neighbours	☑	☑		☑
Head options: *Length* and *Width* of the arrow head, as a percentage of its shaft length	☑			
Arrow length: • **Definido por Min y Max**: Usted especifica la longitud mínima y máxima de las flechas, QGIS interpolará su tamaño en función de la magnitud del vector subyacente • **Escala a magnitud**: la longitud de la flecha es proporcional a la magnitud de su vector • **Corregido**: todos los vectores se muestran con la misma longitud	☑			
Streamlines seeding method: • **En malla/cuadrícula**: se basa en la cuadrícula del usuario para mostrar los vectores • **Aleatoriamente*: la colocación de vectores se realiza aleatoriamente con respecto a una cierta densidad		☑		
Particles count: The amount of «sand» you want to throw into visualisation			☑	
Max tail length: The time until the particle fades out			☑	

continúe en la próxima página

Tabla 14.3 – proviene de la página anterior

Label, Description and Properties	Flecha	Streamlines	Trazos	Barbas de viento
Wind Barbs: • **Longitud**: Especifica la longitud de las barbas de viento. Elige las unidades en un menú desplegable. • **Unidades de datos**: Especifica las unidades de los datos de velocidad del viento. Las barras de viento codifican el viento en nudos (millas náuticas por hora). Por lo tanto, debe seleccionar las unidades en las que están sus datos, o utilizar un *Multiplicador* personalizado para convertir las unidades de sus datos a nudos. Esto garantiza que la velocidad del viento se represente con precisión.				☑

Representación

En la pestaña Rendering QGIS ofrece posibilidades para mostrar y personalizar la estructura de malla. *Ancho de línea* y *Color de línea* se pueden configurar para representar:

- los bordes para mallas 1D
- Para mallas 2D:
  - *Representación de malla nativa*: muestra caras y bordes originales de la capa
  - *Representación de malla triangular*: agrega más bordes y muestra las caras como triángulos

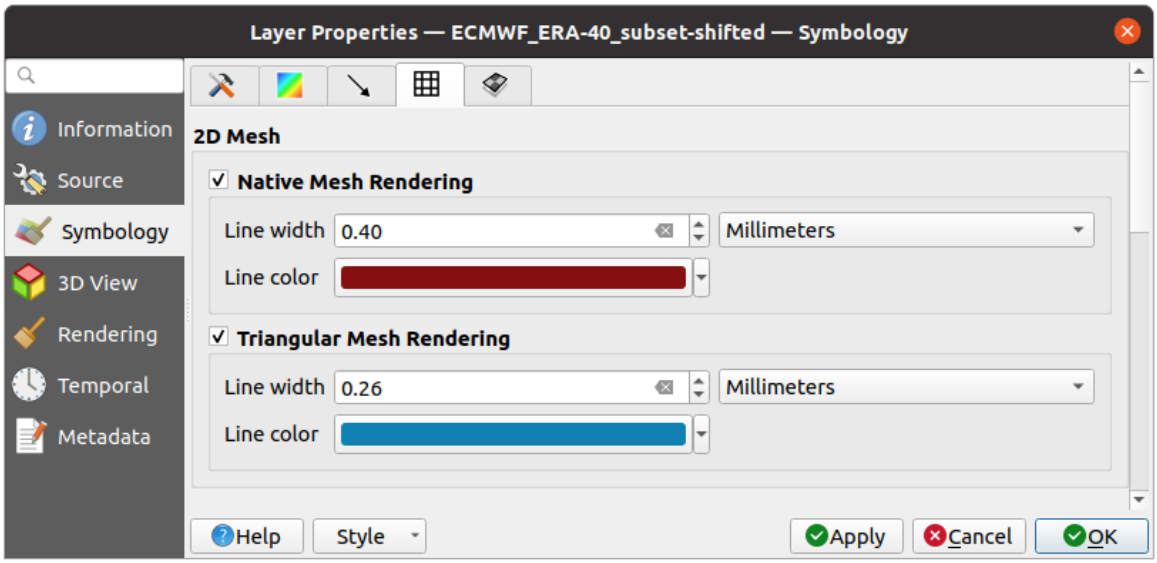


Figura 14.9: Representación de malla 2D

Método de promedio de malla apilada

Las mallas en capas 3D consisten en múltiples mallas 2D no estructuradas apiladas, cada una extruida en la dirección vertical («niveles») por medio de una coordenada vertical. Los vértices y las caras tienen la misma topología en cada nivel vertical. Los valores generalmente se almacenan en los volúmenes que se apilan regularmente sobre la malla base 2d. Para visualizarlos en un lienzo 2D, debe convertir los valores de los volúmenes (3d) en valores de las caras (2d) que se pueden mostrar en la capa de malla. El **|promedio de malla|** Método de promedio de malla apilada proporciona diferentes métodos de promedio/interpolación para manejar esto.

Puede seleccionar el método para obtener los conjuntos de datos 2D y los parámetros correspondientes (índice de nivel, valores de profundidad o altura). Para cada método, se muestra un ejemplo de aplicación en el cuadro de diálogo, pero puede leer más sobre los métodos en https://fvwiki.tuflow.com/Depth_Averaging_Results.

14.3.4 Propiedades de etiquetas

La pestaña *Etiquetas* le ofrece un etiquetado dinámico de los vértices de la malla y de las caras nativas de la malla basado en propiedades geométricas y expresiones personalizadas. También se puede acceder a este cuadro de diálogo desde el panel *Estilos de capa*.

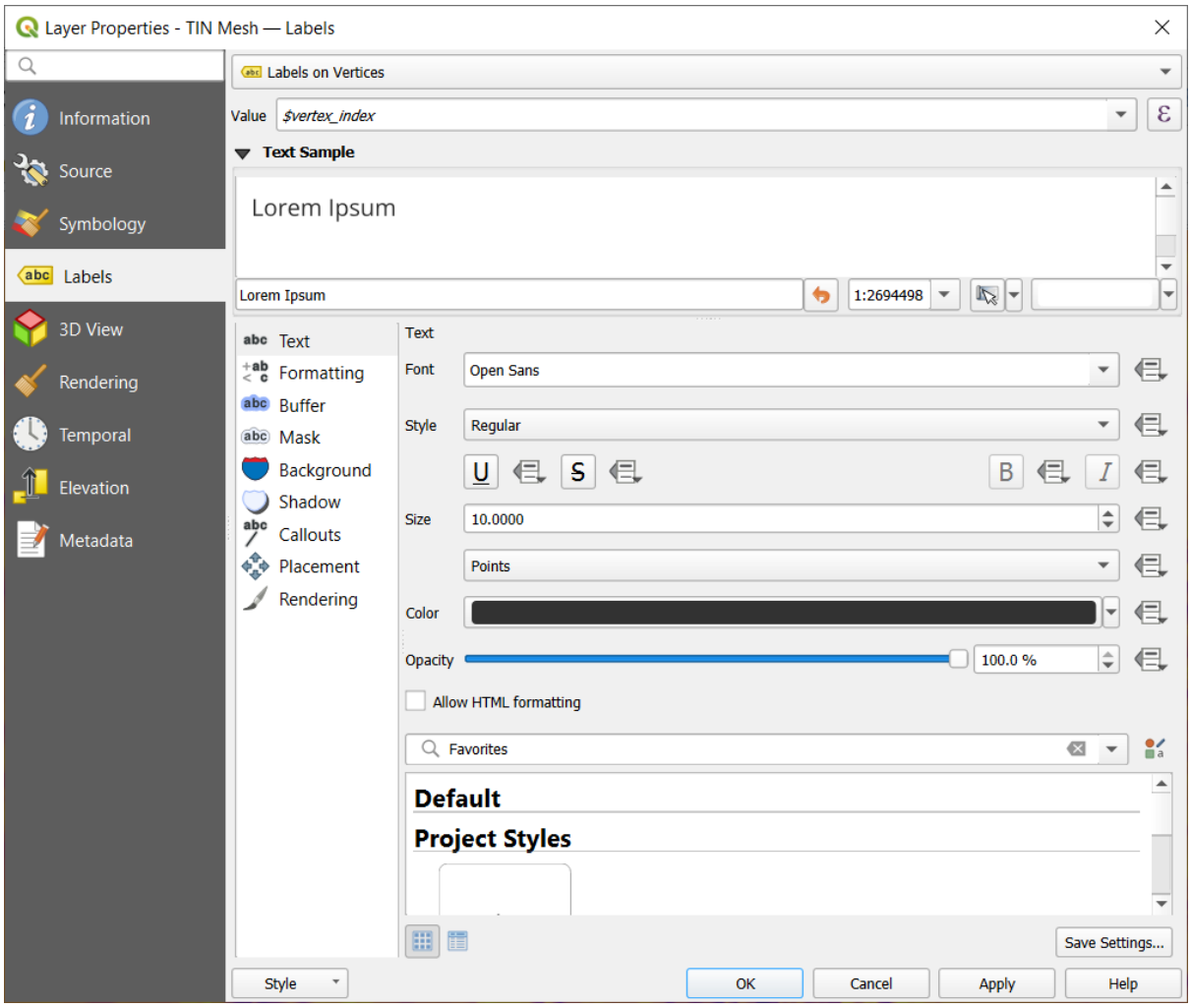


Figura 14.10: Propiedades de las etiquetas de malla

Elija el método de etiquetado en la lista desplegable. Los métodos disponibles son:

- *Sin etiquetas*: el valor por defecto, que no muestra etiquetas de la capa.

- *Etiquetas en vértices*: etiqueta los vértices de la malla como un objeto espacial punto.
- *Etiquetas en caras*: etiqueta las caras de la malla como una característica poligonal.

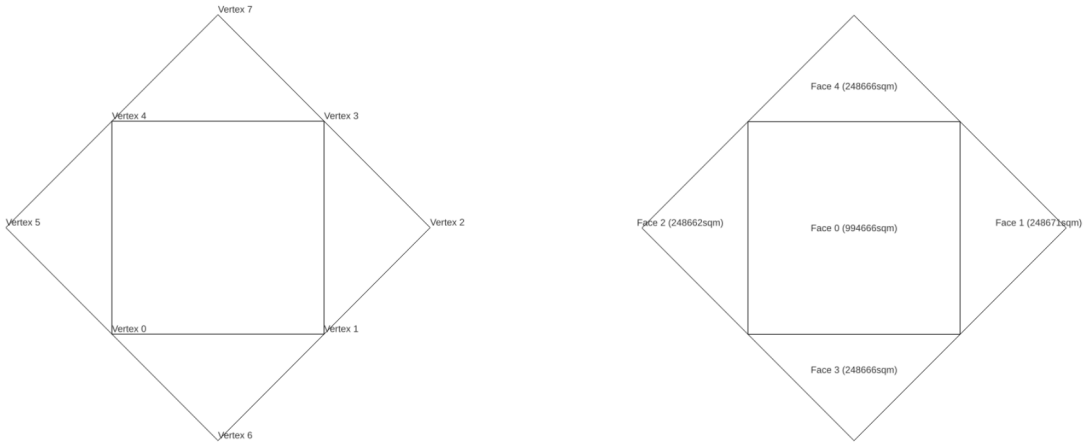


Figura 14.11: Métodos de etiquetado de mallas

Una vez seleccionado el método de etiquetado, abra Diálogo de expresiones y elija una *función de malla*. Puede combinar las funciones de malla con otras funciones de la *lista* para modificar la etiqueta.

A continuación se muestran las opciones para personalizar las etiquetas, en varias pestañas:

- *Text*
- *Formatting*
- *Buffer*
- *Mask*
- *Background*
- *Shadow*
- *Callouts*
- *Placement*
- *Rendering*

La descripción de cómo configurar cada propiedad se expone en *Ajustando una etiqueta*.

14.3.5 Propiedades de visor 3D

Las capas de malla se pueden usar como *terreno en una vista de mapa 3D* en base a los valores de sus vértices Z.

Desde el tabular de propiedades *Vista 3D*, es también posible renderizar el conjunto de datos de la capa de malla en la misma vista 3D. Por lo tanto, el componente vertical de los vértices puede ser igualado a los valores del conjunto de datos (por ejemplo, nivel de la superficie del agua) y la textura de la malla puede ser ajustada para renderizar otros valores del conjunto de datos con color sombreado (por ejemplo velocidad).

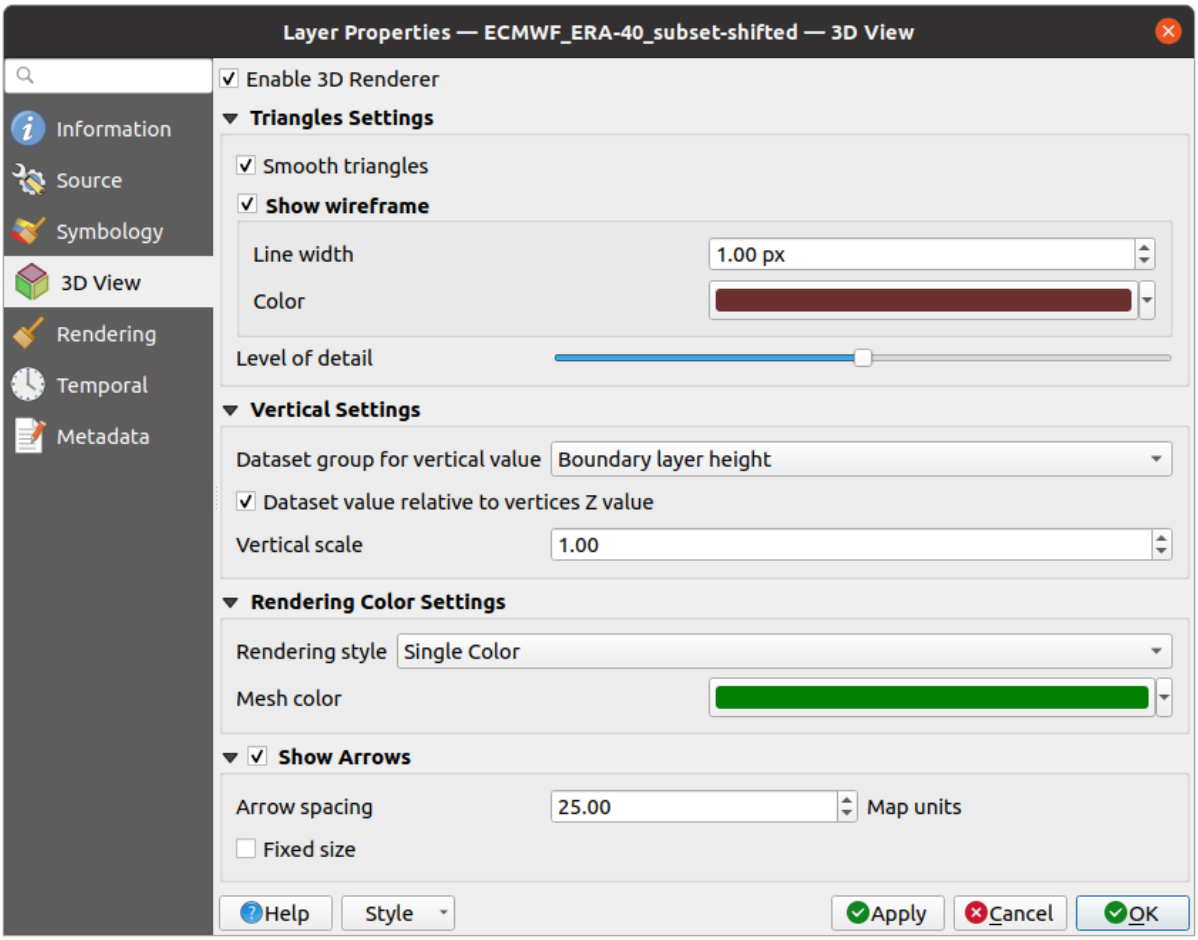


Figura 14.12: Propiedades 3D del conjunto de datos de malla

Check *Habilite 3D Renderer* y podrá editar las siguientes propiedades:

- En *Configuración del triángulo*
  - *Triángulos suaves*: Los ángulos entre triángulos consecutivos se suavizan para una mejor representación 3D
  - *Muestre la estructura alámbrica* en la que puede configurar *Ancho de línea* y *Color*
  - *Nivel de detalle": controla cómo :ref:`simplified <meshrendering>* la capa de malla para renderizar debe ser. En el extremo derecho, está la malla base, y cuanto más a la izquierda, más se simplifica la capa y se representa con menos detalles. Esta opción sólo está disponible si la opción *Simplificar malla* en la pestaña *Renderizado* está activada.
- *Configuración vertical* para controlar el comportamiento de la componente vertical de los vértices de los triángulos renderizados:
  - *Grupo de conjuntos de datos para valor vertical*: el grupo de conjuntos de datos que se usará para el componente vertical de la malla
  - *Valor del conjunto de datos relativo al valor Z de los vértices*: si considerar los valores del conjunto de datos como coordenadas Z absolutas o relativos al valor Z nativo de los vértices
  - *Escala vertical*: el factor de escala a aplicar a los valores Z del conjunto de datos
- *Ajustes de color de renderizado* con un *Estilo de renderizado* que se puede basar en el sombreador de rampa de color establecido en *Simbología de Contornos* (*Shader de rampa de color de contorno 2D*) o como *Color único* con un *Color de malla* asociado

- *Mostrar flechas*: muestra flechas en la entidad 3D del dataset de capa de malla, basado en el mismo grupo de dataset vectorial usado en el:ref:*vector 2D rendering <mesh_symbology_vectors>*. Se muestran utilizando la configuración de color 2D. También es posible definir el *Espaciado entre flechas* y, si es de *Tamaño fijo* o escalado en magnitud. Esta configuración de espaciado define también el tamaño máximo de las flechas porque las flechas no se pueden superponer.

14.3.6 Propiedades de representación

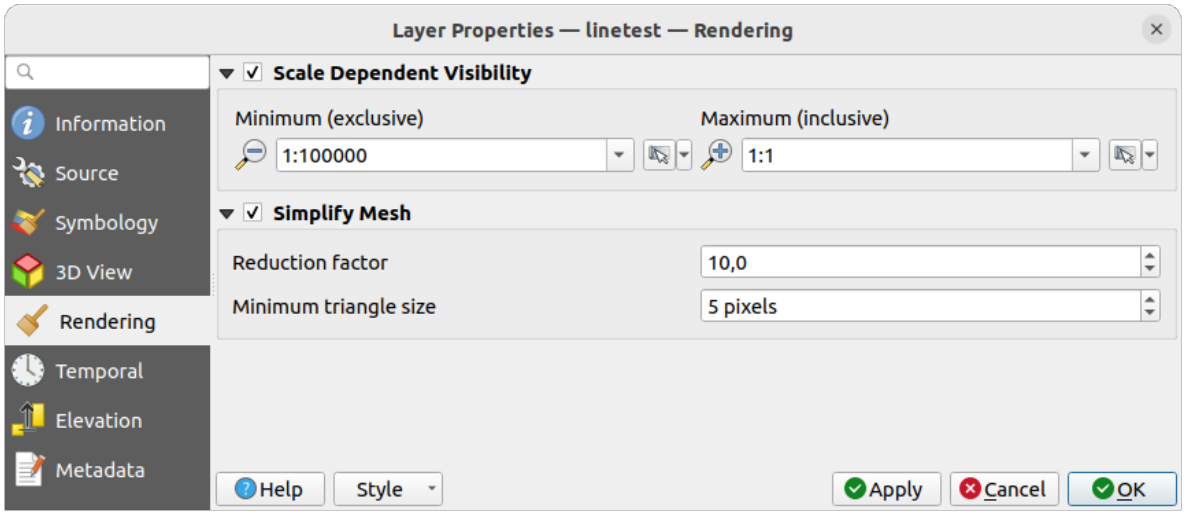


Figura 14.13: Propiedades de renderizado de malla

En el cuadro de grupo *Scale dependent visibility*, puede establecer la escala *Maximum (inclusive)* y *Minimum (exclusive)*, definiendo un rango de escala en el que los elementos de la malla serán visibles. Fuera de este rango, se ocultan. El botón Set to current canvas scale le ayuda a utilizar la escala actual del lienzo del mapa como límite del rango de visibilidad. Véase *Selector de escala de visibilidad* para más información.

Nota

También puede activar la visibilidad dependiente de la escala en una capa desde el panel *Layers*: haga clic con el botón derecho del ratón en la capa y, en el menú contextual, seleccione *Definir la visibilidad de la escala de capas*.

Dado que las capas de malla pueden tener millones de caras, su representación a veces puede ser muy lenta, especialmente cuando todas las caras se muestran en la vista y son demasiado pequeñas para verse. Para acelerar el renderizado, puede simplificar la capa de malla, lo que da como resultado una o más mallas que representan diferentes *levels of detail* y seleccione en qué nivel de detalle desea que QGIS represente la capa de malla. Tenga en cuenta que la malla simplificada contiene solo caras triangulares.

Desde la pestaña *Rendering* check *Simplify mesh* and set:

- a *Factor de reducción*: Controla la generación de sucesivos niveles de mallas simplificadas. Por ejemplo, si la malla base tiene 5M caras, y el factor de reducción es 10, la primera malla simplificada tendrá aproximadamente 500 000 caras, la segunda 50 000 caras, la tercera 5000,… Si un factor de reducción mayor lleva rápidamente a mallas más simples (es decir, con triángulos de mayor tamaño), produce también menos niveles de detalle.
- *Tamaño mínimo del triángulo*: el tamaño medio (en píxeles) de los triángulos que se permite mostrar. Si el tamaño medio de la malla es inferior a este valor, se activa la representación de una malla de menor nivel de detalles.

14.3.7 Propiedades Temporales

El :guilabel:La pestaña *Temporal* proporciona opciones para controlar el renderizado de la capa a lo largo del tiempo. Permite mostrar dinámicamente los valores temporales de los grupos de conjuntos de datos habilitados. Una representación tan dinámica requiere la *temporal navigation* to estar habilitado sobre el lienzo del mapa.

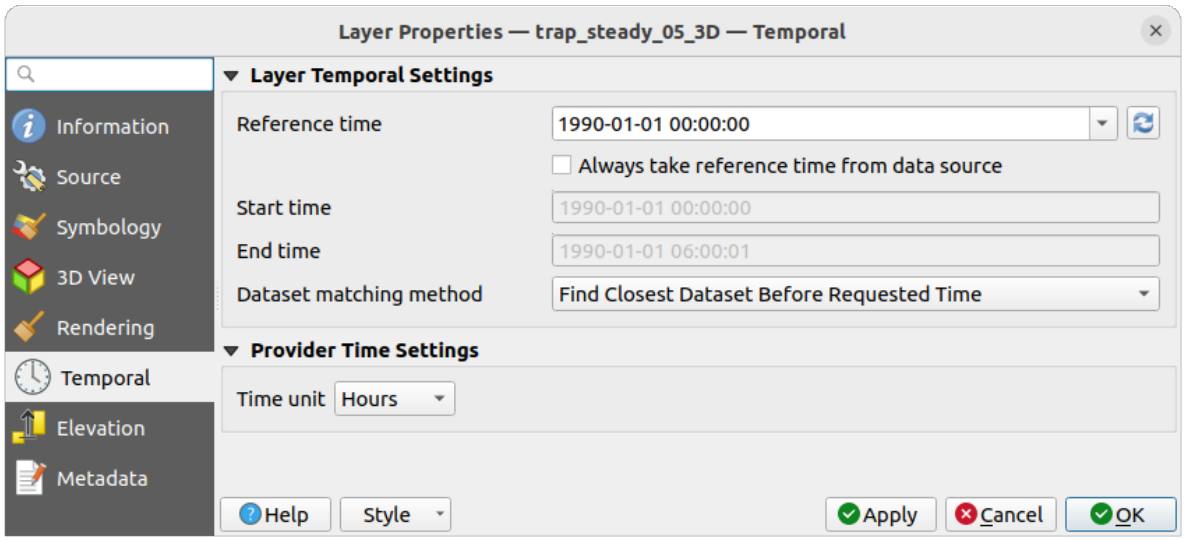


Figura 14.14: Propiedades temporales de malla

Configuración temporal de capa

- *Hora de referencia* del grupo de conjuntos de datos, como fecha y hora absolutas. Por defecto, QGIS analiza la capa de origen y devuelve la primera hora de referencia válida en el grupo de conjuntos de datos de la capa. Si no está disponible, el valor será fijado por el rango de tiempo del proyecto o se retrocederá a la fecha actual. La *Hora de inicio* y la *Hora de finalización* a considerar se calculan entonces en base al paso de tiempo interno del conjunto de datos.

  Es posible establecer una *Hora de referencia* personalizada (y luego el intervalo de tiempo), y revertir los cambios utilizando el botón Recargar desde el proveedor. Con *Tomar siempre la hora de referencia de la fuente de datos* marcado, se asegura de que las propiedades de tiempo se actualizan desde el fichero cada vez que se recarga la capa o se reabre el proyecto.

- *Método de coincidencia del conjunto de datos*: determina el conjunto de datos a mostrar en el momento dado. Las opciones son *Buscar el conjunto de datos más cercano antes de la hora solicitada* o *Buscar el conjunto de datos más cercano a la hora solicitada (después o antes)*.

Configuración de hora del proveedor

- *Unidad de tiempo* extraída de los datos brutos, o definida por el usuario. Esto se puede usar para alinear la velocidad de la capa de malla con otras capas en el proyecto durante la navegación en el tiempo del mapa. Las unidades soportadas son *Segundos*, *Minutos*, *Horas* y *Días*.

14.3.8 Propiedades de elevación

La pestaña *Elevación* proporciona opciones para controlar las propiedades de elevación de la capa dentro de una *vista de mapa 3D* y *Vista de mapa 2D* y su aparición en los *gráficos de herramienta de perfil*. En concreto, puede configurar cómo se interpretan las alturas de su conjunto de datos:

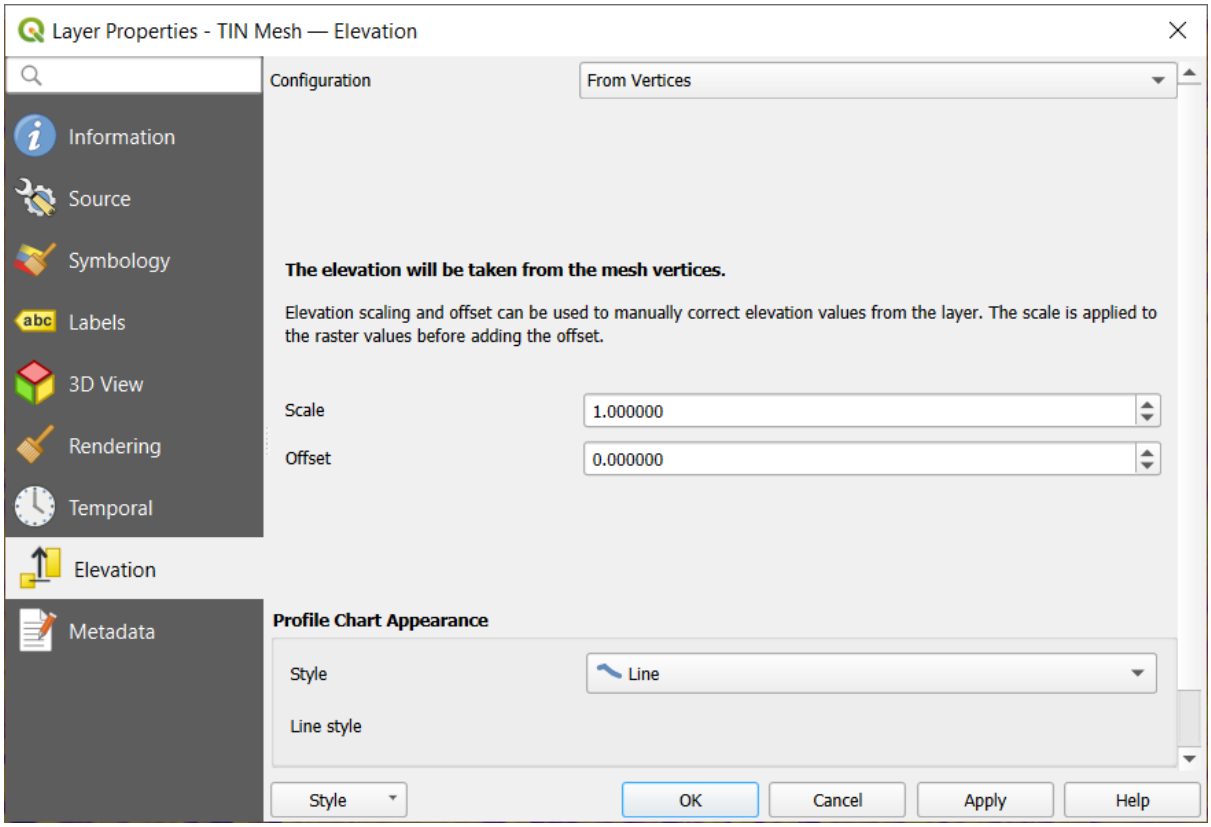


Figura 14.15: Malla Propiedades de elevación

- *De vértices*: cómo deben interpretarse los valores Z de los vértices de la capa de malla como elevación del terreno. Se puede aplicar un factor *Escala* y un factor *Desplazamiento*. Este ajuste está disponible para *Vista de mapa 3D* y *gráficos de herramientas de perfil*.
- *Rango de elevación fijo*: la capa de malla está vinculada a un rango de elevación fijo. Este modo es aplicable cuando una capa tiene una única elevación fija o un rango (rebanada) de valores de elevación. Si se especifica un rango, los valores de la malla se extruirán sobre este rango. Puede establecer los valores de rango de elevación *Inferior* y *Superior* para la capa, y especificar si los *Límites* inferior o superior son inclusivos o exclusivos. Si está activada, la capa sólo será visible en los *mapas 2D filtrados por elevación* cuando el rango de la capa está incluido en el rango Z del mapa.
- *Rango de elevación fijo por grupo*: cada grupo de la capa de malla se asocia a un rango de elevación fijo. Este modo puede utilizarse cuando una capa tiene datos de elevación expuestos a través de diferentes grupos de conjuntos de datos. Esta característica se expone como una tabla editable por el usuario para grupos de conjuntos de datos con valores inferiores y superiores. Puede rellenar los valores inferiores y superiores manualmente o utilizar una expresión *Expresión* para rellenar automáticamente todos los valores de grupo basándose en una expresión. Cuando se activa, la capa se filtrará en la *vista de mapa 2D*, mostrando sólo valores dentro de los rangos del filtro de mapa. En la *vista de mapa 3D* o *gráficas de herramienta perfil*, se mostrará toda la extensión de la capa, ignorando el filtrado.
- *Apariencia del gráfico de perfil*: controla la representación de la elevación de los elementos de malla en el gráfico de perfil. El *Estilo* del perfil se puede establecer como:
  - una *Línea* con un estilo *Línea*

  - una superficie de elevación representada utilizando un símbolo de relleno por encima (*Rellenar por encima*) o por debajo (*Rellenar por debajo*) de la línea curva de elevación. La simbología de la superficie se representa utilizando:
    * un *Estilo de relleno*
    * y una :guilabel:"Límite": la altitud máxima (o mínima) que determina la altura de la superficie de relleno.

14.3.9 Propiedades de metadatos

El *Metadata* le brinda opciones para crear y editar un informe de metadatos en su capa. Consulte *Metadatos* para obtener más información.

14.4 Exploring mesh datasets

To identify mesh elements and their associated datasets, use the *Identify Features* tool from the *Attributes* toolbar. Click on a mesh element in the map canvas and the *Identify Results* panel opens, displaying information about the mesh layer and the identified element. The panel shows:

- Layer name
- Active dataset groups (scalar and/or vector) for the clicked element will be displayed first. On the other hand, inactive dataset groups are shown afterwards. For each group, the following derived information may also be shown, if applicable:
  - **Source** of the dataset (the file where it is stored).
  - **Time step** of the dataset value being displayed. For non-temporal dataset groups, this field remains empty.
  - **Vector x-component** and **Vector y-component** values for vector datasets.

Please note that when temporal navigation is disabled, the results only contain values from the *static dataset groups* defined.

- Geometry information shows the coordinates of the clicked point and may contain derived information in the following:
  - Centroid of the face.
  - Coordinates of a snapped vertex (if the click was snapped to a vertex).
  - Center of a snapped edge (if the click was snapped to an edge).

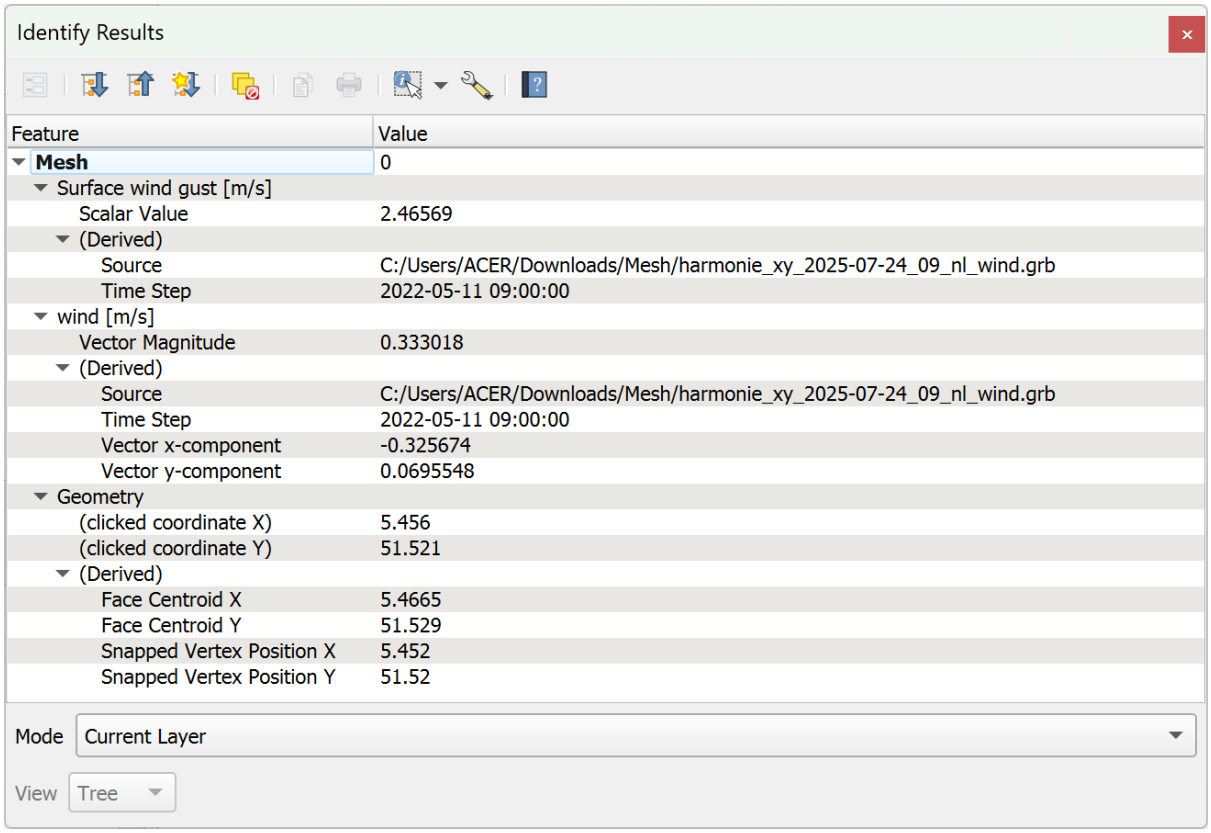


Figura 14.16: Mesh Identify Results panel

14.5 Editando la capa de malla

QGIS permite *crear una capa de malla* desde cero en base a una existente. Usted puede crear/modificar las geometrías de la nueva capa a la que puede asignar conjunto de datos después. También es posible editar una capa de malla existente. Debido a que la operación de edición requiera una capa de solo tramas se le preguntará si quitar cualquier conjunto de datos asociados primero (compruebe que los tiene disponibles si son necesarios después) o crear una copia (solo geometrías) de la capa.

Nota

QGIS no permite digitalizar aristas en capas de malla. Sólo los vértices y las caras son elementos de malla que se pueden crear. Tampoco todos los formatos de malla soportados pueden ser editados en QGIS (ver permisos).

14.5.1 Visión general de las herramientas de digitalización de malla

Para interactuar con o editar un elemento de base de capa de mala, están disponibles las siguientes herramientas.

Tabla 14.4: Herramientas para digitalización de malla

Label	Propósito	Localización
 Ediciones Actuales	Acceso para guardar, revertir o cancelar cambios en todas las capas o en las seleccionadas simultáneamente	:guilabel:`Barra de herramientas de digitalización
 Palanca para Editar	Conmuta on/off la capa en modo edición	:guilabel:`Barra de herramientas de digitalización
 Guardar Edición	Guarda los cambios hechos en la capa	:guilabel:`Barra de herramientas de digitalización
 Deshacer	Deshace el último cambio(s) - `Ctrl+Z`	:guilabel:`Barra de herramientas de digitalización
 Rehacer	Rehace la última acción(es) deshecha - `Ctrl+Shift+Z`	:guilabel:`Barra de herramientas de digitalización
 Habilitar as herramientas Avan	Conmuta on/off el *Panel de Digitalización Avanzada*	:guilabel:Barra de herramientas *Digitalización Avanzada*
 Reindizar caras y vértices	Vuelve a crear el índice y vuelve a numerar los elementos de la malla para su optimización	:guilabel:Menú *Malla*
 Digitaliza Elementos de Malla	Selecciona/Crea vértices y lados	:guilabel:Barra de herramietas *Digitalización de Malla*
 Selecciona Elementos de la Ma	Selecciona vértices y caras superpuestas de un polígono dibujado	:guilabel:Barra de herramietas *Digitalización de Malla*
 Selecciona Elementos de la Ma	Selecciona vértices y lados usando una expresión	:guilabel:Barra de herramietas *Digitalización de Malla*
 Transforma Coordenadas de lo	Modifica las coordenadas de una selección de vértices	:guilabel:Barra de herramietas *Digitalización de Malla*
 Fuerza por Geometrías Selecci	Dividir caras y restringir el valor a Z usando una geometría lineal	:guilabel:Barra de herramietas *Digitalización de Malla*

14.5.2 Explorando la asignación lógica del valor Z

Cuando una capa de malla se pone en modo edición se abre un artilugio *Valor del vértice Z* en la parte superior derecha del lienzo del mapa. De manera predeterminada su valor corresponde al *Valor Z predeterminado* establecido en la pestaña *Ajustes* ► *Opciones* ► *Digitalización*. Cuando hay vértices seleccionados el artilugio muestra el valor medio de Z de los vértices seleccionados.

Durante la edición el *Valor Z del vértice* es asignado a los nuevos vértices. Tambien es posible establecer un valor a medida: edite el artilugio, pulse `Enter` y usted anulará el valor predeterminado y hará uso del nuevo valor en el proceso de digitalización. Pulse el icono **|limpiar Texto|** en el artilugio para restablecer su valor al predeterminado en Opciones.

Reglas de asignación

This is configured in the drop-down menu of the *Digitize mesh elements* tool as *New vertex Z value* option. Following methods are available:

Tabla 14.5: Z value assignment methods

Method	Description
Project terrain	Z value is always taken or calculated from the *project reference terrain*. • If it is **Flat terrain**, the value corresponds to the *Terrain height* set in the project properties. • If it is a **Mesh** layer, the value is interpolated from the surrounding vertices of the mesh. • If it is a **DEM (raster layer)**, the value is assigned from the raster pixel value. • If there is no terrain under the new vertex, the value falls back to the *Default Z value* set in *digitizing options*.
Z widget	Z value is taken from the *Vertex Z value* widget or the *Advanced Digitizing Panel Z* widget (if it is in Locked state).
Prefer mesh, then Z widget	Interpolates the Z value from the active mesh layer if the vertex is on an edge or face, otherwise falls back to the *Vertex Z value* or *Advanced Digitizing Panel Z* widget (if it is in Locked state).
Prefer mesh, then terrain	Interpolates the Z value from the active mesh layer if the vertex is on the edge or face, otherwise uses the project reference terrain value.

The following detailed logic describes the behavior of the **Prefer mesh, then Z widget** strategy.

Tabla 14.6: Matriz de asignación del valor Z a nuevo vértice

Creación del vértice	Vertices selected?	Fuente del valor asignado	Valor Z asignado
Vértice «libre», no conectado a ninguna cara o arista de una cara.	No	*Valor Z del vértice*	Predeterminado o definido por el usuario
		Panel de Digitalización Avanzada (si el artilugio *z* está en estado Bloqueado)	artilugio *z* si está en estado Bloqueado
	Sí	*Valor Z del vértice*	Average of the selected vertices or user-defined
Vértice sobre una arista	—	Capa de malla	Interpolada desde los vértices de la arista
Vértice sobre una cara	—	Capa de malla	Interpolada desde los vértices de la cara
Vértice ajustado a una entidad vectorial 2D	—	*Valor Z del vértice*	Predeterminado o definido por el usuario
Vértice ajustado a una entidad vectorial 3D	—	Capa vectorial	Vértice
Vértice ajustado a un segmento vectorial 3D	—	Capa vectorial	Interpolado a lo largo del segmento de vector

Nota

El artilugio *Valor Z del vértice* está desactivado si el *Panel de Digitalización Avanzada* está habilitado y no se ha seleccionado elementos de la malla El artilugio *z* de este último gobierna la asignación del valor Z.

Modificando el valor Z de los vértices existentes

Para modificar el valor Z de los vértices, la manera más directa es:

1. Seleccionar uno o más vértices. El artilugio *Valor Z del vértice* mostrará el tamaño medio de la selección.
2. Cambie el valor en el artilugio.
3. Pulse `Enter`. El valor introducido se asigna a los vértices y se convierte en el valor predeterminado de los vértices próximos.

Otra manera de cambiar el valor Z de un vértice es moverlo y encajarlo en una entidad de capa vectorial con la capacidad de valor Z. Si están seleccionados más de un vértice, el valor Z no puede ser cambiado de este modo.

El diálogo *Transformar los vértices de la malla* también proporciona medio para modificar el valor Z de una selección de vértices (junto con sus coordenadas X o Y).

14.5.3 Seleccionar elementos de malla

To select mesh elements, you can use the following tools:

-  Digitize Mesh Elements to select different mesh elements, see more at *Using Digitalizar Elementos de Malla*.
-  Select Mesh Elements by Polygon to select different mesh elements by polygon, see more at *Using Seleccionar Elementos de Malla por Polígono*.
-  Select Mesh Elements by Expression to select different mesh elements by expression, you can choose to *Select by vertices* or *Select by faces*, see more at *Usando Seleccionar Elementos de Malla por Expresión*.
-  Select Isolated Vertices to select all vertices that are not part of any mesh face.
-  Select All Vertices to select all vertices of the mesh layer.

Using *Digitalizar Elementos de Malla*

Activar  :sup:Herramienta *Digitalizar Elementos de Malla*. Pasar el cursor sobre un elemento y resaltarlo permitiéndole seleccionarlo.

- Pulsar en un vértice para seleccionarlo.
- Pulsar en el cuadrado pequeño de una cara o borde y se selecciona. Los vértices conectados también son seleccionados. Por el contrario seleccionando todos los vértices de un borde o una cara también seleccionar ese elemento.
- Arrastre un rectángulo para seleccionar elementos superpuestos (una cara seleccionada viene con todos sus vértices). Pulse la tecla `Alt` si desea seleccionar solo los elementos completamente contenidos.
- Para añadir elementos a una selección, pulse kbd:*Shift* mientras los selecciona.
- Para quitar un elemento de una selección, pulse `Ctrl` y vuelva a seleccionarlo. Una cara fuera de la selección también dejará fuera todos sus vértices.

Using *Seleccionar Elementos de Malla por Polígono*

Activar la  :sup:Herramienta`Seleccionar Elementos de Malla por Polígono` y:

- Dibujar un polígono (pulsación izquierda para añadir vértice, `Backspace` para deshacer el último vértice, `Esc` para abortar el polígono y pulsación derecha para validarlo) sobre las geometrías de la malla. Cualquiera de los vértices y caras parcialmente superpuestos será seleccionado. Pulse la tecla `Alt` mientras dibuja si desea seleccionar solo los elementos completamente contenidos.
- Haga pulsación derecha sobre la geometría de una capa vectorial, selecciónela en la lista que emerge y cualquiera de los vértices o caras parcialmente superpuestas de la capa de malla será seleccionada. Use `Alt` mientras dibuja para seleccionar solo los elementos completamente contenidos.
- Para añadir elementos a una selección, pulse kbd:*Shift* mientras los selecciona.
- Para quitar un elemento de la selección, pulse `Ctrl` mientras dibuja sobre la selección del polígono.

Usando *Seleccionar Elementos de Malla por Expresión*

Otra herramienta para seleccionar elementos de malla es  Seleccionar Elementos de Malla por Expresión. Cuando se pulsa, la herramienta abre el diálogo *selector de expresión* de malla desde el cual usted puede:

1. Seleccionar el método de selección:
   - *Seleccionar por vértices*: aplica la expresión introducida a los vértices y devuelve los que coinciden y sus bordes/caras eventualmente asociadas
   - *Seleccionar por caras*: aplica la expresión introducida a las caras y devuelve las que coincidan y sus bordes/vértices asociados
2. Escribir la expresión de selección. Dependiendo del método seleccionado las funciones disponibles en el *Grupo de mallas* será filtrado en consecuencia.
3. Ejecute la consulta configurando como debe comportarse la selección y pulsando:
   -  *Seleccionar*: reemplaza cualquier selección existente en la capa
   -  *Añadir a la selección actual*
   -  *Quitar de la selección actual*

14.5.4 Modificando elementos de la malla

Añadiendo vértices

Para añadir vértices a una capa de malla:

1. Pulse el botón  Digitalizar elementos de malla
2. A *Vertex Z value* widget is always available on the top right corner of the map canvas during digitizing. It allows setting or overriding the Z coordinate assigned to new vertices, depending on the selected *Z value assignment method*.
3. You can choose other method for assigning Z values to new vertices from the drop-down menu in the  Digitize mesh elements tool.
4. Luego haga doble pulsación:
   - fuera de una cara: añada un «vértice libre», que es un vértice no enlazado a ninguna cara. Este vértice se representa por un punto rojo cuando la capa está en modo edición.

- en el borde de la(s) cara(s) existente(s): añade un vértice en el borde, divide la(s) cara(s) en contacto en triángulos conectados al nuevo vértice.
- dentro de una cara: divide la cara en triángulos cuyos bordes conectan los vértices que rodean al nuevo vértice.

With the *Refine neighboring faces when adding vertices* option enabled in the  Digitize mesh elements button drop-down menu, a check is applied to triangular faces that share at least one vertex with the face the new vertex is added to. If their edges do not satisfy Delaunay triangulation rules, they are flipped accordingly.

Añadiendo caras

Para añadir cara a un capa de malla:

1. Pulse el botón  Digitalizar elementos de malla
2. Aparece un widget *Valor del vértice Z* en la esquina superior derecha del vista del mapa. Establezca este valor en la coordenada Z que le gustaría asignar a los vértices posteriores.
3. Pase el ratón por encima de un vértice y haga clic en el pequeño triángulo que aparece junto a él.
4. Mueva el cursor a la siguiente posición de vértice; puede ajustarse a un vértice existente o hacer clic con el botón izquierdo para añadir uno nuevo.
5. Proceda como arriba para añadir tantos vértices como desee para la cara. Pulse el botón `Retroceso` para deshacer el último vértice.
6. Al mover el ratón, aparece una goma que muestra la forma de la cara. Si aparece en verde, la cara prevista es válida y puede hacer clic con el botón derecho para añadirla a la malla. Si está en rojo, la cara no es válida (por ejemplo, porque se auto-intersecta, se solapa con una cara o vértice existente, crea un agujero, …) y no se puede añadir. Tendrías que deshacer algunos vértices y arreglar la geometría.
7. Pulse `Esc` para abortar la digitalización de la cara.
8. Pulsación derecha para validar la cara.

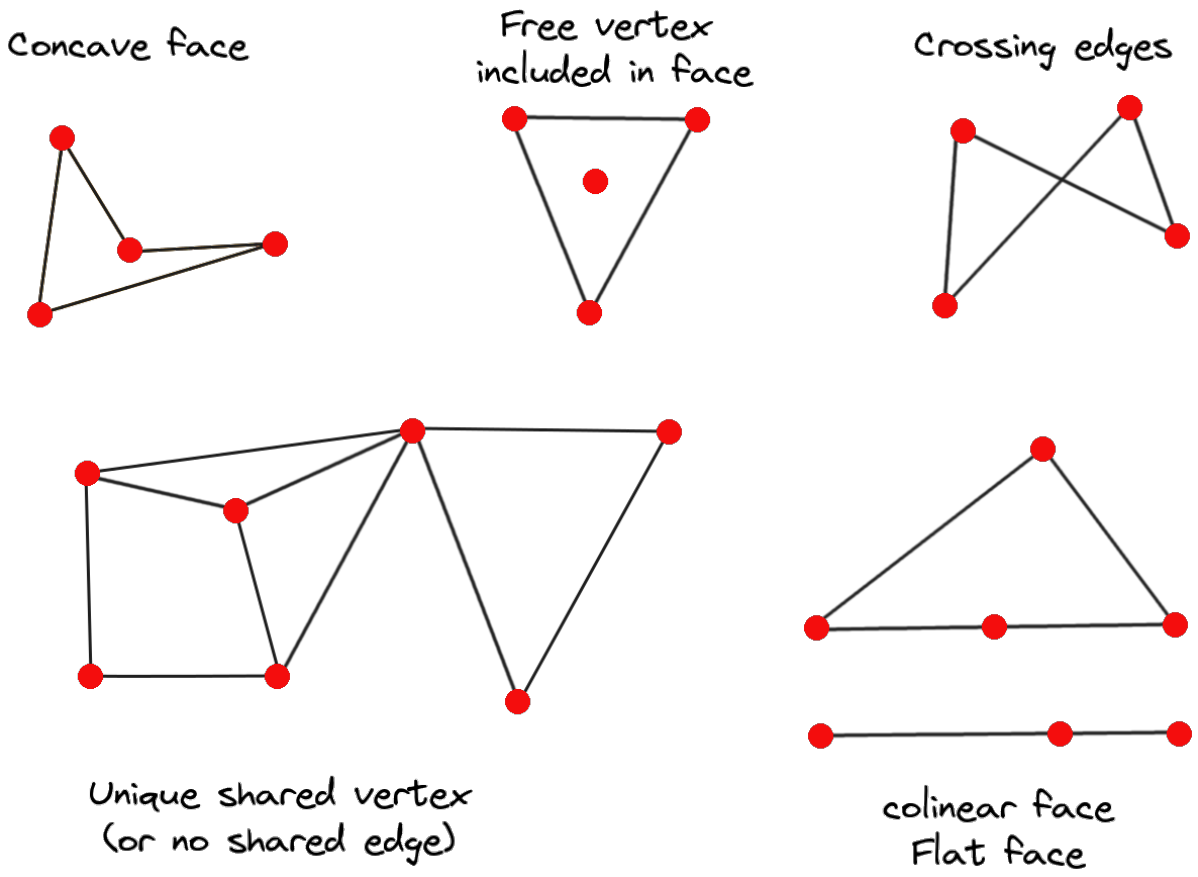


Figura 14.17: Ejemplos de mallas no válidas

Quitando elementos de la malla

1. *Seleccione los elementos objetivo*
2. Habilite la herramienta  Digitalizar elementos de la malla
3. Pulse botón derecho y seleccione:
   - *Quitar los Vértices Seleccionados y Rellenar lo(s) Agujero(s)* o pulse `Ctrl+Del`: quita los vértices y caras enlazadas y rellena lo(s) agujero(s) triangulando desde los vértices vecinos
   - *Quitar los Vértices Seleccionados Sin Rellenar lo(s) Agujero(s)* o pulse `Ctrl+Shift+Del`: quita los vértices y caras enlazadas y no rellena lo(s) agujero(s)
   - *Quitar Cara(s) Seleccionada(s)* o pulsar `Shift+Del`: quita las caras pero mantiene los vértices

Estas opciones son también accesibles desde el menú contextual cuando se pasa sobre un único elemento sin seleccionarlo.

Moviendo elementos de la malla

Para mover vértices y caras de una capa de malla:

1. *Seleccione los elementos objetivo*
2. Habilite la herramienta  Digitalizar elementos de la malla
3. Para empezar a mover el elemento, pulse sobre un vértice o sobre el centroide de una cara/arista
4. Mueva el cursor a la localización objetivo (se admite el ajuste a características vectoriales).
5. Si la nueva ubicación no genera una *malla no válida*, los elementos movidos aparecen en verde. Haga clic de nuevo para liberarlos en esta ubicación. Las caras cuyos vértices están todos seleccionados se trasladan, sus vecinos se remodelan en consecuencia.

Transformando vértices de la malla

La herramienta  Transformar Coordenadas de los Vértices da una manera más avanzada de mover los vértices editando sus coordenadas X, Y y/o Z gracias a expresiones.

1. Seleccionar los vértices donde desea editar las coordenadas
2. Pulse  Transformas las Coordenadas de los Vértices. Se abre un diálogo con una mención del número de vértices seleccionados. Todavía puede añadir o quitar vértices de la selección.
3. Dependiendo de las las propiedades que desee modificar tiene que comprobar *Coordenada X*, *Coordenada Y* y/o *Valor Z*.
4. Después introduzca la posición objetivo en la caja, bien como un valor numérico o una expresión (usando el  Diálogo expresión)
5. With the  Import Coordinates of the Selected Vertex pressed, the X, Y and Z boxes are automatically filled with its coordinates whenever a single vertex is selected. You can adjust Z value by using  *Get the Z value from the project terrain* button and in that case the Z box is deactivated. A convenient and quick way to adjust vertices individually.
6. Press *Vista Previa de la Transformación* para simular la nueva localización de los vértices y previsualizar la malla con la transformación.
   - Si la vista previa es verde, la malla transformada es válida y usted puede aplicar la transformación.
   - Si la vista previa está en rojo, la malla transformada no es válida y usted no puede aplicar la transformación hasta que se corrija.
7. Press *Aplicar la Transformación* para modificar las coordenadas seleccionadas para el conjunto de los vértices.

Reformando geometría de malla

El menú contextual

1. Habilita la  Digitalización de elementos de la malla
2. Seleccionar elemento(s) de malla o no
3. Al pasar sobre un elemento de malla, se destaca.
4. Pulsación derecha y usted puede:
   - *quitar elemento(s)*

- *Dividir Cara(s) Seleccionada(s)* (*Dividir Cara Actual*): divide la cara sobre la que se está desplazando o cada cara de malla cuádruple en dos triángulos
- *Triangulación de Delaunay con vértices Seleccionados*: construye caras triangulares usando los vértices libres seleccionados.
- *Refinar Cara(s) Seleccionada(s)* (*Refinar Cara Actual*): divide la cara en cuatro caras, en base a los vértices añadidos a la mitad de cada arista (un triángulo resulta en triángulos un cuadrilátero en cuadriláteros). También triangula las caras adyacentes conectadas a los nuevos vértices.

Los marcadores de borde

Cuando la  Digitalización de elementos de malla está activa y usted pasa sobre un borde, el borde se resalta y es posible interactuar con él. Dependiendo del contexto pueden estar disponibles los siguientes marcadores:

- un **cuadrado**, en el centro del borde: pulse en él para seleccionar los vértices de los extremos.
- una **cruz** si las dos caras de cada lado se pueden fusionar: pulse sobre ella para borrar el borde y fusionar las caras.
- un **círculo** si la arista está entre dos triángulos: Pulse en él para dar la vuelta a la arista, esto es, conectarla a los otros vértices «libres» de las caras

La herramienta *Fuerza por Geometrías Seleccionadas*

The  :sup:La herramienta *Fuerza por las Geometrías Seleccionadas* proporciona maneras avanzadas de aplicar líneas de ruptura usando geometrías de líneas. Una línea de ruptura forzará a la malla a tener aristas a lo largo de la línea. Tenga en cuenta que la linea de ruptura no se considerará persistente una vez efectuada la operación; las artistas resultante no actuarán como restricciones y podrán ser modificadas como cualquier otro borde. Esto se puede usar, por ejemplo, para modificar localmente una capa de malla con líneas precisas, como riberas de ríos o borde de terraplenes de carreteras.

1. Habilitar la herramienta  Fuerza por las Geometrías Seleccionadas
2. Indicar la geometría a ser usada como «línea de fuerza»; puede ser:
   - seleccionado de una entidad de línea o polígono en el lienzo del mapa: botón derecho sobre la característica vectorial y seleccionarlo de la la lista en el menú contextual.
   - una linea virtual dibujada sobre la trama de la malla: botón izquierdo para añadir vértices, botón derecho para valida. El valor Z de los vértices se establece por medio del artilugio *Valor Z del vértice* o el artilugio *z* si está activado el *Panel de Digitalización Avanzada*. Si la línea se ajusta a un vértice de malla o al vértice o segmento de una entidad vectorial 3D, el nuevo vértice toma el valor Z del elemento ajustado.

Las caras de malla que se superponen a la geometría de la línea o al límite del polígono se verán afectadas de una manera que depende de las opciones que usted haya establecido en la herramienta  Fuerza por Geometrías Seleccionadas del menú desplegado:

-  *Agregar un nuevo vértice en las aristas que se cruzan*: con esta opción se añade un nuevo vértice cada vez que línea de fuerza se cruza con una arista. Esta opción lleva a la división a lo largo de la línea a la que se enfrenta cada uno.

  Sin esta opción, las caras encontradas son quitadas y reemplazadas por las caras que viene de una triangulación con los vértices existentes más los vértices de las líneas forzadas (también se añaden nuevos vértices en el borde del límite donde se cruzan las líneas de fuerza).

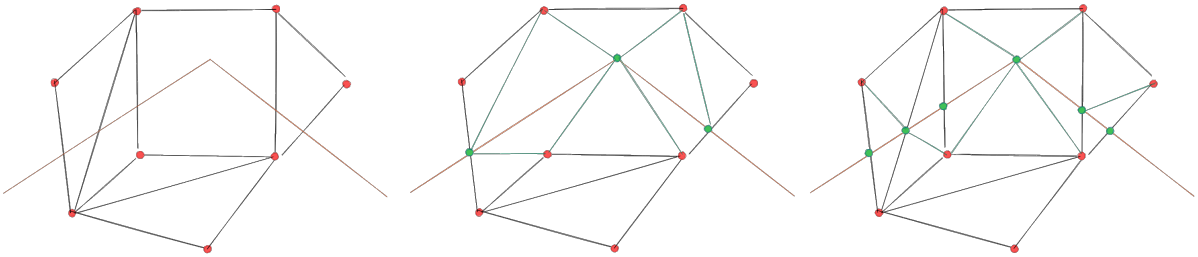


Figura 14.18: Fuerza la Malla usando una geometría de línea - Resultados sin (medio) y con (derecha) nuevo vértice en la intersección de los bordes

- *Interpolar el valor Z desde*: establece cómo se calcula el valor Z de los nuevos vértices. Puede ser desde:
  - la *Malla* misma: el valor Z de los nuevos vértices se interpola de los vértices de la cara dentro de la que cae
  - o *Línea de fuerza*: sí la línea se define por una característica vectorial 3D o una nueva línea dibujada el valor Z se deriva de su geometría. En el caso de una característica lineal 2D, el valor de los nuevos vértices Z es el *Valor del vértice Z*.
- *Tolerancia*: cuando un vértice de malla está más cerca a la línea que el valor de tolerancia, no crea un nuevo vértice en la línea y usa un vértice existente en su lugar. El valor se puede establecer en *Metros a Escala* o en *Unidades del Mapa* (más detalles en *Selector de Unidad*).

14.5.5 Volviendo a indexar mallas

Durante la edición, con el objetivo de permitir un rápido hacer/deshacer de la operaciones, QGIS mantiene sitios vacíos para los elementos borrados, que puede llevar a un crecimiento del uso de memoria y una estructuración de malla ineficiente. La herramienta *Malla* ► *Volver a Indexar Caras y Vértices* está diseñada para quitar estos agujeros y re-numerar los índices de caras y vértices de modo que sean continuos y de alguna manera razonablemente ordenados. Estos optimiza la relación entre caras y véftices e incrementa la eficiencia de cálculo.

Nota

La herramienta *Volver a Indexar Caras y Vértices* tool sguarda la capa y borra las pilas de deshacer/rehacer, deshabilitando cualquier reversión.

14.6 Calculadora de malla

La herramienta *Calculadora de Malla* del menú superior *Malla* le permite realizar cálculos aritméticos y lógicos en grupos de conjuntos de datos existentes para generar un nuevo grupo de conjuntos de datos (ver Figura 14.19).

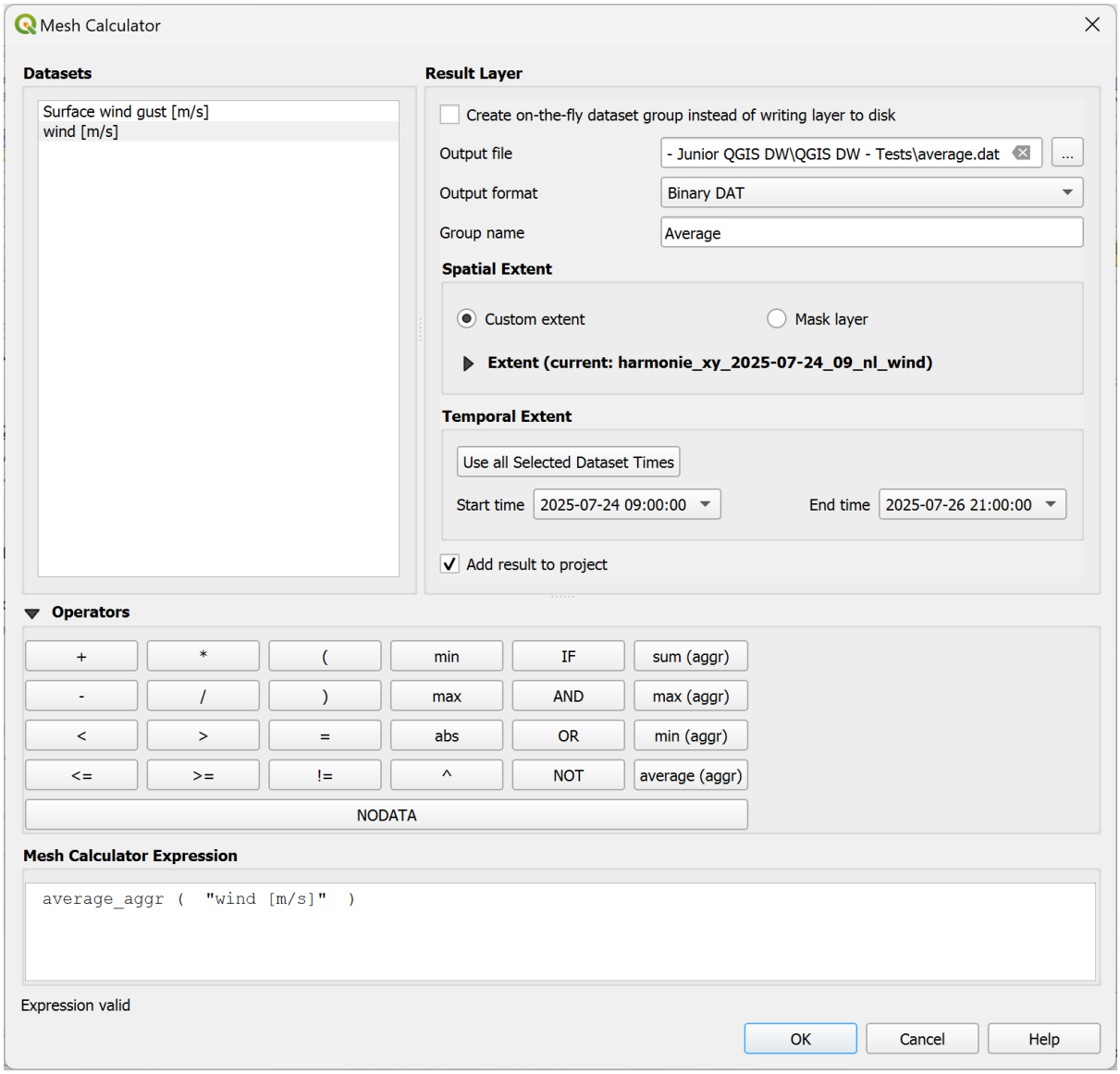


Figura 14.19: Calculadora de malla

La lista *Datasets* contiene todos los grupos de conjuntos de datos en la capa de malla activa. Para usar un grupo de conjuntos de datos en una expresión, haga doble clic en su nombre en la lista y se agregará al campo *Expresión de calculadora de malla*. Luego puede usar los operadores para construir expresiones de cálculo, o simplemente puede escribirlas en el cuadro.

La *Capa Resultado* te ayuda a configurar las propiedades de la capa de salida:

- :guilabel:`Crea un grupo de conjuntos de datos sobre la marcha en lugar de escribir una capa en el disco:
  - Si no está marcada, la salida se almacena en el disco como un nuevo archivo sin formato. Se requiere una ruta *Archivo de salida* y un *Formato de salida*.
  - Si está marcado, se agregará un nuevo grupo de conjuntos de datos a la capa de malla. Los valores del grupo de conjuntos de datos no se almacenan en la memoria, pero cada conjunto de datos se calcula

cuando es necesario con la fórmula ingresada en la calculadora de malla. Ese grupo de conjuntos de datos virtuales se guarda con el proyecto y, si es necesario, se puede eliminar o hacer persistente en el archivo desde la pestaña de propiedades de la capa *Fuente*.

En cualquier caso, debe proporcionar un *Nombre de grupo* para el grupo de conjuntos de datos de salida.

- La *Extensión espacial* a considerar para el cálculo puede ser:
  - defined as a *Custom extent* using the *extent selector* widget.
  - definido por una capa de polígono (*Capa de máscara*) del proyecto: la geometría de las características del polígono se utiliza para recortar los conjuntos de datos de la capa de malla
- La *Extensión temporal* a tener en cuenta para los conjuntos de datos se puede configurar con las opciones *Hora de inicio* y *Hora de finalización*, seleccionadas de los grupos de conjuntos de datos existentes. También se pueden rellenar utilizando el botón *Usar todos los tiempos del dataset seleccionado* para tomar todo el rango.
- With the  *Add result to project* checkbox, the result layer will automatically be added to the legend area and can be visualized. Checked by default.

La sección *Operadores* contiene todos los operadores disponibles. Para añadir un operador al cuadro de expresión de la calculadora de malla, haga clic en el botón correspondiente. Los cálculos matemáticos (+, −, *, …) y las funciones estadísticas (`min`, `max`, `sum (aggr)`, `average (aggr)`, …) están disponibles. Las expresiones condicionales (=, !=, <, >=, `IF`, `AND`, `NOT`, …) devuelven 0 para falso y 1 para verdadero, por lo que pueden utilizarse con otros operadores y funciones. El valor `NODATA` también se puede utilizar en las expresiones.

El widget *Expresión de calculadora de malla* muestra y le permite editar la expresión para ejecutar.

CAPÍTULO 15

Trabajando con Teselas Vectoriales

15.1 ¿Qué son las teselas vectoriales?

Las teselas vectoriales son paquetes de datos geográficos, empaquetados en «mosaicos» predefinidos de forma aproximadamente cuadrada para su transferencia a través de la web. Combinan teselas de mapas ráster pre-renderizados y teselas de mapas vectoriales. El servidor de teselas vectoriales devuelve datos de mapas vectoriales, que se han recortado a los límites de cada tesela, en lugar de una imagen de mapa renderizada previamente. Las teselas recortadas representan los niveles de zoom del servicio de teselas vectoriales, derivados de un enfoque piramidal. Con esta estructura, la transferencia de datos se reduce en comparación con los mapas vectoriales sin teselas. Solo es necesario transferir los datos dentro de la vista de mapa actual y en el nivel de zoom actual. Además, en comparación con un mapa ráster en mosaico, la transferencia de datos también se reduce considerablemente, ya que los datos vectoriales suelen ser mucho más pequeños que un mapa de bits renderizado. Las teselas vectoriales no tienen ninguna información de estilo asignada, por lo que QGIS necesita aplicar un estilo cartográfico para mostrar los datos.

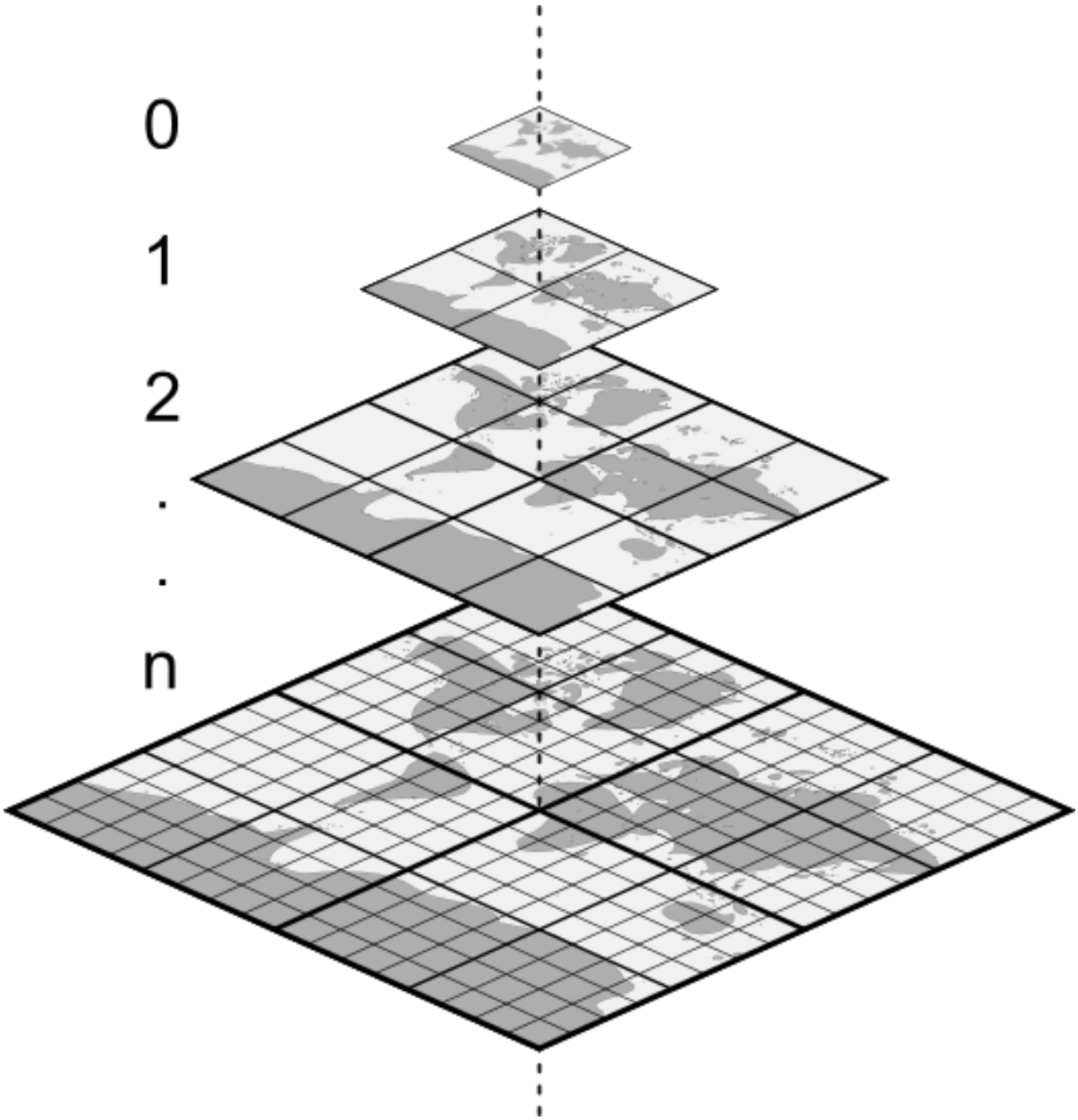


Figura 15.1: Estructura piramidal de teselas vectoriales con niveles de zoom

15.2 Formatos Soportados

Hay soporte para teselas vectoriales a través de:

- fuentes (HTTP/S) remotas - con plantilla XYZ - `type=xyz&url=http://example.com/{z}/{x}/{y}.pbf`
- archivos locales - con plantilla XYZ - p.ej. `type=xyz&url=file:///path/to/tiles/{z}/{x}/{y}.pbf`
- base de datos local MBTiles - p.ej. `type=mbtiles&url=file:///path/to/file.mbtiles`

Para cargar un conjunto de datos de teselas vectoriales en QGIS, utiliceladdVectorTileLayer| *Tesela vectorial* en el cuadro de diálogo *Administrador de fuente de datos*. Lea *Usando de Servicios de Teselas Vectoriales* para más detalles.

15.3 Propiedades del conjunto de datos de teselas vectoriales

El cuadro de diálogo Propiedades de las capas de mosaico vectorial *Propiedades de capa* proporciona las siguientes secciones:



15.3.1 Propiedades de información

La pestaña *Información* es de solo lectura y representa un lugar interesante para obtener rápidamente información resumida y metadatos en la capa actual. Los datos proporcionados son:

- según el proveedor de la capa: nombre, URI, tipo de fuente y ruta, número de niveles de zoom
- propiedades personalizadas, utilizadas para almacenar en el proyecto activo información adicional sobre la capa. Se pueden crear y gestionar más propiedades utilizando PyQGIS, concretamente a través del método setCustomProperty().
- the Coordinate Reference System: name, units, method, accuracy, reference (i.e., whether it is static or dynamic)
- seleccionado de los *filled metadata*: acceso, extensiones, enlaces, contactos, historial…

15.3.2 Propiedades de fuente

La pestaña *Fuente* muestra información básica sobre el mosaico vectorial seleccionado, incluyendo:

- el *Nombre de la capa* a mostrar en el *Panel de capas*;
- el *Sistema de Referencia de Coordenadas*: Muestra el *Sistema de referencia de coordenadas (SRC)* de la capa. Puede cambiar el SRC de la capa, seleccionando uno utilizado recientemente en la lista desplegable o haciendo clic en el botón Seleccionar SRC (véase *Selector del Sistema de Coordenadas de Referencia*). Utilice este proceso sólo si el SRC de la capa es incorrecto o no está especificado.
- For file based vector tiles (`.vtpk` or `.mbtiles`), a *Layer source* group indicates the path to the source of the dataset and allows for replacing the loaded layer. Edit the path shown in the text box or press … Browse to select another file on the disk.

[1] Also available in the *Layer styling panel*

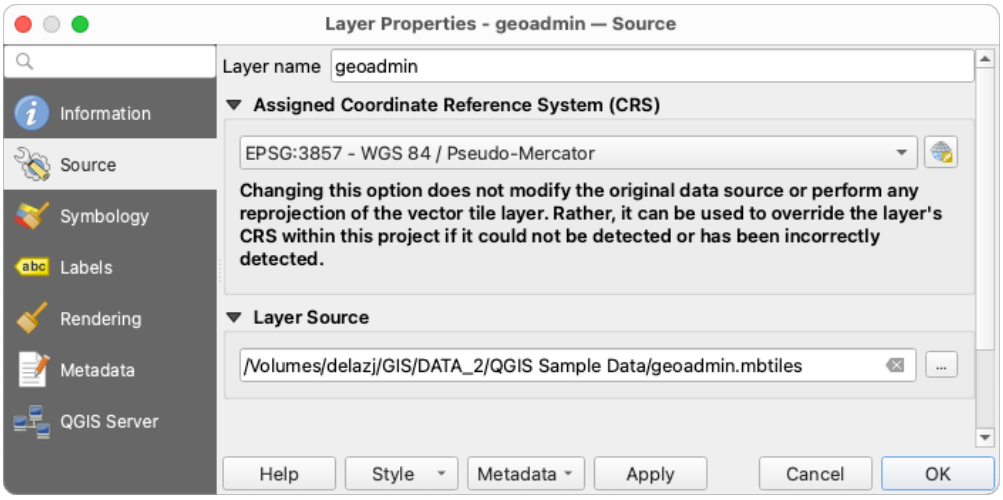


Figura 15.2: Propiedades de las teselas vectoriales - Diálogo Fuente

15.3.3 Propiedades de simbología y etiquetas

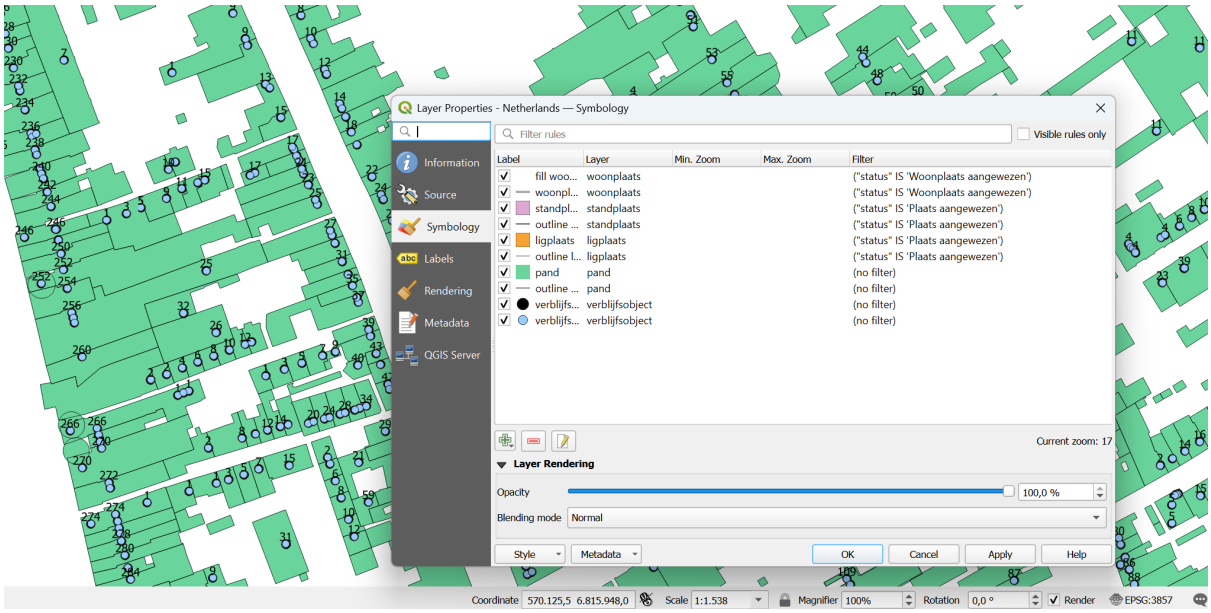


Figura 15.3: Simbología de capa de teselas vectoriales

Establecer reglas

Como las teselas vectoriales constan de geometrías de puntos, líneas y polígonos, se dispone de los símbolos correspondientes. Para aplicar un estilo cartográfico (con simbología y/o etiquetas), puede:

- Utilice una *URL de estilo* al crear la *Conexión de teselas vectoriales*. La simbología se mostrará inmediatamente en la pestaña  *Symbología* después de cargar la capa en QGIS.
- O construya su propia simbología y etiquetado en las pestañas correspondientes de las propiedades de las capas. Por defecto, QGIS asigna un símbolo idéntico a las características en función de su tipo de geometría.

En ambos casos, el establecimiento de un estilo para una tesela vectorial se basa en un conjunto de *reglas* aplicada a los objetos espaciales, indicando:

- un *símbolo* o *etiqueta* vectorial
- a *Etiqueta*, un título para la identificación exhaustiva de la regla
- el nombre de una *Capa* concreta a la que debe aplicarse la regla, si no se aplica a `(todas las capas)`
- una *Zoom Mín. Zoom* y un *Zoom Máx. Zoom*, para el rango de visualización. La simbología y el etiquetado pueden depender del nivel de zoom.
- a *Filtro*, una expresión QGIS para identificar los objetos espaciales en los que se aplicará el estilo

Cada regla se añade pulsando el botón Añadir regla y seleccionando el tipo de símbolos (*Marcador*, *Línea*, *Relleno*) correspondiente al tipo de geometría del objeto espacial. También puede Eliminar las reglas seleccionadas o Editar la regla actual.

En la parte inferior se muestra *Zoom actual*. Marque la opción *Sólo reglas visibles* en la parte superior del diálogo para filtrar la lista de reglas a sólo aquellas que son visibles en el nivel de zoom actual. Esto facilita el trabajo con estilos vectoriales complejos y la localización de reglas problemáticas. El cuadro de texto *Reglas de filtrado* también le ayuda a encontrar fácilmente una regla, buscando en los campos *Etiqueta*, *Capa* y *Filtro*.

En Figura 15.3 configuramos el estilo para la capa `landuse` de OpenStreetMap. Para una mejor visibilidad la mayoría de las reglas están deseleccionadas.

Reproducción de capas

Desde la pestaña *Simbología*, también puede configurar algunas opciones que actúan invariablemente sobre todas las características de la capa:

- *Opacidad*: Con esta herramienta puede hacer visible la capa subyacente en el lienzo del mapa. Utilice el control deslizante para adaptar la visibilidad de la capa vectorial a sus necesidades. También puede definir con precisión el porcentaje de visibilidad en el menú situado junto al deslizador.
- *Modo de mezcla*: Con estas herramientas puede conseguir efectos de renderizado especiales que antes sólo conocía de los programas gráficos. Los píxeles de las capas superpuestas y subpuestas se mezclan mediante los ajustes descritos en *Modos de Mezcla*.

Estilos

Disponible en la parte inferior de la mayoría de las pestañas, el menú *Estilos* ► proporciona accesos directos para guardar, cargar, crear, cambiar estilos para aplicar a las teselas vectoriales. Las teselas vectoriales pueden tener su estilo guardado desde QGIS como archivos *QML* y pueden ser importados como:

- *QML* archivos (*QML: el formato de archivo de estilo QGIS*)
- :guilabel:Archivos de configuración de estilo *MapBox GL Json*

Más detalles en *Guardar y compartir Propiedades*.

15.3.4 Propiedades de representación

En *Visibilidad dependiente de la escala*, puede establecer las escalas *Máxima (inclusive)* y *Mínima (exclusiva)*, definiendo un rango de escalas en el que las características serán visibles. Fuera de este rango, se ocultan. El botón Establecer en la escala actual del lienzo le ayuda a utilizar la escala actual del lienzo del mapa como límite del rango de visibilidad. Véase *Selector de escala de visibilidad* para más información.

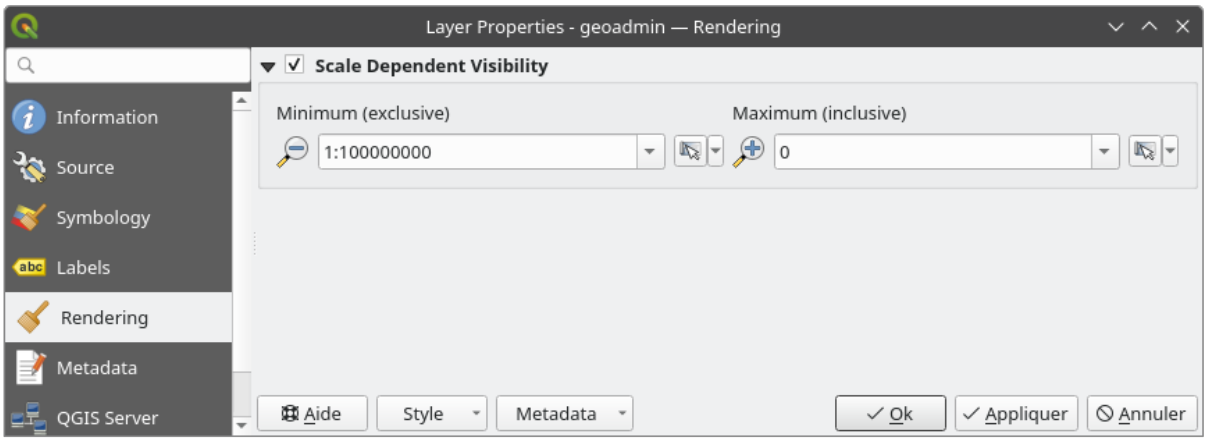


Figura 15.4: Propiedades de las teselas vectoriales - Diálogo de renderizado

15.3.5 Propiedades de metadatos

El *Metadata* le brinda opciones para crear y editar un informe de metadatos en su capa. Consulte *Metadatos* para obtener más información.

15.4 QGIS Server Properties

The *QGIS Server* tab helps you configure settings of the data when published by QGIS Server. The configuration concerns:

- *Description*: provides information to describe the data, such as *Short name*, *Title*, *Abstract*, a *Keyword list*, and a *Data URL* whose *Format* can be in `text/html`, `text/plain` or `application/pdf`.
- *Attribution*: a *Title* and *URL* to identify who provides the data
- *Metadata URL*: a list of *URL* for the metadata that can be of `FGDC` or `TC211` *Type*, and in `text/plain` or `text/xml` *Format*
- *Legend URL*: a *URL* for the legend, in either `image/png` or `image/jpeg` *Format*

CAPÍTULO 16

Trabajar con Puntos Nubes

16.1 Introducción a las Nubes de Puntos

**¿Qué es una Nube de Puntos?

Una nube de puntos es una imagen tridimensional de un espacio formada por muchos individuos de puntos de datos (hasta miles de millones, incluso billones). Cada uno de los puntos tiene una coordenada x, y y z. Dependiendo del método de captura, las nubes de puntos suelen tener también atributos adicionales procedentes de la captura, como valores de color o intensidad. Estos atributos pueden utilizarse, por ejemplo, para mostrar nubes de puntos en diferentes colores. En QGIS, una nube de puntos puede utilizarse para generar una imagen tridimensional del paisaje (o de otro espacio).

Formatos admitidos

QGIS es compatible con los formatos de datos Entwine Point Tile (EPT) y LAS/LAZ. Para trabajar con nubes de puntos, QGIS siempre guarda los datos en EPT. EPT es un formato de almacenamiento que consta de varios archivos guardados en una carpeta común. Para permitir un acceso rápido a los datos, EPT utiliza indexación. Para obtener más información sobre el formato EPT, consulte la página web de Entwine <https://entwine.io/en/latest/entwine-point-tile.html>_.

Si los datos están en formato LAS o LAZ, QGIS los convertirá a EPT cuando se carguen por primera vez. Dependiendo del tamaño del archivo, esto puede llevar algún tiempo. En este proceso, se crea una subcarpeta en la carpeta en la que se encuentra el archivo LAS/LAZ según el esquema **:archivo:`ept_`** + **:archivo:`nombre_archivo_LAS/LAZ`**. Si dicha subcarpeta ya existe, QGIS carga el EPT inmediatamente (lo que conlleva una reducción del tiempo de carga).

Merece la pena saberlo

En QGIS no es posible (todavía) editar nubes de puntos. Si desea manipular su nube de puntos, puede utilizar CloudCompare, una herramienta de procesamiento de nubes de puntos de código abierto. También la Point Data Abstraction Library (PDAL - similar a GDAL) le ofrece opciones para editar nubes de puntos (PDAL es sólo línea de comandos).

Debido al gran número de datos de puntos no es posible mostrar una tabla de atributos de las nubes de puntos en GIS. Sin embargo, **|identificar|** *Identify tool* soporta nubes de puntos, así que usted puede visualizar todos los atributos incluso de un único dato de punto.

Si desea crear una nueva capa, en el mismo formato u otro formato compatible, a partir de una capa de nube de puntos existente, consulte *Creación de nuevas capas a partir de una capa existente.*.

16.2 Propiedades de las nubes de puntos

El cuadro de diálogo *Propiedades de capa* de una capa de nube de puntos ofrece ajustes generales para la capa y su representación. También proporciona información sobre la capa.

Para acceder al diálogo:guilabel:*Propiedades de la capa*:

- En el panel *Capas*, haga doble clic en la capa o haga clic con el botón derecho y seleccione *Propiedades…* en el menú contextual;
- Vaya al menú *Capa* ► *Propiedades de capa…* cuando la capa este seleccionada.

El cuadro de diálogo de la nube de puntos *Propiedades de capa* ofrece las siguientes secciones:



Nota

La mayoría de las propiedades de una capa de nube de puntos pueden guardarse o cargarse desde un archivo `.qml` utilizando el menú *Style* situado en la parte inferior del diálogo de propiedades. Más detalles en *Guardar y compartir Propiedades*.

16.2.1 Propiedades de información

La pestaña *Information* es de solo lectura y representa un lugar interesante para obtener rápidamente información resumida y metadatos para la capa actual. La información proporcionada es:

- Datos generales como el nombre en el proyecto, la ruta de origen, la última vez que se guardó y el tamaño, el proveedor utilizado
- propiedades personalizadas, utilizadas para almacenar en el proyecto activo información adicional sobre la capa. Las propiedades personalizadas por defecto pueden incluir *notas de capa*. Se pueden crear y gestionar más propiedades utilizando PyQGIS, concretamente a través del método setCustomProperty().
- En función del proveedor de la capa: extensión y número de puntos
- El sistema de referencia de coordenadas: nombre, unidades, método, precisión, referencia (es decir, si es estático o dinámico).
- Metadatos suministrados por el proveedor: fecha de creación, versión, formato de los datos, escala X/Y/Z, …
- Escogidos de la pestaña *Metadato* (donde pueden ser editados): acceso, extensión, enlaces, contactos, historia…

[1] Also available in the *Layer styling panel*

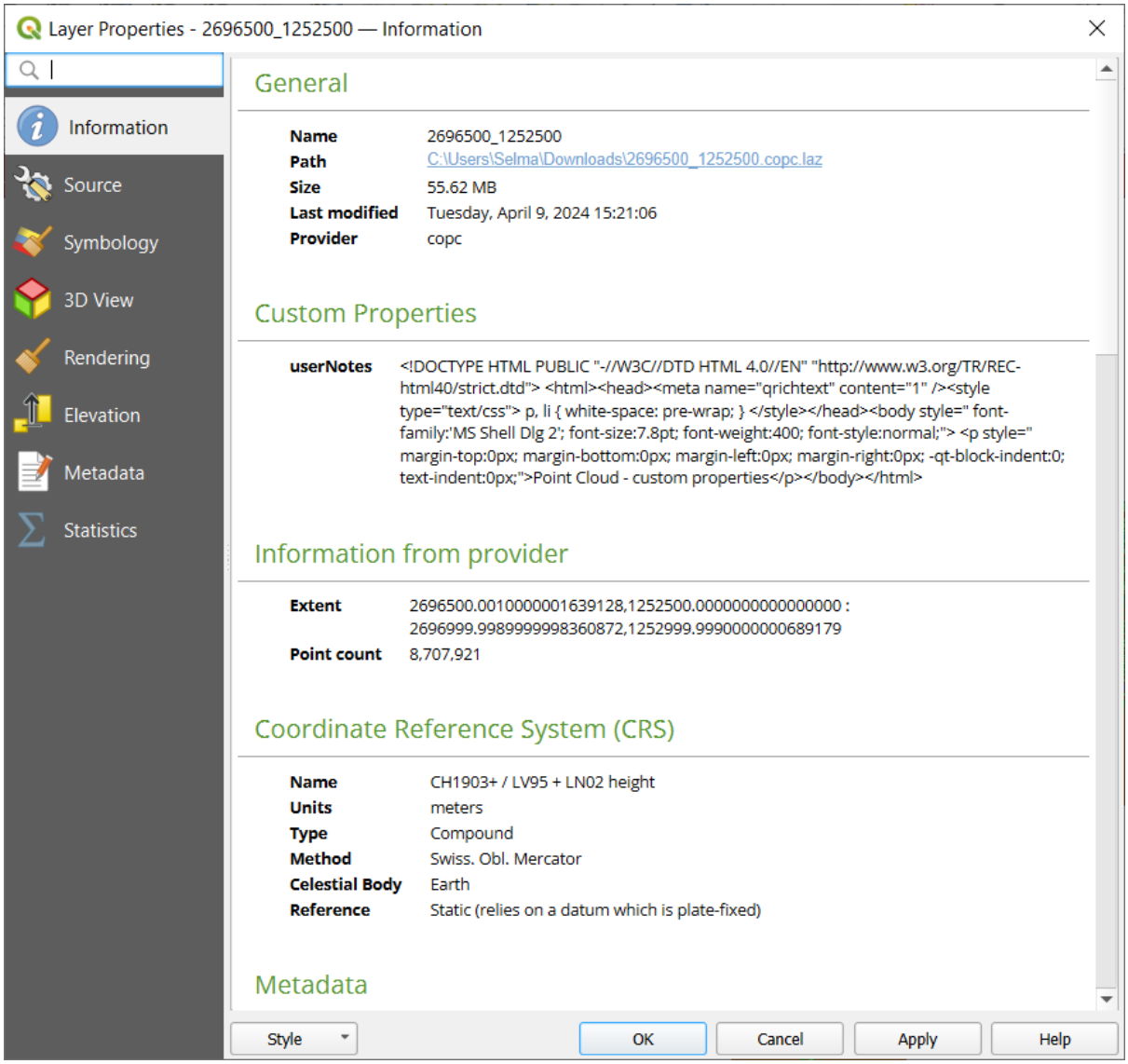


Figura 16.1: Pestaña de información de la nube de puntos

16.2.2 Propiedades de fuente

En la pestaña *Fuente* puede ver y editar información básica sobre la capa de la nube de puntos:

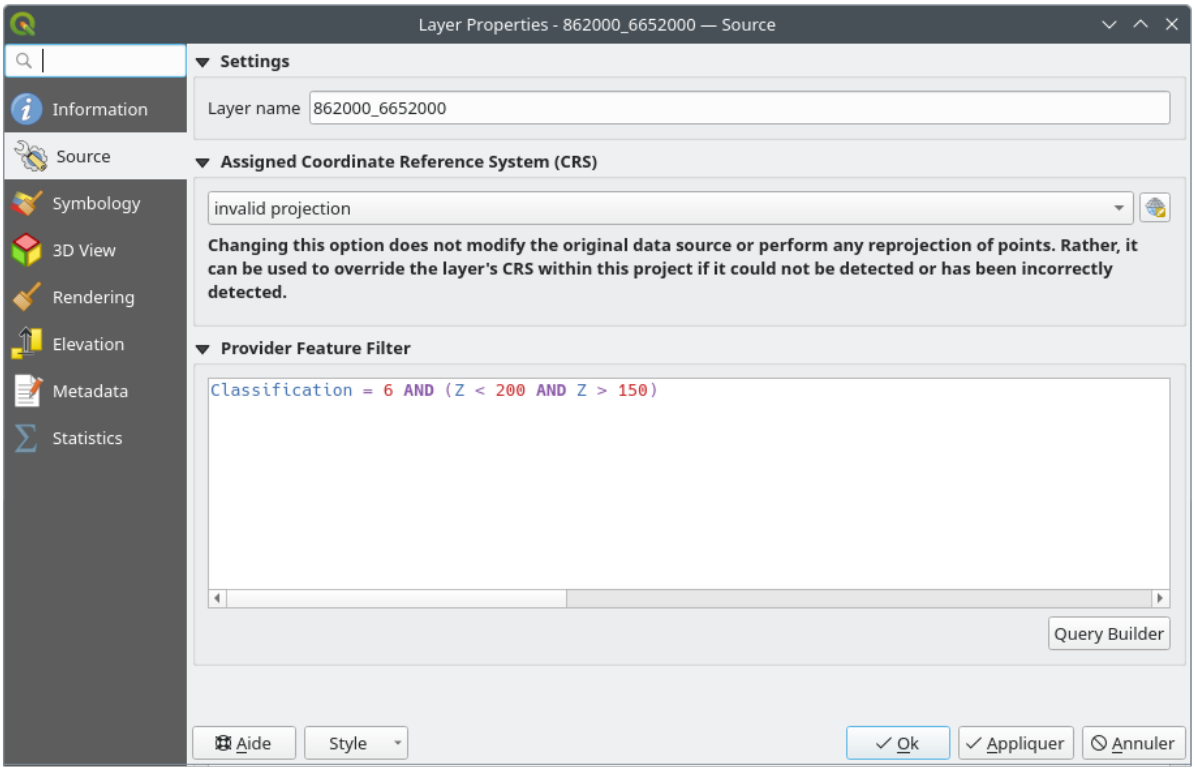


Figura 16.2: Pestaña fuente de nube de puntos

- *Ajustes*: Establezca un nombre de capa distinto del nombre de archivo de la capa que se utilizará para identificar la capa en el proyecto (en el Panel de capas, con expresiones, en la leyenda de diseño de impresión,…).
- *Sistema de Referencia de Coordenadas Asignado (CRS)*: Aquí puede cambiar el el *Sistema de Referencia de Coordenadas* asignado de la capa, seleccionando uno, usado recientemente, de la lista desplegable o pulsando en el botón Seleccionar Proyección CRS (vea *Selector del Sistema de Coordenadas de Referencia*). Use este proceso sólo si el CRS aplicado a la capa está equivocado o si no se ha aplicado ninguno.
- *Filtro de objeto espacial del proveedor*: permite restringir los datos a utilizar en el proyecto actual a un subconjunto, utilizando funciones y expresiones a nivel del proveedor de datos PDAL. Pulse el botón *Constructor de consultas* en la parte inferior para empezar a configurar el filtro.

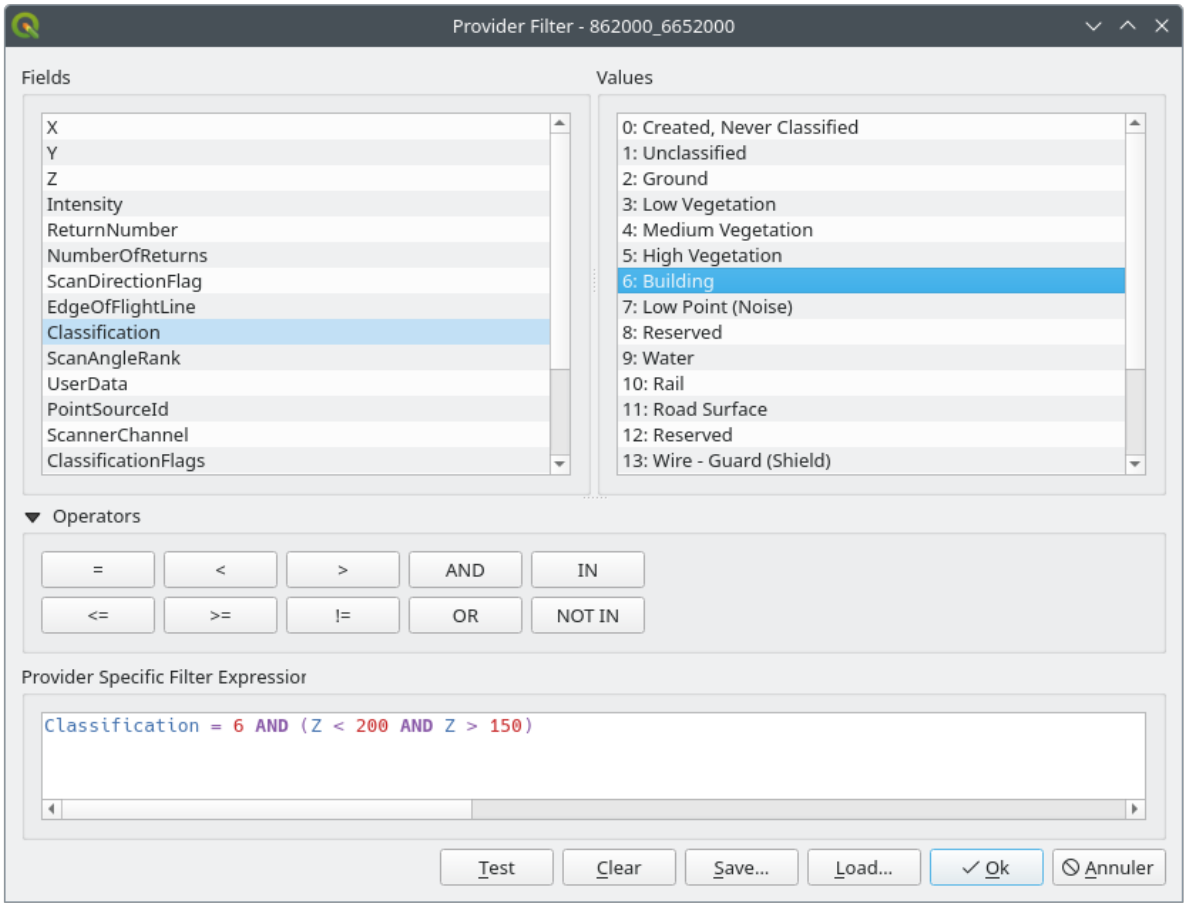


Figura 16.3: Filtrado de la capa de nubes de puntos a edificios a una altitud específica

En la parte inferior del cuadro de diálogo, puede escribir una *Expresión de filtro específica del proveedor*. Dicha expresión puede construirse gracias a:

- *Campos*: la lista contiene todos los atributos de la capa de la nube de puntos. Para añadir un atributo a la expresión, haga doble clic en su nombre o simplemente escríbalo en el cuadro de texto.
- *Valores*: enumera los valores o estadísticas del atributo actualmente seleccionado, como un par `clave:valor`. Para añadir un valor al campo de expresión, haga doble clic en su nombre en la lista: dependiendo del tipo de atributo, se añadirá a la expresión la clave o el valor. O escriba el valor en el cuadro de texto de la expresión.
- *Operadores*: esta barra de herramientas contiene todos los operadores utilizables. Para añadir un operador al campo de expresión, haga clic en el botón correspondiente. Están disponibles los operadores relacionales (= , > , …) y los operadores lógicos (`AND`, `OR`, …).

El botón *Probar* le ayuda a comprobar la sintaxis de su consulta. Utilice el botón *Borrar* para borrar la consulta y revertir la capa a su estado original (es decir, cargar completamente todos los puntos de la capa). Es posible *Guardar…* la consulta como un archivo `.QQF`, o *Cargar…* la consulta desde un archivo en el diálogo.

Cuando se aplica un filtro, QGIS trata el subconjunto resultante como si fuera la capa entera. Por ejemplo si aplicó el :ref:` filtro arriba <figure_point_cloud_querybuilder>` para filtrar edificios, no puede, por ejemplo, visualizar, consultar, guardar o editar puntos que sean de clasificación de vegetación porque no forman parte del subconjunto.

Truco

Las capas filtradas se indican en el Panel de capas

En el panel *Capas*, una capa filtrada aparece con un icono  Filter al lado que indica la consulta utilizada cuando el ratón pasa por encima del icono. Haga doble clic en el icono para abrir el cuadro de diálogo *Query Builder* para su edición. También puede hacerse a través del menú *Capa* ► *Filtro…*.

16.2.3 Propiedades de simbología

En la pestaña  *Simbología* se realizan los ajustes para el renderizado de la nube de puntos. En la parte superior se encuentran los ajustes de los diferentes renderizadores de objetos espaciales. En la parte inferior, hay secciones con las que se pueden hacer ajustes generales para toda la capa y que se aplican sobre los renderizadores de objetos espaciales.

Tipos de renderizado de objetos espaciales

Existen diferentes opciones para renderizar nubes de puntos que pueden seleccionarse utilizando el menú desplegable situado en la parte superior de la pestaña *Simbología* (véase Figura 16.4):

-  *Sólo extensión*: Sólo se muestra un cuadro delimitador de la extensión de los datos; conveniente para tener una visión general de la extensión de los datos. Como de costumbre, el *Símbolo widget* le ayuda a configurar cualquier propiedad (color, trazo, opacidad, subcapas, …) que desee para la caja.
-  *Atributo por rampa*: Los datos se dibujan sobre un gradiente de color. Véase punto_nube_rampa.
-  *RGB*: Dibuja los datos utilizando los valores de color rojo, verde y azul. Véase *Renderizador RGB*.
-  *Clasificación*: Los datos se dibujan utilizando diferentes colores para las distintas clases. Véase clasificación_de_nube_punto.

Cuando se carga una nube de puntos, QGIS sigue una lógica para seleccionar el mejor renderizador:

- si el conjunto de datos contiene información de color (atributos rojo, verde, azul), se utilizará el renderizador RGB
- si el conjunto de datos contiene un atributo `Clasificación`, se utilizará el renderizador clasificado
- de lo contrario, volverá a la representación basada en el atributo Z

Sí no conoce los atributos de la nube de puntos la pestaña  *Estadísticas* :ref:` <point_clouds_statistics>` proporciona una buena visión general de que atributos están contenidos en la nube de puntos y en que rango se sitúan los valores.

Para cada renderizador, puede mejorar la visualización de los datos ajustando el *tamaño del símbolo de punto* o habilitar *triangulación de superficie*.

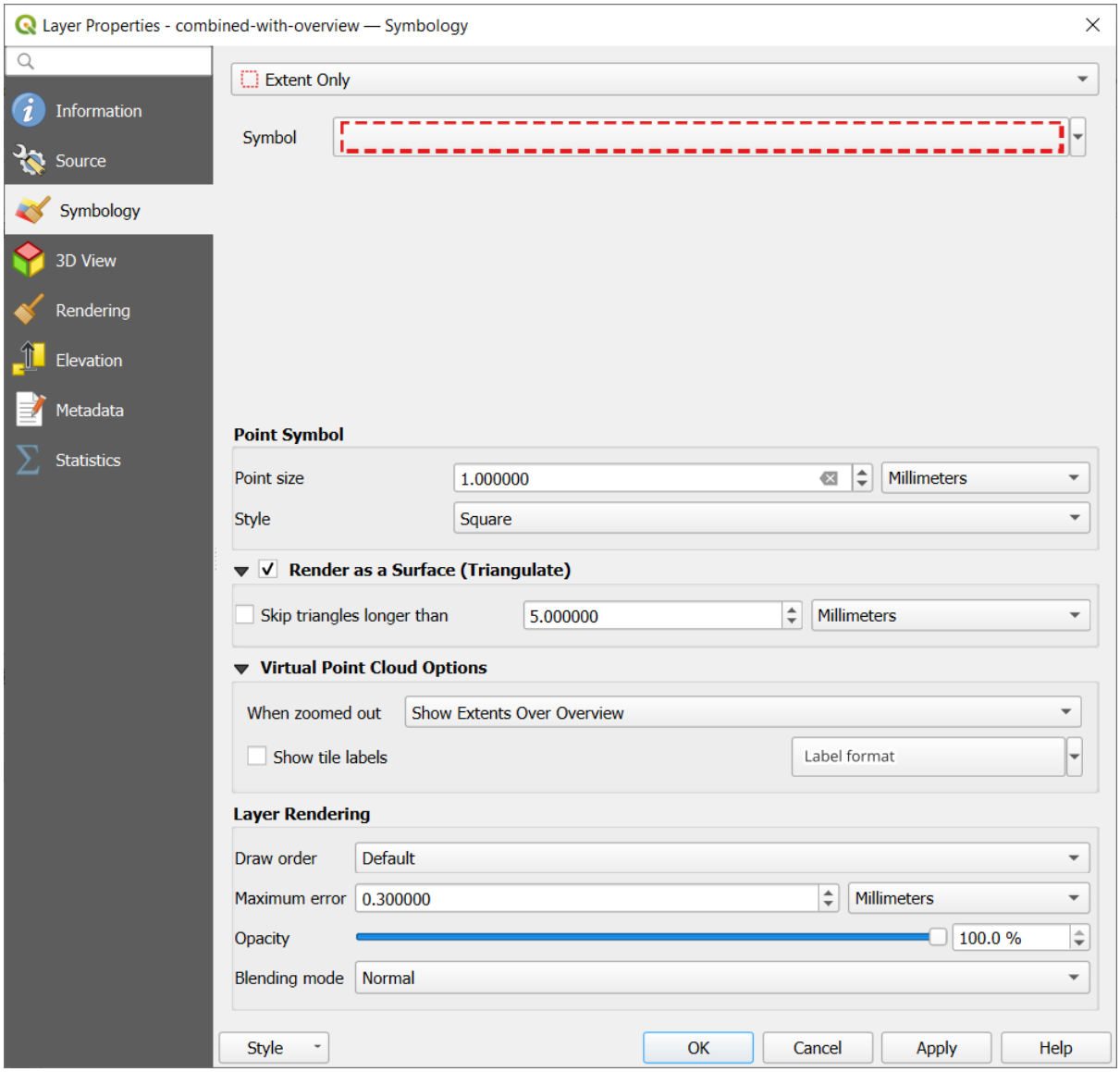


Figura 16.4: Pestaña simbología de la nube de puntos

Atributo por renderizador de rampa

Con *Atributo por rampa*, los datos pueden mostrarse por valores numéricos sobre un gradiente de color. Dichos valores numéricos pueden ser, por ejemplo, un atributo de intensidad existente o el valor Z. En función de un valor mínimo y un valor máximo, los demás valores se extienden al gradiente de color mediante interpolación. Los distintos valores y su asignación a un determinado color se denominan «mapa de color» y se muestran en la tabla. Hay varias opciones de ajuste, que se describen debajo de la figura.

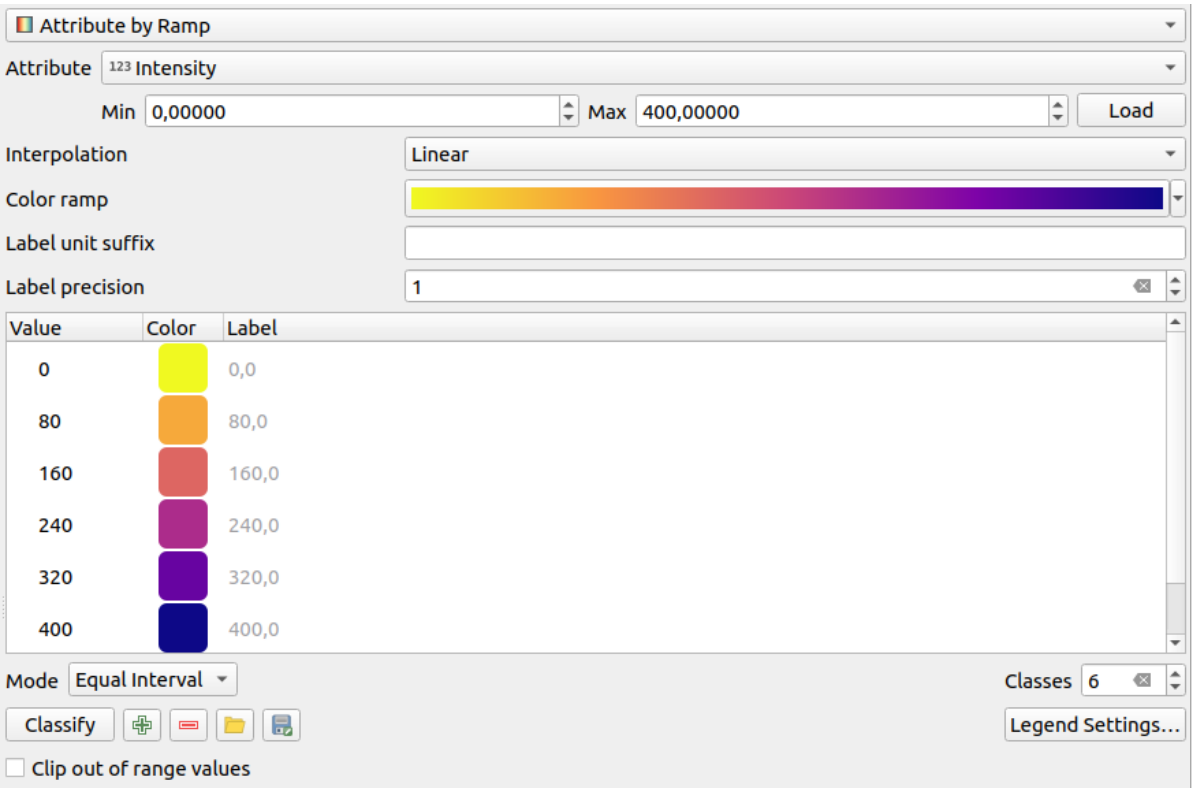


Figura 16.5: Pestaña simbología de la nube de puntos: Atributo por rampa

- *Mín* y *Máx* definen el rango que se aplica a la rampa de color: el valor *Mín* representa el extremo izquierdo, el valor *Máx* el extremo derecho de la rampa de color, los valores intermedios se interpolan. Por defecto QGIS detecta el mínimo y el máximo del atributo seleccionado, pero pueden modificarse. Una vez modificados los valores, puede restaurar los valores por defecto pulsando el botón *Cargar*.
- La entrada *Interpolación* define cómo se asigna el color a los valores:
  - *Discreto* (aparece un símbolo <= en la cabecera de la columna *Valor*): El color se toma de la entrada del mapa de colores más cercana con un valor igual o superior
  - *Lineal* El color se interpola linealmente a partir de las entradas del mapa de colores situadas por encima y por debajo del valor del píxel, lo que significa que a cada valor del conjunto de datos le corresponde un color único
  - *Exacto* (aparece un símbolo = en la cabecera de la columna *Valor*): Sólo a los píxeles con valor igual a una entrada del mapa de color se les aplica un color; los demás no se renderizan.
- El widget *Rampa de color* le ayuda a seleccionar la rampa de color que desea asignar al conjunto de datos. Como es habitual con *este widget*, puede crear uno nuevo y editar o guardar el seleccionado actualmente.
- *Sufijo unidad etiqueta* añade una etiqueta después del valor en la leyenda, y *Precisión etiqueta* controla el número de decimales a mostrar.

La clasificación *Modo* le ayuda a definir cómo se distribuyen los valores entre las clases:

- *Continuo*: El número de clase y el color se obtienen de las paradas de la rampa de color; los valores límite se establecen siguiendo la distribución de paradas en la rampa de color (puede encontrar más información sobre paradas en *Estableciendo una Rampa de Color*).
- *Intervalo igual*: El número de clases se establece mediante el campo *Classes* al final de la línea; los valores límite se definen de forma que todas las clases tengan la misma magnitud.

Las clases se determinan automáticamente y se muestran en la tabla del mapa de colores. Pero también puede editar estas clases manualmente:

- Hacer doble clic en una *Valor* de la tabla permite modificar el valor de la clase

- Al hacer doble clic en la columna *Color* se abre el widget *Selector de color*, donde puede seleccionar un color para aplicarlo a ese valor
- Doble clic en la columna *Etiqueta* para modificar la etiqueta de la clase
- Al hacer clic con el botón derecho sobre las filas seleccionadas en la tabla de colores, aparece un menú contextual para *Cambiar color…* y *Cambiar opacidad…* para la selección

Debajo de la tabla están las opciones para restaurar las clases por defecto con *Clasificar* o para manualmente Agregar valores o Borrar valores seleccionados de la tabla.

Dado que un mapa de color personalizado puede ser muy complejo, también existe la opción de Cargar un mapa de color existente o de Guardar para utilizarlo en otras capas (como un archivo `txt`).

Si ha seleccionado *Lineal* para *Interpolación*, también puede configurar:

- *Recortar valores fuera de rango* Por defecto, el método lineal asigna el color de la primera clase (o de la última) a los valores del conjunto de datos que son inferiores al valor *Mín* (o superiores al valor *Máx*). Marque esta opción si no desea mostrar estos valores.
- *Configuración de la leyenda*, para su visualización en el panel *Capas* y en la *leyenda de composición*. La personalización funciona del mismo modo que con una capa ráster (encontrará más detalles en *Personalizar leyenda ráster*).

Renderizador RGB

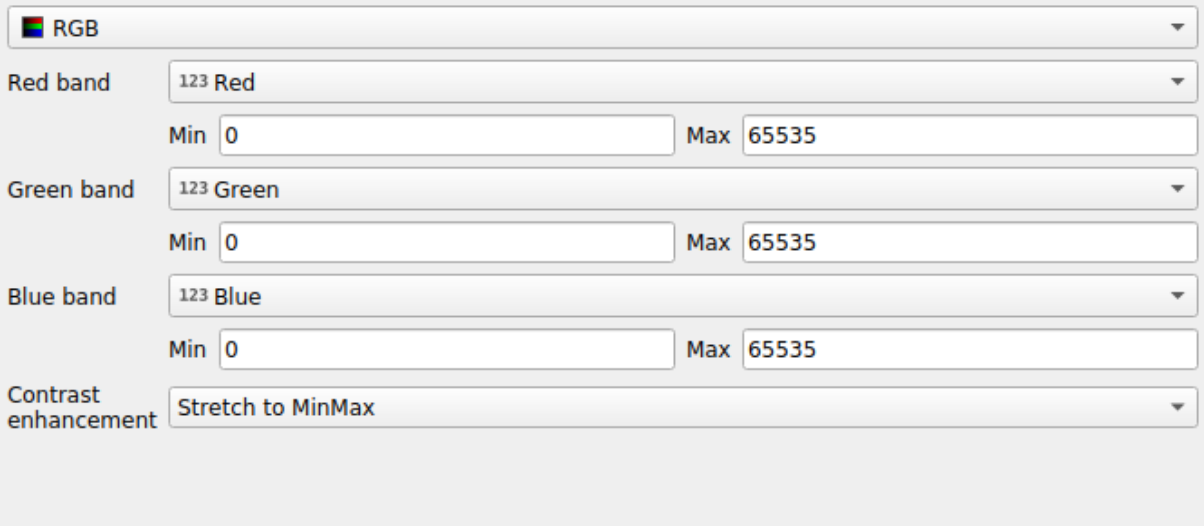


Figura 16.6: El renderizador RGB de nubes de puntos

Con el renderizador *RGB*, se utilizarán tres atributos seleccionados de la nube de puntos como componentes rojo, verde y azul. Si los atributos tienen el nombre correspondiente, QGIS los selecciona automáticamente y obtiene los valores *Mín* y *Máx* para cada banda y escala la coloración en consecuencia. Sin embargo, también es posible modificar los valores manualmente.

Se puede aplicar un método *Mejora del contraste* a los valores: *Sin mejora*, *Stretch to MinMax*, *Estirar y recortar a MinMax* y *Recortar a MinMax*.

Nota

La herramienta *Mejora del contraste* aún está en desarrollo. Si tiene problemas con ella, debería utilizar el ajuste por defecto *Estirar a MínMáx*.

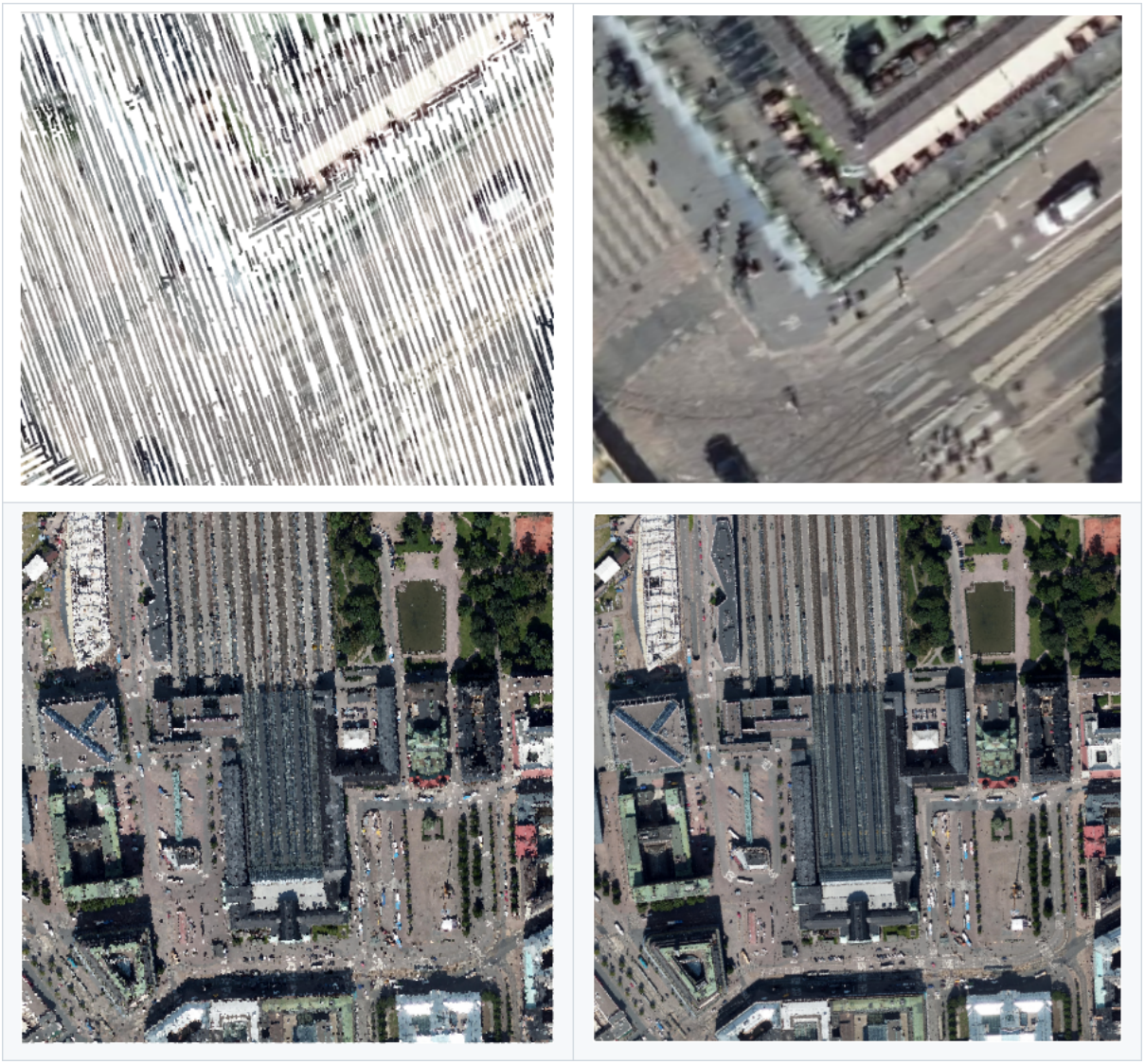


Figura 16.7: Ejemplo de renderizador RGB (izquierda) combinado con la opción de triangulación de superficies (derecha)

Renderizador de clasificación

En la representación *Clasificación*, la nube de puntos se muestra diferenciada por colores en función de un atributo. Se puede utilizar cualquier tipo de atributo (numérico, cadena, …). Los datos de las nubes de puntos suelen incluir un campo llamado `Clasificación`. Suele contener datos determinados automáticamente por el postprocesado, por ejemplo, sobre la vegetación. Con *Atributo* puede seleccionar el campo de la tabla de atributos que se utilizará para la clasificación. Por defecto, QGIS utiliza las definiciones de la especificación LAS (ver tabla "ASPRS Standard Point Classes" en el PDF en ASPRS home page). Sin embargo, los datos pueden desviarse de este esquema; en caso de duda, deberá solicitar las definiciones a la persona o institución de la que haya recibido los datos.

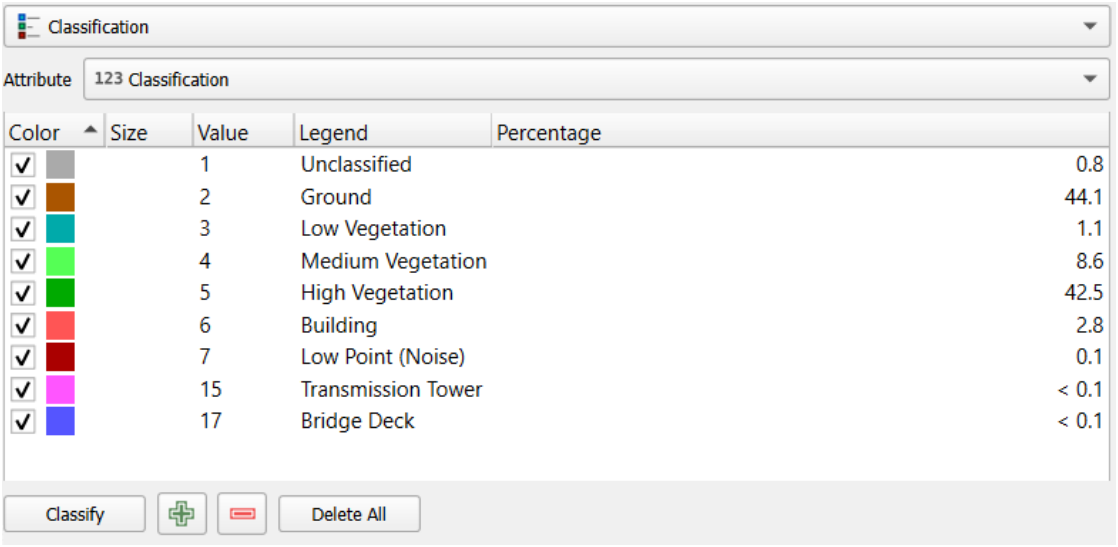


Figura 16.8: El renderizador de clasificación de nubes de puntos

En la tabla se muestran todos los valores utilizados con el color y la leyenda correspondientes. Al principio de cada fila hay una casilla de verificación ; si no está marcada, este valor ya no se muestra en el mapa. Con un doble clic en la tabla, puede modificar:

- la etiqueta *Color*: se abre el widget *Selector de color*
- la *Tamaño*: asignar un tamaño de `0` a una categoría hará que vuelva a utilizar el tamaño predeterminado *tamaño del punto* establecido para la capa
- el *Valor*
- la *Leyenda*
- la columna *Porcentaje* le mostrará la representación de la categoría dentro de la capa.

También puede hacer clic con el botón derecho del ratón en una o varias filas para abrir un menú contextual con las opciones para cambiar el color, la opacidad o el tamaño de la fuente.

Debajo de la tabla hay botones con los que puede cambiar las clases por defecto generadas por QGIS:

- Con el botón *Clasificar* se pueden clasificar los datos automáticamente: se añaden todos los valores que aparecen en los atributos y que aún no están presentes en la tabla
- Con Añadir y Borrar, se pueden añadir o eliminar valores manualmente.
- *Borrar todo* elimina todos los valores de la tabla

Consejo

En el panel *Capas*, puede hacer clic con el botón derecho del ratón sobre una entrada de hoja de clase de una capa para configurar rápidamente la visibilidad de los objetos espaciales correspondientes.

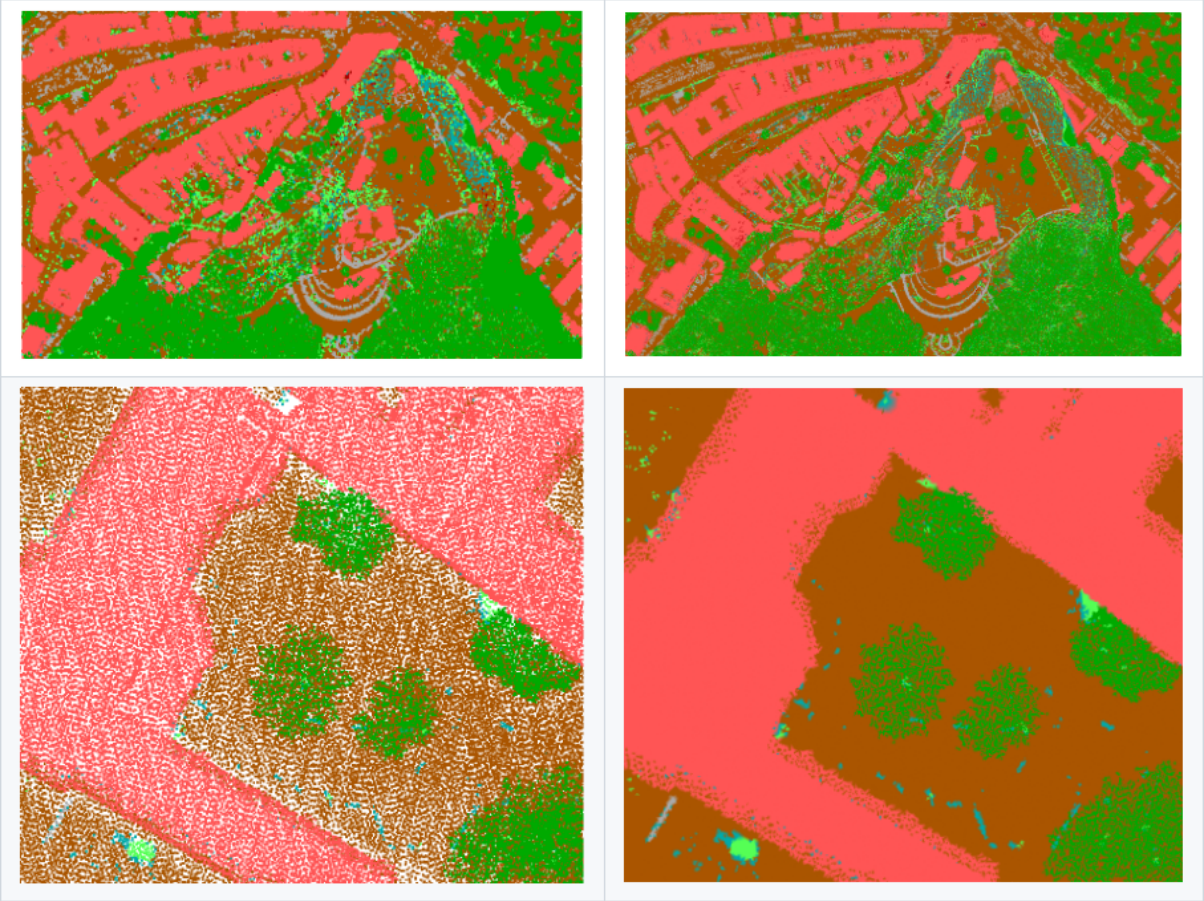


Figura 16.9: Ejemplo de renderizador de clasificación (izquierda) combinado con la opción de triangulación de superficies (derecha)

Símbolo de punto

En *Símbolo de punto*, puede establecerse el tamaño y la unidad (por ejemplo, milímetros, píxeles, pulgadas) con la que se muestra cada punto de datos. Se puede seleccionar *Círculo* o *Cuadrado* como estilo para los puntos.

Renderizar como superficie (triangular)

Marque la casilla de verificación *Renderizar como superficie (triangular)* para habilitar la triangulación de la capa de nube de puntos en la vista 2D. Esta opción permite renderizar triángulos en lugar de puntos. Cada punto mantiene su color para la interpolación en el triángulo. Puede controlar la longitud horizontal de los triángulos calculados:

Al marcar la casilla de verificación *Omitir triángulos más largos que* y configurar el valor umbral, puede controlar la longitud máxima de un lado de los triángulos que se tendrán en cuenta en el plano horizontal. Esto puede resultar especialmente útil si desea identificar agujeros reales en los datos.

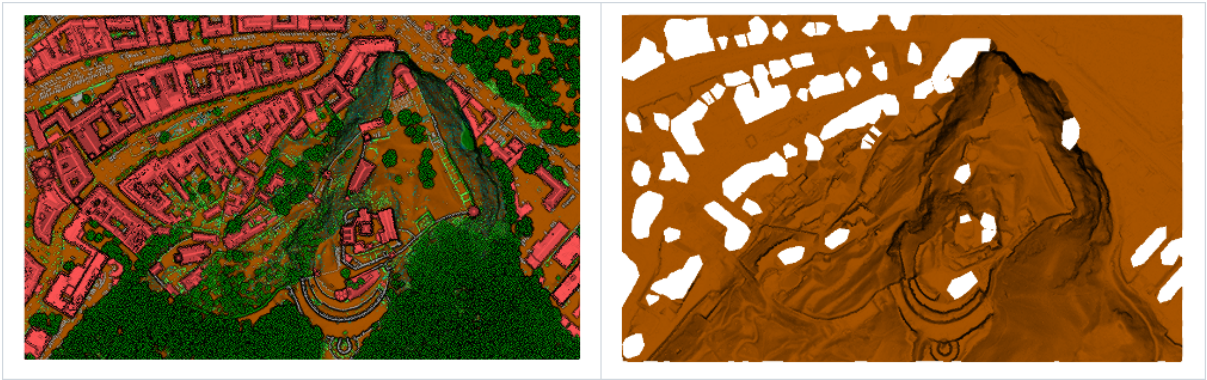


Figura 16.10: Representación de datos como una superficie con sombreado de mapa (izquierda) y con sombreado de mapa, filtrando triángulos grandes (derecha)

Virtual Point Cloud Options

The *Virtual Point Cloud Options* are available only when the layer is a *virtual point cloud (VPC)*. QGIS renders the overview of the VPC when zoomed out, if an overview is present. This provides a seamless transition from the overview to the detailed display of individual point clouds as you zoom in.

You can control how the VPC is displayed when zoomed out using the available options:

- *Show Extents Only*: Only the extents of the underlying point clouds are displayed.
- *Show Overview Only*: Only the overview is displayed (if available).
- *Show Extents Over Overview*: The extents are displayed on top of the overview.

You can also choose to *Show tile labels* to display the tile names and set the label format.

Renderizado de capas

En la sección *Renderizado de capas* dispone de las siguientes opciones para modificar el renderizado de la capa:

- *Orden de representación*: permite controlar si el orden de representación de las nubes de puntos en el lienzo del mapa 2D debe basarse en su valor Z. Es posible renderizar :
  - con el orden *Predeterminado* en el que se almacenan los puntos en la capa,
  - de *De abajo a arriba* (los puntos con valores Z mayores cubren los puntos inferiores dando el aspecto de una verdadera foto ortográfica),
  - o desde *De arriba a abajo* donde la escena aparece vista desde abajo.
- *Máximo error*: Las nubes de puntos suelen contener más puntos de los necesarios para su visualización. Con esta opción se establece lo densa o escasa que será la visualización de la nube de puntos (también puede entenderse como *máximo espacio permitido entre puntos*). Si estableces un valor alto (por ejemplo, 5 mm), habrá espacios visibles entre los puntos. Un valor bajo (por ejemplo, 0,1 mm) podría forzar el renderizado de una cantidad innecesaria de puntos, haciendo el renderizado más lento (se pueden seleccionar diferentes unidades).
- *Opacidad*: Con esta herramienta puede hacer visible la capa subyacente en el lienzo del mapa. Utilice el control deslizante para adaptar la visibilidad de la capa a sus necesidades. También puede definir con precisión el porcentaje de visibilidad en el menú situado junto al deslizador.
- *Modo de mezcla*: Con esta herramienta puede conseguir efectos de renderizado especiales. Los píxeles de las capas superpuestas y subyacentes se mezclan mediante los ajustes descritos en *Modos de Mezcla*.
- *Iluminación de la cúpula del ojo*: aplica efectos de sombreado al lienzo del mapa para una mejor representación de la profundidad. La calidad del renderizado depende de la propiedad *orden de dibujo* ; el orden de dibujo *Predeterminado* puede dar resultados subóptimos. Se pueden controlar los siguientes parámetros:

  - *Fuerza*: aumenta el contraste, lo que permite una mejor percepción de la profundidad
  - *Distancia*: representa la distancia de los píxeles utilizados respecto al píxel central y tiene el efecto de hacer los bordes más gruesos.

16.2.4 Propiedades de visor 3D

En la pestaña  *Vista 3D* puedes realizar los ajustes para el renderizado de la nube de puntos en mapas 3D.

Modos de renderizado 3D

Las siguientes opciones pueden seleccionarse en el menú desplegable de la parte superior de la pestaña:

- *Sin representación*: Los datos no se muestran
- *Seguir Simbología 2D*: Sincroniza los objetos espaciales renderizados en 3D con la *simbología asignada en 2D.*
-  *Color Único*: Todos los puntos se muestran en el mismo *color* sean cuales sean sus atributos
-  *Atributo por rampa*: Interpola un atributo dado sobre una rampa de color y asigna a las características su color correspondiente. Véase *Atributo por renderizador de rampa*.
-  *RGB*: Utiliza diferentes atributos de las características para establecer los componentes de color Rojo, Verde y Azul a asignarles. Véase *Renderizador RGB*.
-  *Classificación*: diferencia los puntos por color en función de un atributo. Véase *Renderizador de clasificación*.

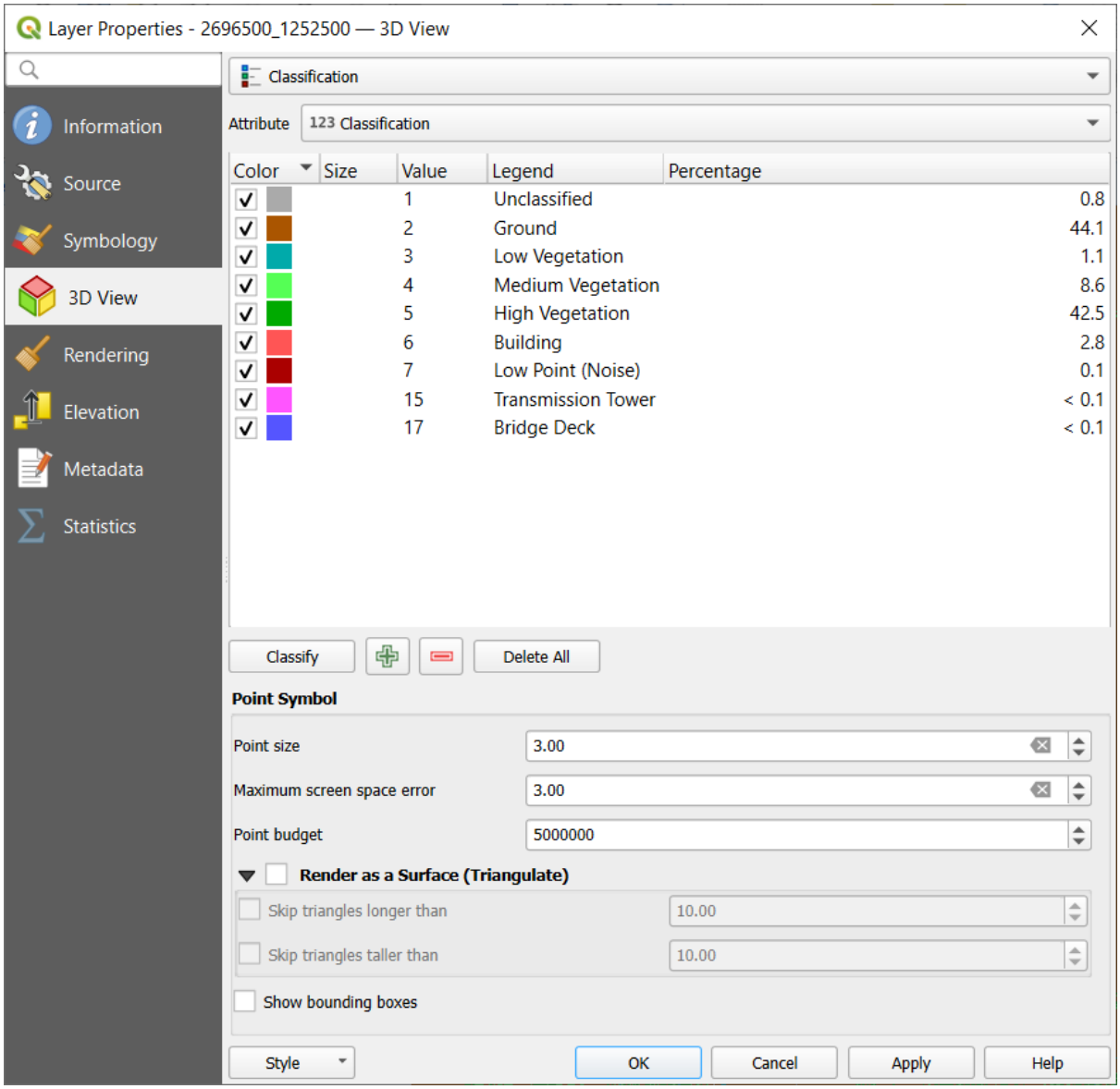


Figura 16.11: Pestaña de la vista 3D de la nube de puntos con el renderizador de clasificación

Símbolo de punto 3D

En la parte inferior de la pestaña *Vista 3D* se encuentra la sección *Símbolo de punto*. Aquí puede realizar ajustes generales para toda la capa que son los mismos para todos los renderizadores. Existen las siguientes opciones:

- *Tamaño del punto*: Se puede establecer el tamaño (en píxeles) con el que se muestra cada punto de datos
- *Error de espacio máximo de pantalla*: Con esta opción usted establece la densidad o escasez en la que será mostrada la nube de puntos (en pixels). Si establece un número grande (p.ej. 10), habrá espacios visibles entre los puntos; un valor bajo (p.ej. 0) podría forzar la entrega de una cantidad innecesaria de puntos, haciendo la representación más lenta (puede encontrar más detalles en *Simbología :ref:`Error máximo <point_clouds_symbology_maxerror>*).
- *Presupuesto de puntos*: Para evitar renderizaciones largas, puede establecer el número máximo de puntos que se renderizarán
- Marque *Renderizar como superficie (Triangular)* para renderizar la capa de nube de puntos en la vista 3D con una superficie sólida obtenida por triangulación. Puede controlar las dimensiones de los triángulos

calculados:

  - *Omitir triángulos mayores que* un valor umbral: establece en el plano horizontal, la longitud máxima de un lado de los triángulos a considerar
  - *Omitir triángulos más altos que* un valor umbral: establece en el plano vertical, la altura máxima de un lado de los triángulos a considerar

- *Mostrar cajas delimitadoras*: Especialmente útil para depuración, muestra las cajas delimitadoras de los nodos de la jerarquía

16.2.5 Propiedades de representación

En el cuadro de grupo *Visibilidad dependiente de la escala*, puede establecer la escala *Máximo (inclusive)* y *Mínimo (exclusivo)*, definiendo un rango de escala en el que las características serán visibles. Fuera de este rango, se ocultan. El botón Ajustar a la escala actual del lienzo le ayuda a utilizar la escala actual del lienzo del mapa como límite del rango de visibilidad. Véase *Selector de escala de visibilidad* para más información.

Nota

También puede activar la visibilidad dependiente de la escala en una capa desde el panel *Layers*: haga clic con el botón derecho del ratón en la capa y, en el menú contextual, seleccione *Definir la visibilidad de la escala de capas*.

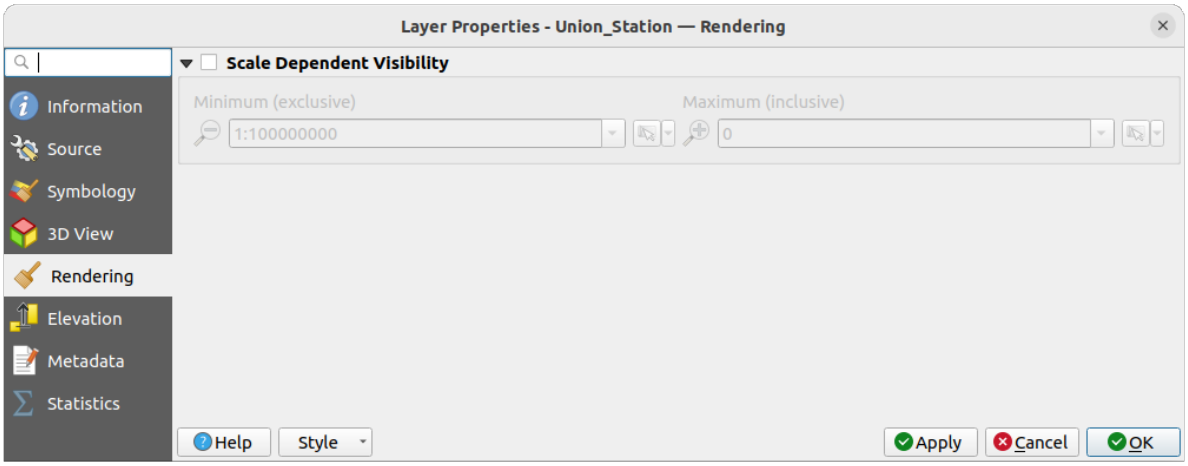


Figura 16.12: Pestaña de renderizado de nubes de puntos

16.2.6 Propiedades de elevación

En la pestaña *Elevación*, puede establecer correcciones para los valores Z de los datos. Esto puede ser necesario para ajustar la elevación de los datos en los mapas 3D y su apariencia en los *gráficos de herramienta de perfil*. Existen las siguientes opciones:

- *Sistema de referencia vertical*: Si el SRC de su capa de nube de puntos es compuesto (incluida una dimensión Z), el SRC vertical utilizado para la capa se derivará automáticamente del componente vertical del SRC de la capa. En este caso, no puede establecer manualmente un SRC vertical diferente, y la opción para cambiarlo estará desactivada. Si su capa de nube de puntos utiliza un SRC horizontal (2D) (aunque es poco común), puede seleccionar manualmente un SRC vertical específico haciendo clic en Seleccionar SRC. Los sistemas de referencia verticales son compatibles con las capas de nube de puntos en:

  - *Perfiles de elevación*
  - *Identificar resultados de la herramienta*
  - *Vistas de mapas 3D*

- En el grupo *Elevación*:
  - Se puede establecer una *Escala*: Si se introduce aquí `10`, un punto que tenga un valor Z = `5` se mostrará a una altura de `50`.
  - Se puede introducir un *desplazamiento* al nivel z. Esto es útil para hacer coincidir diferentes fuentes de datos en su altura entre sí. Por defecto, se utiliza como valor el valor z más bajo contenido en los datos.

    Este valor también puede restaurarse con el botón  Actualizar al final de la línea.
- En *Exactitud del gráfico de perfil*, *Error máximo* le ayuda a controlar la densidad o dispersión de los puntos que se representarán en el perfil de elevación. Los valores mayores dan como resultado una generación más rápida con menos puntos incluidos.
- En *Apariencia del Gráfico de Perfiles*, puede controlar la visualización de los puntos:
  - *Tamaño del punto*: el tamaño con el que se representan los puntos, en las unidades admitidas (milímetros, unidades de mapa, píxeles, …)
  - *Estilo*: si renderizar los puntos como *Círculo* o *Cuadrado*.
  - Aplicar una única *Color* a todos los puntos visibles en la vista de perfil.
  - Marque la casilla *Respetar la coloración de la capa* para mostrar los puntos con el color asignado a través de su *simbología 2D*
  -  *Aplicar el efecto de opacidad por distancia a la curva*, reduciendo la opacidad de los puntos que están más alejados de la curva del perfil.

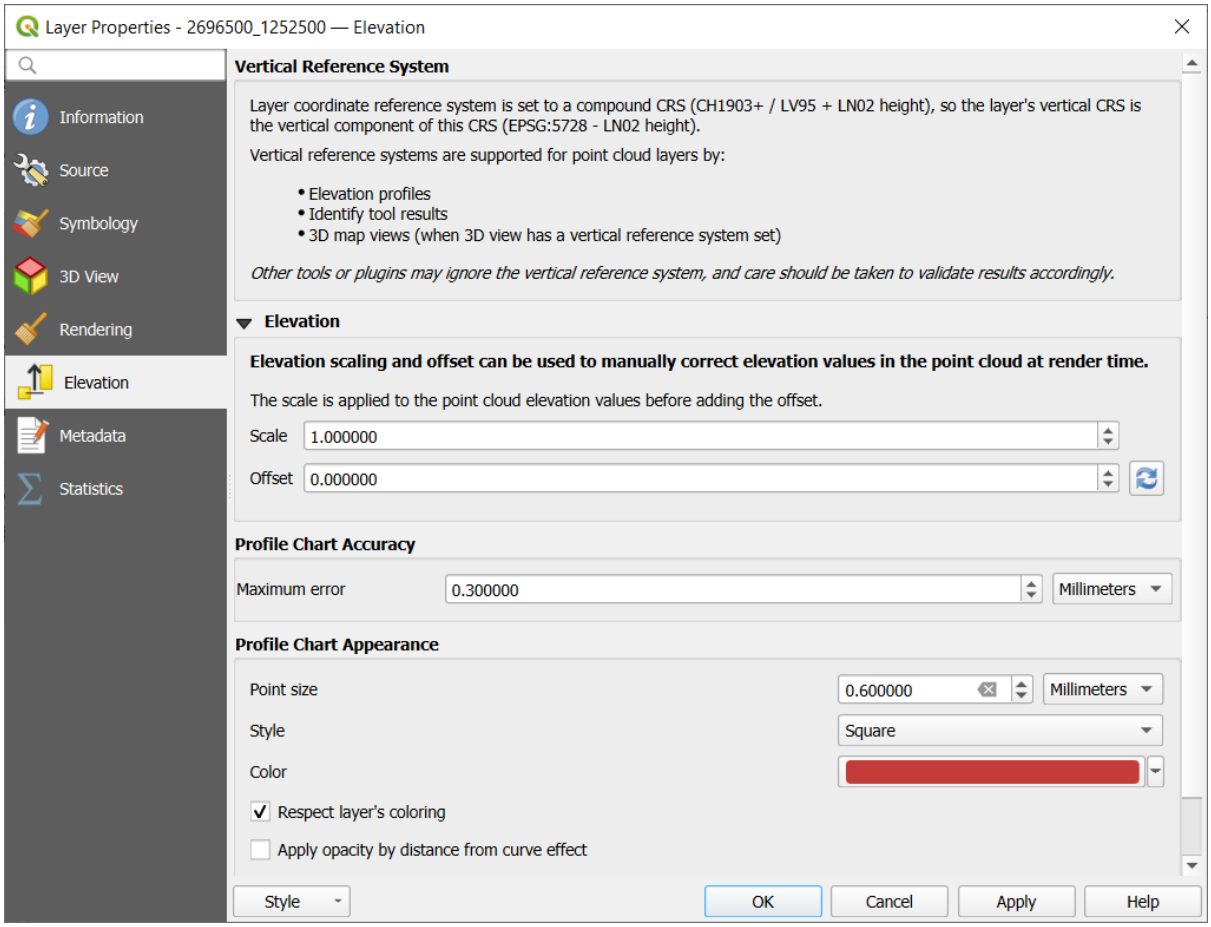


Figura 16.13: Pestaña de elevación de la nube de puntos

16.2.7 Propiedades de metadatos

El *Metadata* le brinda opciones para crear y editar un informe de metadatos en su capa. Consulte *Metadatos* para obtener más información.

16.2.8 Propiedades estadísticas

En la pestaña *Estadísticas* puede obtener una visión general de los atributos de su nube de puntos y su distribución.

En la parte superior encontrará la sección *Estadística de atributos*. Aquí se listan todos los atributos contenidos en la nube de puntos, así como algunos de sus valores estadísticos: *Mínimo*, *Máximo*, *Media*, *Desviación estándar*.

Si hay un atributo *Clasificación*, entonces hay otra tabla en la sección inferior. Aquí se enumeran todos los valores contenidos en el atributo, así como su abundancia absoluta *Conteo* y relativa %.

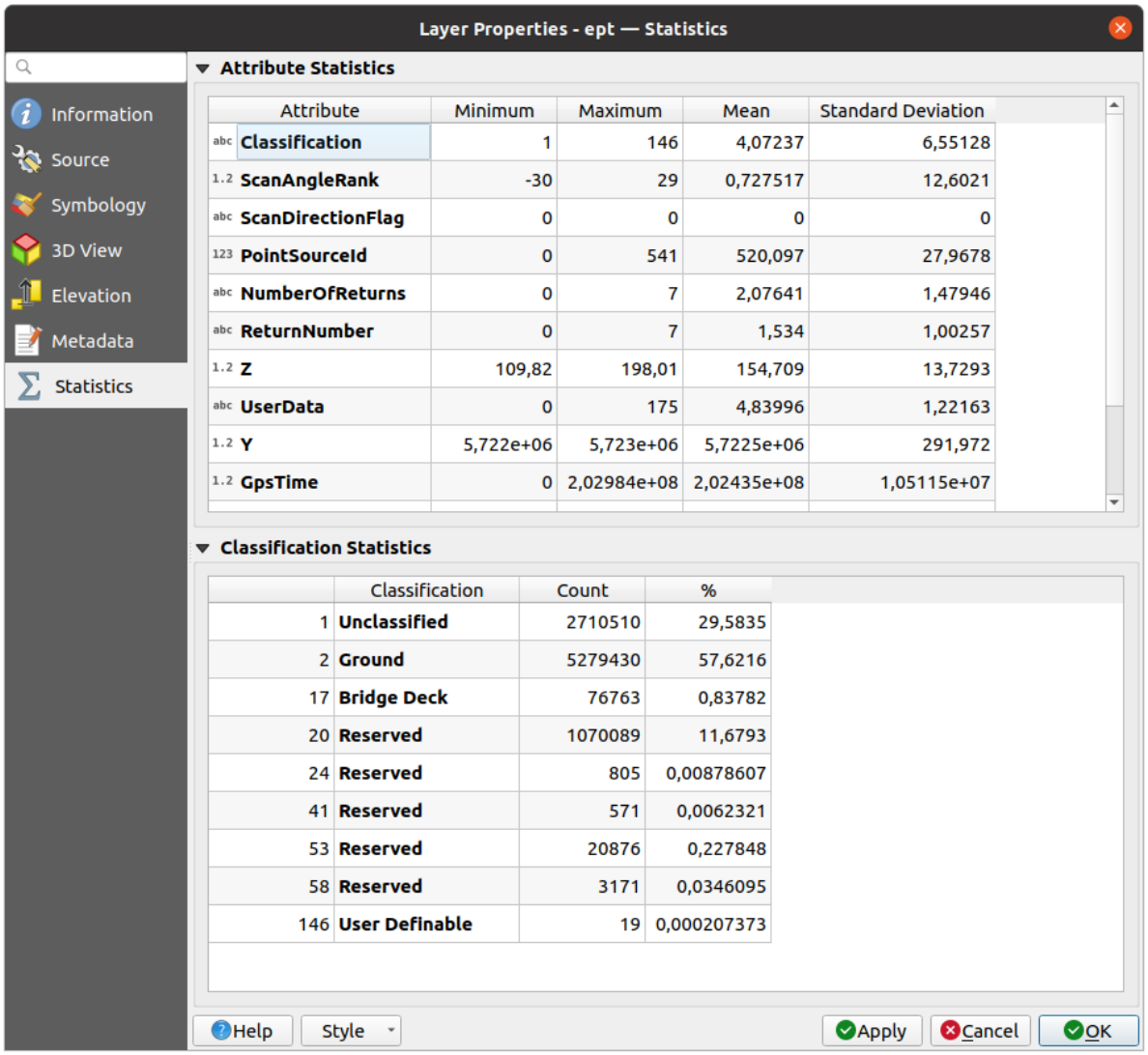


Figura 16.14: Pestaña de estadísticas de la nube de puntos

16.3 Nube de puntos virtual

Los estudios Lidar de grandes áreas son a menudo conjuntos de datos de varios terabytes con muchos miles de millones de puntos. Representar estos grandes conjuntos de datos en un único archivo de nube de puntos no resulta práctico debido a las dificultades de almacenamiento, transferencia, visualización y análisis. Por ello, los datos de nubes de puntos suelen almacenarse y distribuirse divididos en mosaicos cuadrados (por ejemplo, `1km x 1km`), cada uno de los cuales tiene un tamaño de archivo más manejable (por ejemplo, ~200 MB cuando está comprimido).

El mosaico de datos resuelve los problemas relacionados con el tamaño de los datos, pero plantea problemas a la hora de procesar o visualizar un área de interés que no cabe por completo en un único mosaico. Los usuarios tienen que desarrollar flujos de trabajo que tengan en cuenta varios mosaicos y hay que tener especial cuidado con los datos situados cerca de los bordes de los mosaicos para evitar artefactos no deseados en los resultados. Del mismo modo, cuando se visualizan datos de nubes de puntos, resulta engorroso cargar muchos archivos individuales y aplicar la misma simbología.

Este es un ejemplo de varios mosaicos de nubes de puntos cargados en QGIS. Cada mosaico tiene un estilo basado en los valores Z mínimo/máximo del mosaico, creando artefactos visibles en los bordes de los mosaicos. El estilo tiene que ser ajustado para cada capa por separado:

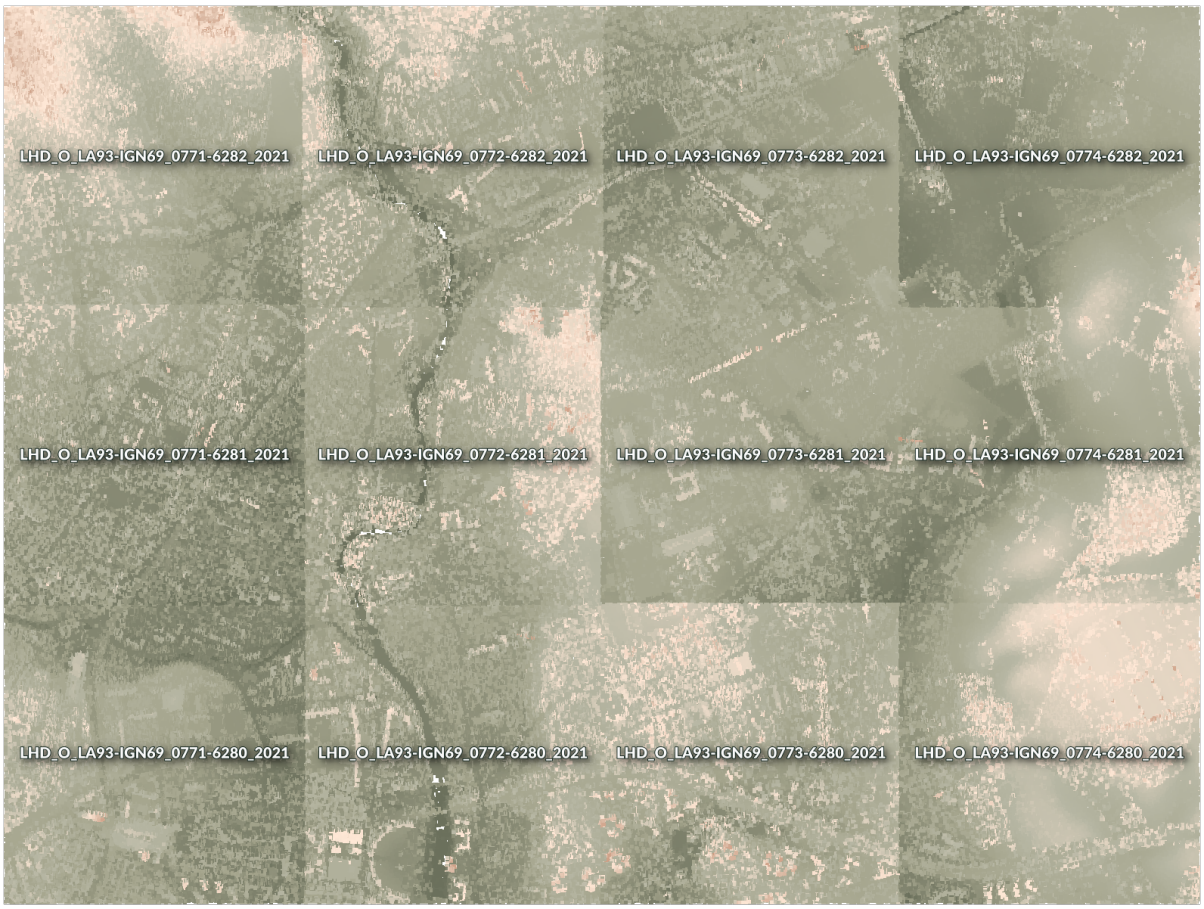


Figura 16.15: Nubes de puntos individuales cargadas, con artefactos en los bordes

En el mundo de los SIG, muchos usuarios están familiarizados con el concepto de archivos ráster virtuales. Un ráster virtual es un archivo que simplemente hace referencia a otros archivos ráster con datos reales. De este modo, el software SIG trata todo el conjunto de datos compuesto por muchos archivos como una única capa ráster, lo que facilita enormemente la visualización y el análisis de todos los rásters incluidos en el archivo virtual.

Tomando prestado el concepto de ráster virtual de GDAL, **la nube de puntos virtual (VPC)** es un formato de archivo que hace referencia a otros archivos de nubes de puntos. El software compatible con nubes de puntos virtuales maneja todo el conjunto de datos en mosaico como una única fuente de datos.

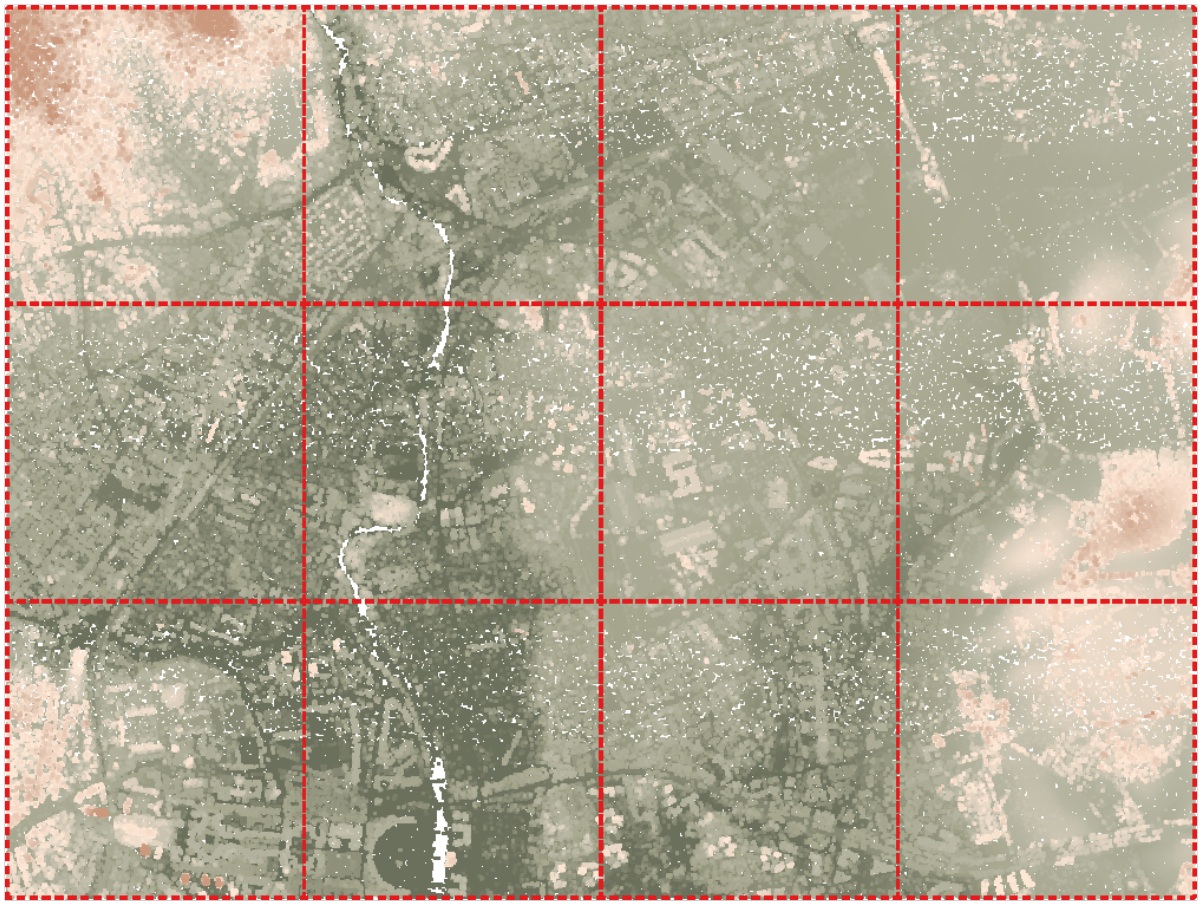


Figura 16.16: La nube de puntos virtual

La visualización y manipulación de nubes de puntos virtuales es mucho más fluida y sencilla.

En esencia, un archivo de nube de puntos virtual es un simple archivo JSON con extensión `.vpc`, que contiene referencias a archivos de datos reales (por ejemplo, archivos `.LAS`, `.LAZ` o `.COPC`) y metadatos adicionales extraídos de los archivos. Aunque es posible escribir archivos VPC a mano, se recomienda encarecidamente crearlos utilizando una herramienta automatizada, como:

- El algoritmo Procesos *Crear una nube de puntos virtual (VPC)*
- El comando `build_vpc` de la herramienta PDAL wrench

Para más detalles, consulte la especificación VPC que también contiene las mejores prácticas y extensiones opcionales (como resúmenes).

CAPÍTULO 17

Trabajar con teselas 3D

17.1 ¿Qué son las teselas 3D?

Las teselas 3D son una especificación para la transmisión y el renderizado de conjuntos de datos geoespaciales 3D a gran escala. Utilizan una estructura jerárquica para gestionar y mostrar de forma eficiente el contenido 3D, optimizando el rendimiento mediante la carga dinámica de los niveles de detalle adecuados. Esta tecnología se utiliza ampliamente en planificación urbana, arquitectura, simulación, juegos y realidad virtual, proporcionando una solución estandarizada e interoperable para visualizar datos geográficos complejos.



Figura 17.1: Ejemplo de teselas 3D

Consulte *Utilización de servicios de escena 3D en mosaico* para obtener instrucciones sobre cómo añadir estas fuentes de datos a QGIS.

17.1.1 State of 3D Tiles Support

Actualmente, QGIS admite dos formatos de teselas 3D:

- «Cesium» 3D Tiles are used primarily for complex 3D models of buildings or whole cities. Such datasets can be provided by cloud-based platforms such as Cesium Ion or Google (*Photorealistic 3D Tiles*).
- Quantized Mesh tiles are used for terrain elevation data.

Supported *3D Tiles* features:

- QGIS currently has partial support for 3D Tiles 1.0 and partial support for 3D Tiles 1.1.
- For 3D Tiles 1.0, the tile format Batched 3D Model (`b3dm`) is supported. The tile formats Instanced 3D Model (`i3dm`), Point Cloud (`pnts`) and Composite (`cmpt`) are not supported yet.
- For 3D Tiles 1.1, the `3DTILES_content_gltf` extension is supported, so tile contents may also be encoded as glTF 2.0 (text or binary). The `EXT_mesh_gpu_instancing` extension is not supported yet.
- Currently only explicit tiling is supported. Implicit tiling is not supported yet
- Currently only tilesets using EPSG:4978 are supported.

17.2 Propiedades de las teselas 3D

El cuadro de diálogo teselas 3D *Propiedades de capa* ofrece las siguientes secciones:



17.2.1 Propiedades de información

La pestaña *Información* es de solo lectura y representa un lugar interesante para obtener rápidamente información resumida y metadatos en la capa actual. Los datos proporcionados son:

- en función del proveedor de la capa: nombre, URL, tipo y ruta de origen, número de niveles de zoom
- propiedades personalizadas, utilizadas para almacenar en el proyecto activo información adicional sobre la capa. Se pueden crear y gestionar más propiedades utilizando PyQGIS, concretamente a través del método setCustomProperty().
- el sistema de referencia de coordenadas: nombre, unidades, método, precisión, referencia (es decir, si es estático o dinámico)
- recogidos de los *metadatos rellenados*: acceso, extensiones, enlaces, contactos, historial…

17.2.2 Propiedades de fuente

La pestaña *Fuente* muestra información básica sobre el mosaico 3D seleccionado, incluyendo:

- el *Nombre de la capa* a mostrar en el *Panel de capas*;
- el *Sistema de Referencia de Coordenadas*: Muestra el *Sistema de referencia de coordenadas (SRC)* de la capa. Puede cambiar el SRC de la capa, seleccionando uno utilizado recientemente en la lista desplegable o haciendo clic en el botón Seleccionar SRC (véase *Selector del Sistema de Coordenadas de Referencia*). Utilice este proceso sólo si el SRC de la capa es incorrecto o no está especificado.

[1] Also available in the *Layer styling panel*

17.2.3 Propiedades de simbología

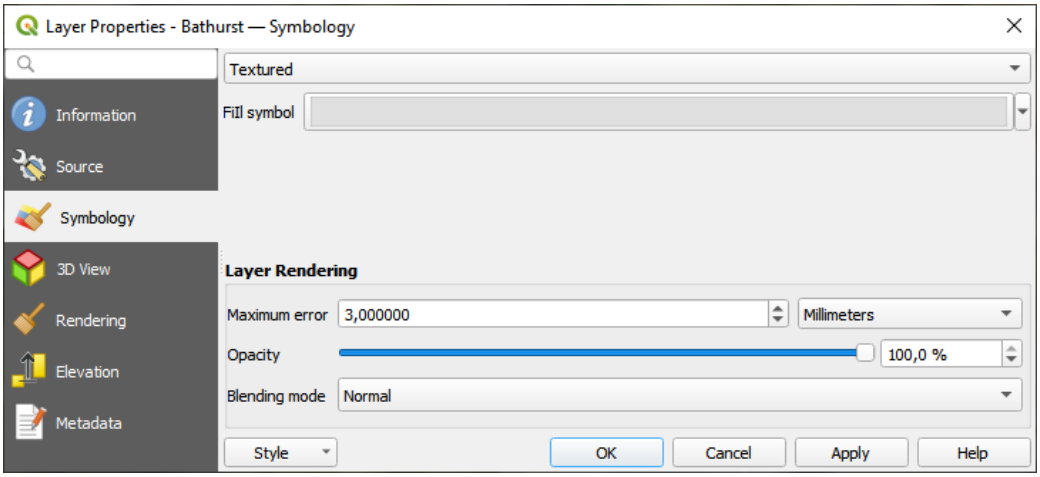


Figura 17.2: Simbología de capas de tesela 3D

De forma predeterminada, la capa está estilizada usando textura, pero puede cambiarla para ver la malla wireframe detrás de la escena eligiendo *Wireframe* en el menú desplegable. También puede cambiar el relleno de la malla y los símbolos de línea similares a los polígonos vectoriales. Marcando *Usar colores de textura* renderizará cada elemento de la malla con el valor medio de la textura completa. Esta es una buena opción para probar cuando se trata de un gran conjunto de datos y desea obtener una visión general rápida de los datos.

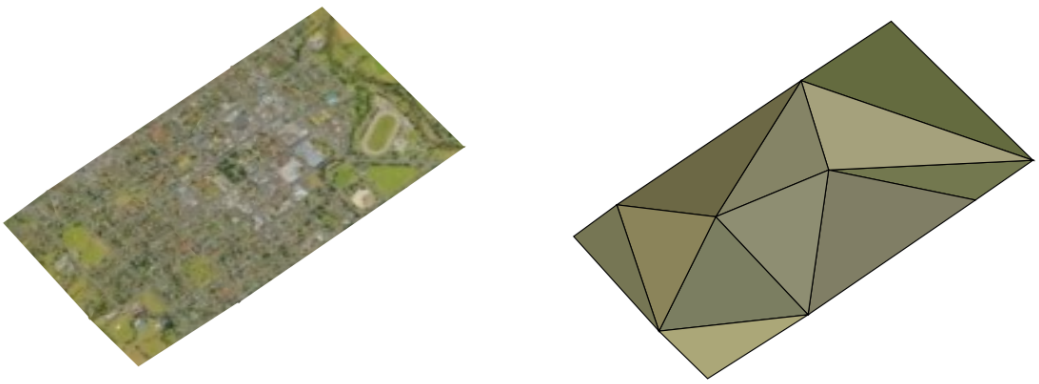


Figura 17.3: Teselas 3D - texturizadas y wireframe

Desde la pestaña *Simbología*, también puede configurar algunas opciones que actúan invariablemente sobre todas las características de la capa:

- *Error máximo*: Este parámetro determina el nivel de detalle mostrado en el modelo 3D. Al igual que las nubes de puntos, los mosaicos 3D suelen contener más información de la necesaria para su representación visual. Al ajustar este parámetro, se controla el equilibrio entre la densidad de visualización y la velocidad de renderizado. Un valor mayor (por ejemplo, 5 mm) puede introducir huecos perceptibles entre los elementos, mientras que un valor menor (por ejemplo, 0,1 mm) podría conducir a la representación de un número excesivo de detalles, lo que podría ralentizar el proceso de renderizado. Se pueden seleccionar diferentes unidades para adaptar el ajuste a sus necesidades específicas.

- *Opacity*: Adjusts the visibility of the underlying layer on the map canvas using this tool. Use slider to tailor the visibility of your scene layer according to your preferences. Alternatively, specify the exact percentage of visibility through the menu next to the slider.
- *Modo de mezcla*: Con estas herramientas puede conseguir efectos de renderizado especiales que antes sólo conocía de los programas gráficos. Los píxeles de las capas superpuestas y subpuestas se mezclan mediante los ajustes descritos en *Modos de Mezcla*.

17.2.4 Propiedades de visor 3D

- *Maximum screen space error*: Determines the threshold for swapping terrain tiles with more detailed ones (and vice versa) - i.e. how soon the 3D view will use higher quality tiles. Lower numbers mean more details in the scene at the expenses of increased rendering complexity.
- *Show bounding boxes*: Shows 3D bounding boxes of the terrain tiles (useful for troubleshooting terrain issues).

Para ver los datos puede abrir *Vista del nuevo mapa 3D*.

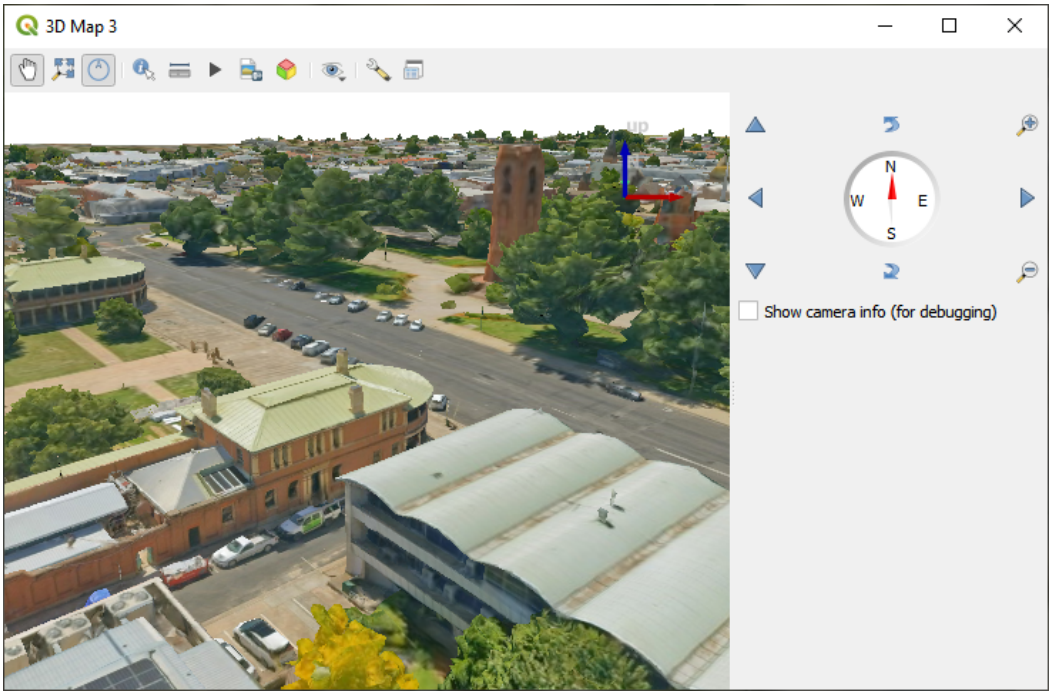


Figura 17.4: Vista de Mapa 3D

17.2.5 Propiedades de representación

En *Visibilidad dependiente de la escala*, puede establecer las escalas *Máxima (inclusive)* y *Mínima (exclusiva)*, definiendo un rango de escalas en el que las características serán visibles. Fuera de este rango, se ocultan. El botón Establecer en la escala actual del lienzo le ayuda a utilizar la escala actual del lienzo del mapa como límite del rango de visibilidad. Véase *Selector de escala de visibilidad* para más información.

17.2.6 Propiedades de elevación

La pestaña  *Elevación* proporciona opciones para controlar las propiedades de elevación de la capa dentro de una *vista de mapa 3D*. En concreto, puede establecer:

- *Superficie de elevación*: cómo deben interpretarse los valores Z de los vértices de la capa 3D como elevación del terreno. Se puede aplicar un factor *Escala* y un factor *Desplazamiento*.

17.2.7 Propiedades de metadatos

El  *Metadata* le brinda opciones para crear y editar un informe de metadatos en su capa. Consulte *Metadatos* para obtener más información.

CAPÍTULO 18

Diseñando los mapas

Con Imprimir diseños e informes, puede crear mapas y atlas, e imprimirlos o guardarlos como archivos de imagen, PDF o SVG.

18.1 Resumen de la Composición de Impresión

The print layout provides growing layout and printing capabilities. It allows you to add elements such as the QGIS 2D or 3D map canvas, text labels, images, legends, scale bars, basic shapes, arrows, attribute and simple tables, elevation profiles and HTML frames… You can size, group, align, position and rotate each element and adjust their properties to create your layout.

The layout can be printed or exported to image formats, PostScript, PDF or to SVG. You can save the layout as a template and load it again in another session. Finally, generating several maps based on a template can be done through the atlas generator.

18.1.1 Sesión de muestra para principiantes

Antes de empezar a trabajar con la composición de impresión, debe cargar algunas capas ráster o vectoriales en el lienzo de mapa de QGIS y adaptar sus propiedades a su conveniencia. Una vez que todo esté renderizado y simbolizado a su gusto, haga clic en el icono Nueva composición de Impresión de la barra de herramientas *Proyecto* o seleccione *Proyecto* ► *Nueva composición de impresión*. Se le pedirá que elija un título para la nueva maqueta.

Para demostrar como crear un mapa por favor siga las siguientes instrucciones.

1. En el lado izquierdo, seleccione el botón de la barra de herramientas Agregar mapa y describa un rectángulo en el lienzo manteniendo pulsado el botón izquierdo del ratón. Dentro del rectángulo dibujado, la vista del mapa QGIS del lienzo.

2. Seleccione el botón de la barra de herramientas Agregar Barra de Escala y haga click con el botón izquierdo del ratón en el lienzo del compositor de diseño. Una barra de escala será agregada al lienzo.

3. Seleccione el botón de la barra de herramientas Agragar leyenada y describa un rectángulo en el lienzo manteniendo pulsado el botón izquierdo del ratón. Dentro del rectángulo descrito será dibujada la leyenda.

4. Seleccione el icono  Seleccionar/Mover elemento para seleccionar el mapa en el lienzo y moverlo un poco.
5. Mientras el elemento del mapa aún está seleccionado, también puede cambiar el tamaño del elemento del mapa. Haga clic mientras mantiene presionado el botón izquierdo del ratón, en un pequeño rectángulo blanco en una de las esquinas del elemento del mapa y arrástrelo a una nueva ubicación para cambiar su tamaño.
6. Haga clic en el panel *Propiedades del elemento* de la parte inferior izquierda y busque la configuración de la orientación. Cambie el valor del ajuste *Orientación del mapa* a "15.00°". Debería ver cómo cambia la orientación del elemento del mapa.
7. Ahora, puede imprimir o exportar su diseño de impresión a formatos de imagen, PDF o SVG con las herramientas de exportación en el menú *Diseño*.
8. Finalmente, puede guardar su diseño de impresión dentro del archivo del proyecto con el botón  Guardar proyecto.

Puede añadir varios elementos al diseño de impresión. También es posible tener más de una vista de mapa o leyenda o barra de escala en el lienzo del diseño de impresión, en una o varias páginas. Cada elemento tiene sus propias propiedades y, en el caso del mapa, su propia extensión. Si desea eliminar algún elemento del lienzo de diseño, puede hacerlo con las teclas `Borrar` o `Retroceso`.

18.1.2 El Administrador de Composiciones

El *Administrador de Composiciones* es la ventana principal para administrar los diseños de impresión en el proyecto. Le brinda una descripción general de los diseños de impresión e informes existentes en el proyecto y ofrece herramientas para:

- buscar un diseño;
- agregar una nueva composición de impresión o un nuevo informe desde cero, plantilla o duplicando uno existente;
- renombrar o borrar cualquiera de ellos;
- abrirlos en el proyecto.

Para abrir el diálogo del Administrador de Composición:

- desde el diálogo principal de QGIS, seleccione el menú *Proyecto* ► *Administrador de Composiciones…* o haga click en el botón  Administrador de Composiciones en la *Barra de Herramientas de Proyecto*;
- desde un cuadro de diálogo de diseño o informe de impresión, seleccione *Diseño -> Layout Manager….* o haga clic en el botón  Administrador de Composiciones en *Barra de Herramientas Composiciones*.

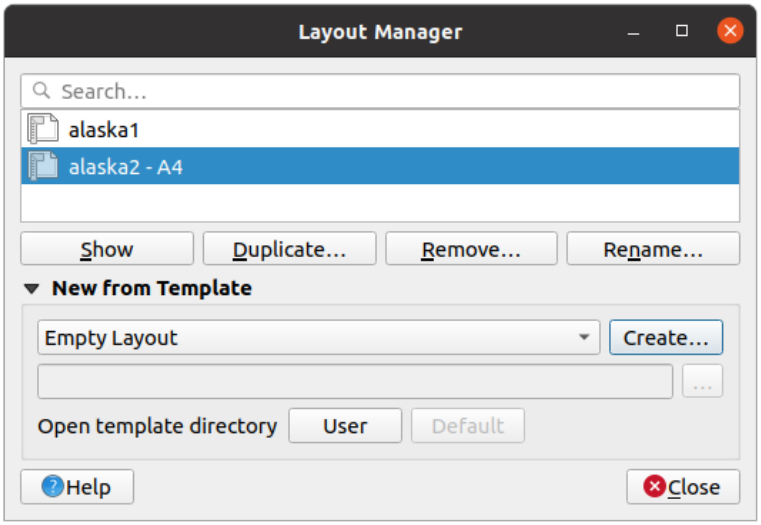


Figura 18.1: El Administrador de Impresión de Composición

El administrador de composición enumera en su parte superior todos los diseños de impresión o informes disponibles en el proyecto con herramientas para:

- mostrar la selección: puede seleccionar varios informes y/o composicion(es) de impresión y abrirlos con un solo clic. Hacer doble clic en un nombre también lo abre;
- duplicar la composición de impresión o informe seleccionado (disponible solo si se selecciona un elemento): crea un nuevo cuadro de diálogo utilizando el seleccionado como plantilla. Se le pedirá que elija un nuevo título para el nuevo diseño;
- cambiar el nombre del informe o composición (disponible solo si se selecciona un elemento): se le pedirá que elija un nuevo título para el diseño;
- eliminar la composición: los diseños de composición(es) seleccionados se eliminarán del proyecto.

En la parte inferior, es posible crear nuevos diseños de impresión o informes a partir de cero o de una plantilla. Por omisión, QGIS buscará plantillas en el *perfil de usuario* y los directorios de plantillas de la aplicación (accesibles con los dos botones de la parte inferior del marco), sino también en cualquier carpeta declarada como *Ruta(s) para buscar plantillas de impresión adicionales* en *Configuración* ► *Opciones* ► *Diseños*. Las plantillas encontradas se listan en el combobox. Seleccione un elemento y pulse el botón *Crear* para generar un nuevo informe o diseño de impresión.

También puede utilizar plantillas de diseño de una carpeta personalizada; en ese caso, seleccione *específico* en la lista desplegable de plantillas, busque la plantilla y presione *Crear*.

Truco

Creación de composiciones de impresión basados en plantillas desde el panel del navegador

Arrastre y suelte un archivo de plantilla de composición de impresión `.qpt` desde cualquier explorador de archivos en el lienzo del mapa o haga doble clic en él en :ref:`panel del explorador `para generar una nueva composición de impresión a partir de la plantilla.

18.1.3 Menús, herramientas y paneles de la composición de impresión

Al abrir el diseño de impresión aparece un lienzo en blanco que representa la superficie del papel cuando se utiliza la opción de impresión. Inicialmente encontrará botones a la izquierda junto al lienzo para añadir elementos de diseño de impresión: Lienzos de mapas 2D o 3D, etiquetas de texto, imágenes, leyendas, barras de escala, formas básicas, flechas, tablas de atributos o simples, marcos HTML, perfiles de elevación,…. En esta barra de herramientas también encontrará botones para navegar, ampliar una zona y desplazar la vista por el diseño, así como botones para seleccionar cualquier elemento del diseño y mover la extensión de los elementos del mapa.

Figura 18.2 muestra la vista inicial del diseño de impresión antes de que cualquier elemento fuera añadido.

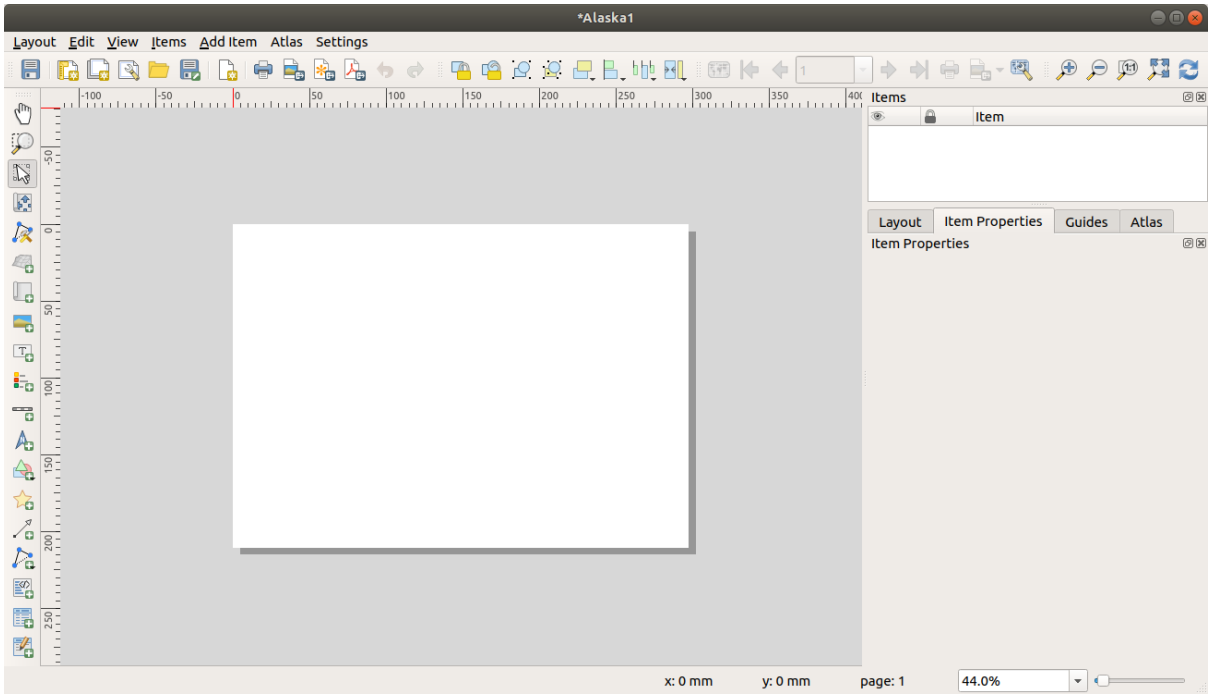


Figura 18.2: Diseño de impresión

A la derecha, al lado del lienzo, encontrará dos conjuntos de paneles. El superior contiene los paneles *elementos* y *deshacer historia* y el inferior contiene los paneles *Diseño*, *propiedades de elemento* y *Atlas*.

- El panel *elementos* proporciona una lista de todos los elementos del compositor de impresión añadidos al lienzo y formas de interactuar globalmente con ellos (ver *El Panel Elementos* para mas información).
- El panel *Historial de deshacer* muestra el historial de todos los cambios aplicados a la maqueta. Con un clic del ratón, es posible deshacer y rehacer pasos de maquetación hacia atrás y hacia delante hasta un determinado estado.
- El panel *Diseño* le permite establecer parámetros generales para aplicar al diseño al exportar o trabajar dentro (ver *El Panel de Diseño* para más detalles);
- El panel *Propiedades del elemento* muestra las propiedades del elemento seleccionado. Haga clic en el icono Seleccionar/Mover elemento para seleccionar un elemento (por ejemplo, leyenda, barra de escala o etiqueta) en el lienzo. Luego haga clic en el panel *Propiedades del elemento* y personalice la configuración del elemento seleccionado (consulte :ref:` layout_items` para obtener información detallada sobre la configuración de cada elemento).
- El panel *Atlas* le permite habilitar la generación de un atlas para el diseño actual y le da acceso a sus parámetros (consulte *Generar un Atlas* para obtener información detallada sobre el uso de la generación de atlas).

En la parte inferior de la ventana de diseño de impresión, puede encontrar una barra de estado con la posición del ratón, el número de página actual, un cuadro combinado para establecer el nivel de zoom, el número de elementos seleccionados si corresponde y, en el caso de la generación de atlas, el número de entidades.

En la parte superior de la ventana del compositor de impresión, puede encontrar menús y otras barras de herramientas. Todas las herramientas de diseño de impresión están disponibles en menús y como iconos en una barra de herramientas.

Las barras de herramientas y los paneles pueden desactivarse y activarse utilizando el botón derecho del ratón sobre cualquier barra de herramientas, o a través de *Ver* ► *Barras de herramientas* ► o *Ver* ► *Paneles* ►.

Menús y Herramientas

Menú Diseño

El *Diseño* proporciona acción para gestionar el diseño:

- Guarde el archivo del proyecto directamente desde la ventana del compositor de impresión.
- Crea una composición de impresión nueva y en blanco *Nueva Composición…*.
- *Duplicar Composición..* : Crea un nuevo diseño de impresión duplicando el actual.
- Eliminar el diseño actual con *Borrar Composición…*.
- Abrir el *Administrador de Composiciones…*.
- *Composiciones* ► : Abrir una composición de impresión existente.

Una vez diseñada la composición, con **|fileSaveAs | :guilabel:`Guardar como plantilla` y |fileOpen|** *Añadir elementos desde plantilla*, puede guardar el estado actual de una sesión de composición de impresión como un archivo de plantilla `.qpt` y cargar sus elementos nuevamente en otra sesión/composición de impresión.

En el menú *Diseño*, también hay formas poderosas de compartir información geográfica producida con QGIS que puede incluirse en informes o publicarse. Estas herramientas son *Exportar como imagen…*, *Exportar como PDF…*, *Exportar como SVG…* y *Print…*.

A continuación se muestra una lista de todas las herramientas disponibles en este menú con información conveniente.

Tabla 18.1: «Layout» Menu Tools

Herramienta	Atajos	Barra de herramietas	Referencia
Guardar Proyecto	`Ctrl+S`	*Composiciones*	*Introducción a proyectos QGIS*
Nueva Composición	`Ctrl+N`	*Composiciones*	*El Administrador de Composiciones*
Duplicar Composición		*Composiciones*	*El Administrador de Composiciones*
Eliminar Composición			
Layout Manager…		*Composiciones*	*El Administrador de Composiciones*
Layouts ►			
Propiedades de composición…			*El Panel de Diseño*
Renombrar Composición…			
Propiedades de la página…			*Trabajar con las propiedades de la p*
Añadir páginas…		*Composiciones*	*Trabajar con las propiedades de la p*
Añadir elementos desde plantilla		*Composiciones*	*Creando un elemento de composición*
Guardar como plantilla…		*Composiciones*	*El Administrador de Composiciones*
Exportar como imagen…		*Composiciones*	*Exportar como imagen*
Exportar como SVG…		*Composiciones*	*Exportar como SVG*

continúe en la próxima pá

Tabla 18.1 – proviene de la página anterior

Herramienta	Atajos	Barra de herramietas	Referencia
Exportar como PDF…		*Composiciones*	*Exportar como PDF*
Configuración de la página de la impresora…	`Ctrl+Shift+P`		
Print…	`Ctrl+P`	*Composiciones*	*Crear salida*
Close	`Ctrl+Q`		

Menú Editar

El menú *Editar* ofrece herramientas para manipular elementos de diseño de impresión. Incluye acciones comunes como herramientas de selección, copiar/cortar/pegar y deshacer/rehacer (ver *El panel de historial de deshacer: acciones de reversión y restauración*) funcionalidad para los elementos del diseño.

Al usar la acción Pegar, los elementos se pegarán de acuerdo con la posición actual del ratón. Usando la opción *Editar -> Pegar en lugar* o presionando :kbd:` Ctrl + Shift + V` pegará los elementos en la página actual, en la misma posición que estaban en su página inicial. Asegura copiar / pegar elementos en el mismo lugar, de una página a otra.

A continuación se muestra una lista de todas las herramientas disponibles en este menú con información conveniente.

Tabla 18.2: «Edit» Menu Tools

Herramienta	Atajos	Barra de herramieta	Referencia
Deshacer(último cambio)	`Ctrl+Z`	*Composicio*	*El panel de historial de deshacer: acciones de reversión y restauración*
Rehacer (último cambio)	`Ctrl+Y`	*Composicio*	*El panel de historial de deshacer: acciones de reversión y restauración*
Borrar	`Del`		
Cortar	`Ctrl+X`		
Copiar	`Ctrl+C`		
Pegar	`Ctrl+V`		
Pegar en lugar	`Ctrl+Shift+V`		
Seleccionar todo	`Ctrl+A`		
Deseleccionar todo	`Ctrl+Shift+A`		
Invertir Selección			
Seleccionar el siguiente elemento debajo	`Ctrl+Alt+[`		
Seleccionar el siguiente elemento encima	`Ctrl+Alt+]`		
Desplzar Composición	`P`	*Caja de Herramient*	
Zoom	`Z`	*Caja de Herramient*	
Seleccionar/Mover elemento	`V`	*Caja de Herramient*	*Interactuando con los elementos de composición*
Mover Contenido	`C`	*Caja de Herramient*	*El elemento del mapa*

continúe en la próxima página

Tabla 18.2 – proviene de la página anterior

Herramienta	Atajos	Barra de herramieta	Referencia
Editar Elementos de Nodos		*Caja de Herramient*	*Los elementos de forma basados en nodos*

Ver menú

El menú *Ver* dá acceso a las herramientas de navegación y ayuda a configurar el comportamiento general del compositor de impresión. Además de las herramientas de zoom habituales, dispone de medios para:

- Actualizar vista (si encuentra la vista en un estado inconsistente);
- habilite una *cuadrícula* a la que podría ajustar elementos al moverlos o crearlos. La configuración de las cuadrículas se realiza en *Configuración -> Opciones de diseño…* o en *Panel de diseño*;
- habilite *guías* a las que podría ajustar elementos al moverlos o crearlos. Las guías son líneas rojas que puede crear haciendo clic en la regla (arriba o en el lado izquierdo del diseño) y arrastrar y soltar en la ubicación deseada;
- *Guías inteligentes*: utiliza otros elementos de diseño como guías para ajustar dinámicamente a medida que mueve o cambia la forma de un elemento;
- *Limpiar Guías* para borrar todas las guías actuales;
- *Mostrar Recuadros delimitadores* alrededor de los elementos para identificar mejor su selección;
- *Mostrar reglas* alrededor del diseño;
- *Mostrar páginas* o configurar páginas transparentes. A menudo, el diseño se utiliza para crear diseños no impresos, p. Ej. para su inclusión en presentaciones u otros documentos, y es deseable exportar la composición utilizando un fondo totalmente transparente. A veces se lo denomina «lienzo infinito» en otros paquetes de edición.

En la composición de impresión, puede cambiar el nivel de zoom usando la rueda del ratón o el control deslizante y el cuadro combinado en la barra de estado. Si necesita cambiar al modo panorámico mientras trabaja en el área de diseño, puede mantener presionada la tecla `Barra espaciadora` o la rueda del ratón. Con `Ctrl + Barra espaciadora`, puede cambiar temporalmente al modo Acercar, y con :kbd:` Ctrl + Alt + Barra espaciadora`, al modo Alejar.

Los paneles y barras de herramientas se pueden habilitar desde el men *Ver* ->. Para maximizar el espacio disponible para interactuar con una composición, puede marcar el *Ver -> Alternar visibilidad del panel* o presione `Ctrl + Tab`; todos los paneles están ocultos y solo los paneles previamente visibles se restauran cuando se desmarca.

También es posible cambiar al modo de pantalla completa para tener más espacio para interactuar presionando `F11` o usando :menuselection:` Ver -> ` *Alternar pantalla completa*.

Tabla 18.3: «View» Menu Tools

Herramienta	Atajos	Barra de herramietas	Referencia
Actualizar	`F5`	*Navegación*	
Vista Previa►			
Acercarse	`Ctrl++`	*Navegación*	
Alejarse	`Ctrl+-`	*Navegación*	
Acercamiento a 100%	`Ctrl+1`	*Navegación*	
Zoom completo	`Ctrl+0`	*Navegación*	

continúe en la próxima página

Tabla 18.3 – proviene de la página anterior

Herramienta	Atajos	Barra de herramietas	Referencia
Zoom a la Anchura			
Mostrar cuadrícula	`Ctrl+'`		*Guías y Cuadrículas*
Ajustar a cuadrícula	`Ctrl+Shift+'`		*Guías y Cuadrículas*
Mostrar guías	`Ctrl+;`		*Guías y Cuadrículas*
Ajustar a guías	`Ctrl+Shift+;`		*Guías y Cuadrículas*
Guías inteligentes	`Ctrl+Alt+;`		
Administrar guías…			*El Panel Guías*
Limpiar guías			*El Panel Guías*
Mostrar reglas	`Ctrl+R`		
Mostrar rectángulos delimitadores	`Ctrl+Shift+B`		
Mostrar páginas			
Toolbars ►			*Paneles y Barras de Herramientas*
Paneles ►			*Paneles y Barras de Herramientas*
Alternar pantalla completa	`F11`		*Ver*
Alternar Visibilidad del Panel	`Ctrl+Tab`		*Ver*

Menú de Elementos

Los *Elementos* le ayuda a configurar la posición de los elementos en la composición y las relaciones entre ellos (consulte *Interactuando con los elementos de composición*).

Tabla 18.4: «Items» Menu Tools

Herramienta	Atajos	Barra de herramietas	Referencia
Agrupar	`Ctrl+G`	*Acciones*	*Agrupando elementos*
Desagrupar	`Ctrl+Shift+G`	*Acciones*	*Agrupando elementos*
Subir	`Ctrl+]`	*Acciones*	*Alineación y distribución*
Bajar	`Ctrl+[`	*Acciones*	*Alineación y distribución*
Traer al frente	`Ctrl+Shift+]`	*Acciones*	*Alineación y distribución*
Enviar al fondo	`Ctrl+Shift+[`	*Acciones*	*Alineación y distribución*
Bloquear elementos seleccionados	`Ctrl+L`	*Acciones*	*Bloqueando elementos*
Desbloquear todo	`Ctrl+Shift+L`	*Acciones*	*Bloqueando elementos*
Alinear elementos ►		*Acciones*	*Alineación y distribución*
Distribuir elementos ►		*Acciones*	*Moviendo y redimensionando elementos*
Cambiar tamaño ►		*Acciones*	*Moviendo y redimensionando elementos*

Menú añadir elemento

Estas son herramientas para crear elementos de diseño. Cada uno de ellos está profundamente descrito en el capítulo *Elementos del Layout*.

Tabla 18.5: «Add Item» Menu Tools

Herramienta	Barra de herramietas	Referencia
Añadir Mapa	*Caja de Herramientas*	*El elemento del mapa*
Agregar Mapa 3D	*Caja de Herramientas*	*El Elemento Mapa 3D*
Añadir Imagen	*Caja de Herramientas*	*El elemento Imagen*
Añadir Etiqueta	*Caja de Herramientas*	*El elemento etiqueta*
Añadir Texto Dinámico ►		*The Label Item*
Añadir Leyenda	*Caja de Herramientas*	*El elemento leyenda*
Añadir Barra de Escala	*Caja de Herramientas*	*El elemento de barra de escala*
Añadir Flecha del Norte	*Caja de Herramientas*	*El elemento Flecha del Norte*
Añadir Forma ►	*Caja de Herramientas*	*El elemento Forma Regular*
► *Agregar rectángulo*	*Caja de Herramientas*	*El elemento Forma Regular*
► *Agregar elipse*	*Caja de Herramientas*	*El elemento Forma Regular*
► *Agregar triángulo*	*Caja de Herramientas*	*El elemento Forma Regular*
Agregar marcador	*Caja de Herramientas*	*El elemento marcador*
Añadir Flecha	*Caja de Herramientas*	*El elemento flecha*
Añadir Elemento Nodo ►	*Caja de Herramientas*	*Los elementos de forma basados en nodos*
► *Agregar Polígono*	*Caja de Herramientas*	*Los elementos de forma basados en nodos*
► *Agregar Polilínea*	*Caja de Herramientas*	*Los elementos de forma basados en nodos*
Añadir HTML	*Caja de Herramientas*	*El elemento del marco HTML*
Agregar Tabla de Atributos	*Caja de Herramientas*	*El elemento de la tabla de atributos*
Agregar Tabla Fija	*Caja de Herramientas*	*El elemento tabla fija*
Añadir perfil de elevación	*Caja de Herramientas*	*Elemento del perfil de elevación*

Menú Atlas

Tabla 18.6: «Atlas» Menu Tools

Herramienta	Atajos	Barra de herramietas	Referencia
Vista Preliminar de Atlas	`Ctrl+ALt+/`	*Atlas*	*Obtenga una vista previa y genere un atlas*
Primer Objeto Espacial	`Ctrl+<`	*Atlas*	*Obtenga una vista previa y genere un atlas*
Objeto Espacial Anterior	`Ctrl+,`	*Atlas*	*Obtenga una vista previa y genere un atlas*
Objeto Espacial Siguiente	`Ctrl+.`	*Atlas*	*Obtenga una vista previa y genere un atlas*
Último Objeto Espacial	`Ctrl+>`	*Atlas*	*Obtenga una vista previa y genere un atlas*

continúe en la próxima página

Tabla 18.6 – proviene de la página anterior

Herramienta	Atajos	Barra de herramietas	Referencia
Print Atlas…		*Atlas*	*Obtenga una vista previa y genere un atlas*
Export Atlas as Images…		*Atlas*	*Obtenga una vista previa y genere un atlas*
Export Atlas as SVG…		*Atlas*	*Obtenga una vista previa y genere un atlas*
Export Atlas as PDF…		*Atlas*	*Obtenga una vista previa y genere un atlas*
Configuración de Atlas		*Atlas*	*Generar un Atlas*

Menú Configuración

El menú *Configuración* ► *Opciones de la composición…* es un atajo para el menú *Configuración* ► *Opciones* ► *Composiciones* del lien<o principal de QGIS. Aquí, puede configurar algunas opciones que se usarán por defecto en cualquier nueva composición de impresión:

- *Composición predeterminada* le permite especificar la fuente predeterminada a utilizar;
- Con *aspecto de la cuadrícula*,puede establecer el estilo de la cuadrícula y su color. Hay tres tipos de cuadrículas: **Puntos**, líneas **Solidas** y **Cruces**;
- *Valores predefinidos de cuadrícula y guía* define el espaciado, el desplazamiento y la tolerancia de la cuadrícula (ver *Guías y Cuadrículas* para mas detalles);
- *Rutas de las composiciones*: para administrar la lista de rutas personalizadas para buscar plantillas de impresión.

El menú *Configuración* ► *Atajos de teclado…* le permite utilizar el *gestor de atajos* en la interfaz de diseño de impresión.

Menús contextuales

Dependiendo de dónde haga click con el botón derecho en el cuadro de diálogo de composición de impresión, abre un menú contextual con varias funciones:

- Haga clic con el botón derecho en la barra de menú o en cualquier barra de herramientas y obtendrá la lista de paneles de composiciones y barras de herramientas que puede habilitar o deshabilitar con un solo clic.
- Haga clic derecho sobre una regla y podrá *Mostrar guías*, *Ajustar a guías*, *Administrar guías… `abrir el :ref:`Panel de guías<layout_guides_panel>* o :guilabel:` Limpiar guías`. También es posible ocultar las reglas.
- Haga click con el botón derecho en el lienzo de la composición de impresión y:
  - Podrá *Deshacer* y *Rehacer* los cambios recientes, o *Pegar* cualquier elemento copiado (solo disponible si no se selecciona ningún elemento).
  - Si hace clic en una página, también puede acceder al panel *Propiedades de la página* actual o *Eliminar página*.
  - Si hace clic en un elemento seleccionado, puede cortarlo o copiarlo, así como abrir el panel *Propiedades del elemento*.
  - Si se selecciona más de un elemento, puede agruparlos y/o desagruparlos si al menos un grupo ya está en la selección.
- Hacer clic con el botón derecho dentro de un cuadro de texto o un widget de spinbox de cualquier panel de diseño proporciona opciones de edición para manipular su contenido.

El Panel de Diseño

En el panel de *Diseño*, puede definir la configuración global de su composición de impresión.

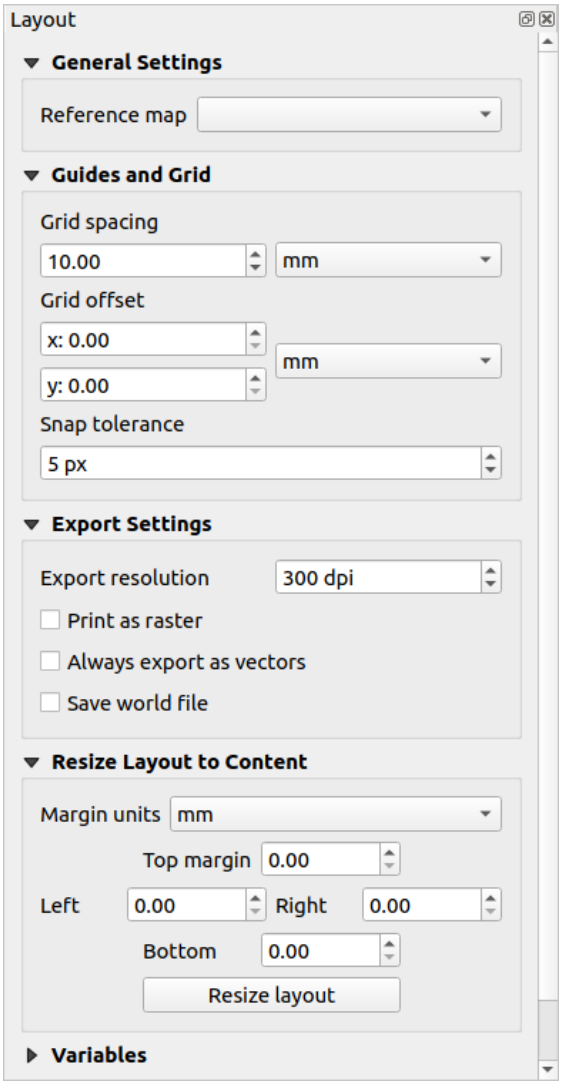


Figura 18.3: Configuración de diseño en el compositor de impresión

Configuración General

En una composición de impresión, puede utilizar más de un elemento de mapa. El *mapa de Referencia* representa el elemento del mapa que se utilizará como mapa maestro del diseño. Se asigna siempre que haya un elemento de mapa en el diseño. El compositor utilizará este mapa en cualquiera de sus propiedades y variables para calcular unidades o escala. Esto incluye exportar el diseño de impresión a formatos georreferenciados.

Además, los nuevos elementos de diseño, como la barra de escala, la leyenda o la flecha del norte, tienen por defecto sus ajustes (orientación, capas mostradas, escala, …) vinculados al elemento de mapa sobre el que se dibujan, y vuelven al mapa de referencia si no hay ningún mapa superpuesto.

Guías y Cuadrículas

Puede poner algunas marcas de referencia en su hoja de papel para ayudarle a colocar con precisión algunos elementos. Estas marcas pueden ser:

- líneas horizontales o verticales simples (llamadas **Guías**) colocadas en la posición que desee (consulte *El Panel Guías* para la creación de guías).
- o **Cuadrícula** regular: una red de líneas horizontales y verticales superpuestas sobre el diseño.

Configuraciones como *Espaciado de cuadrícula* o :guilabel:` Desplazamiento de cuadrícula` se pueden ajustar en este grupo, así como *Tolerancia de ajuste* para usar con los elementos. La tolerancia es la distancia máxima por debajo de la cual el cursor del mouse se ajusta a una cuadrícula o una guía, mientras se mueve, cambia el tamaño o crea un elemento.

Si se deben mostrar la cuadrícula o las guías se establece en el menú *Ver*. Allí, también puede decidir si se pueden usar para ajustar elementos de diseño. Cuando tanto una línea de cuadrícula como una línea de guía están dentro de la tolerancia de un punto, las guías siempre tendrán prioridad, ya que se han establecido manualmente (por lo tanto, se supone que se han colocado explícitamente en ubicaciones de ajuste altamente deseables y deben seleccionarse sobre el cuadrícula general).

> **Nota**
>
> En el menú *Configuración -> Opciones de composición*, también puede configurar los parámetros de cuadrícula y guías expuestos anteriormente. Sin embargo, estas opciones solo se aplicarán por defecto a los nuevos diseños de impresión.

Configuración de exportación

Puede definir una resolución a utilizar para todos los mapas exportados en *Exportar resolución*. Esta configuración se puede anular cada vez que exporta un mapa.

Debido a algunas opciones de renderizado avanzadas (*modo de fusión*, *efectos* …), un elemento de diseño puede necesitar rasterización para poder exportarse correctamente. QGIS lo rasterizará individualmente sin forzar que todos los demás elementos también se rastericen. Esto permite imprimir o guardar como PostScript o PDF para mantener los elementos tanto como sea posible como vectores, p. Ej. un elemento de mapa con opacidad de capa no forzará a rasterizar etiquetas, barras de escala, etc. Sin embargo, puedes:

- forzar la rasterización de todos los elementos marcando el cuadro *Imprimir como ráster*;
- o use la opción opuesta, es decir *Exportar siempre como vectores*, para forzar la exportación para mantener los elementos como vectores cuando se exportan a un formato compatible. Tenga en cuenta que, en algunos casos, esto podría hacer que la salida se vea diferente al diseño.

Donde el formato lo hace posible (por ejemplo,:file:.*TIF*, `.PDF`) exportar una composición de impresión da como resultado por defecto un archivo georreferenciado (basado en el elemento *Mapa de referencia* en el grupo *Configuración general*). Para otros formatos, la salida georreferenciada requiere que genere un archivo mundial marcando *Guardar archivo mundial*. El archivo mundial se crea junto al mapa(s) exportado, tiene el nombre de la salida de la página con el elemento del mapa de referencia y contiene información para georreferenciarlo fácilmente.

Cambiar el tamaño de la composición al contenido

Usando la herramienta *Cambiar tamaño de página* en este grupo, crea una composición de página única cuya extensión cubre el contenido actual de la composición de impresión (con algunos *márgenes* opcionales alrededor de los límites recortados).

Tenga en cuenta que este comportamiento es diferente de la opción:ref:*ajustar al contenido<crop_to_content>* en que todos los elementos se colocan en una página real y única en reemplazo de todas las páginas existentes.

Variables

La lista de *Variables* enumera todas las variables disponibles en el nivel del diseño (que incluye todas las variables globales y del proyecto).

También permite al usuario administrar variables a nivel de diseño. Haga clic en el  para agregar una nueva variable de nivel de diseño personalizada. Del mismo modo, seleccione una variable de nivel de diseño personalizada de la lista y haga clic en el botón  para quitarlo.

Más información sobre el uso de variables en la sección Herramientas generales.

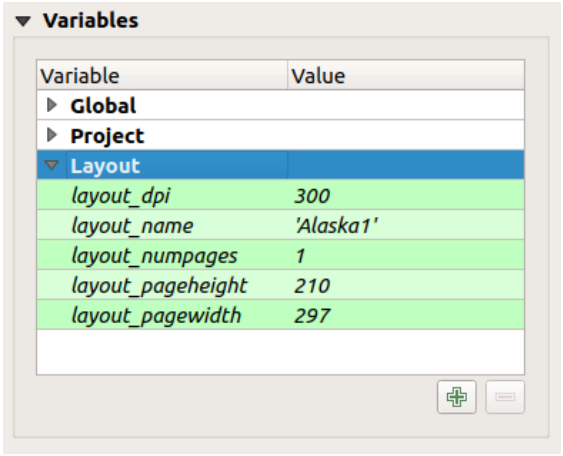


Figura 18.4: Editor de variables en el compositor de impresión

Trabajar con las propiedades de la página

Un diseño puede estar compuesto por varias páginas. Por ejemplo, una primera página puede mostrar un lienzo de mapa y una segunda página puede mostrar la tabla de atributos asociada con una capa, mientras que una tercera muestra un marco HTML que se vincula al sitio web de su organización. O puede agregar muchos tipos de elementos en cada página.

Añadiendo una nueva página

Además, se puede hacer un diseño usando diferentes tamaños y/u orientaciones de páginas. Para agregar una página, seleccione  *Agregar páginas…* desde el menú *Diseño* o *Barra de Herramientas Composiciones*. Se abre el cuadro de diálogo *Insertar páginas* y se le pide que complete:

- el número de páginas para insertar;
- la posición de la página(s): antes o después de una página determinada o al final de la composición de impresión;
- El *Tamaño de página*: podría ser de un formato de página preestablecido (`A4`, `B0`, `Legal`, `Carta`, `ANSI A`, `Arch A` y sus derivados, así como un tipo de resolución, como `1920x1080` o `1024x768`) con una asociada *Orientación* (Vertical u Horizontal).

  El tamaño de la página también puede tener un formato `personalizado`; En ese caso, deberá ingresar su *Anchura* y *Altura* (con la proporción de tamaño bloqueada si es necesario) y seleccionar la unidad que se usará `mm`, `cm`, `px`, `pt`, `in`, `ft`… La conversión de los valores ingresados se aplica automáticamente al cambiar de una unidad a otra.

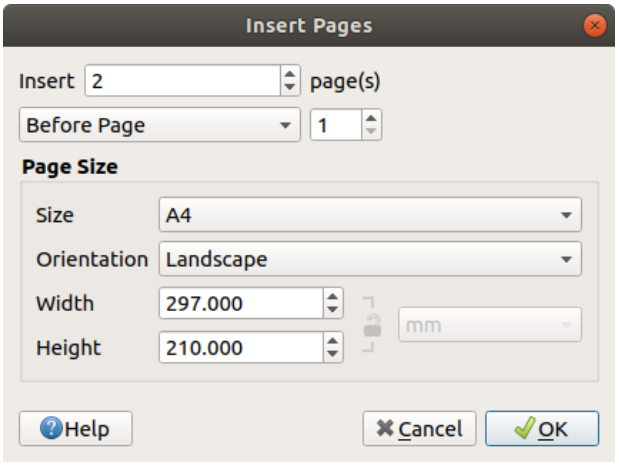


Figura 18.5: Crear una nueva página en la composición de impresión

Actualizar las propiedades de la página

Cualquier página se puede personalizar posteriormente a través del panel Página *Propiedades del elemento*. Para acceder a las propiedades de una página, puede:

- Haga clic con el botón izquierdo del ratón en una sección vacía de la página.
- Haga clic con el botón derecho del ratón en una página y seleccione *Propiedades de la página…*.
- Vaya a *Diseño* ► *Propiedades de la página…* mientras pasa el cursor por encima de la página.

El panel *Propiedades del elemento* se abre con ajustes como:

- para las maquetas con varias páginas, muestra el número de página activa y el recuento de páginas disponibles

- el marco *Tamaño de página* descrito anteriormente. Puede modificar cada propiedad utilizando las opciones de anulación definidas por datos (consulte *Explorar los botones de suplantación definida por datos con atlas* para un uso);
- El *Fondo* de la página actual usando el *color* o símbolo que quiera.
- the *Apply to all Pages* button updates the current page properties to the other existing pages in the layout;
- la *Excluir la página de las exportaciones* para controlar si la página actual con su contenido debe incluirse en la *salida de diseño*;

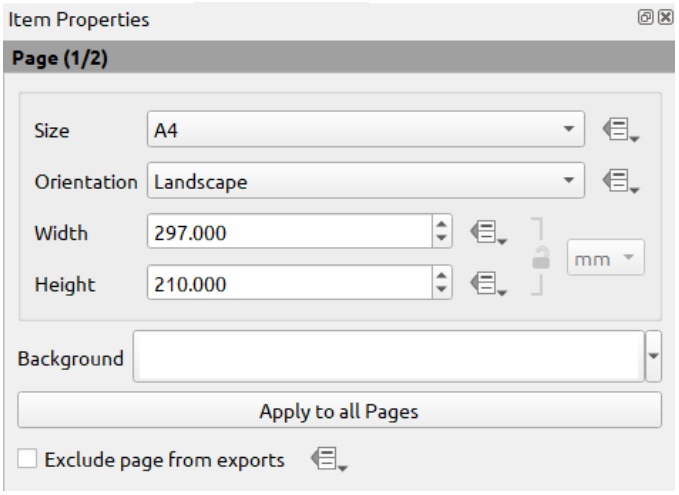


Figura 18.6: Diálogo de propiedades de página

El Panel Guías

Las guías son referencias de líneas verticales u horizontales que puede colocar en una página de diseño para ayudarlo a colocar los elementos, al crearlos, moverlos o cambiar su tamaño. Para estar activo, las guías requieren las opciones de *Ver* -> *Mostrar guías* y :menuselection:` Ver -> Ajustar a guías` para ser marcadas. Para crear una guía, existen dos métodos diferentes:

- Si la opción *Ver* ► *Mostrar reglas* está habilitada, arrastre una regla y suelte el botón del mouse dentro del área de la página, en la posición deseada.
- para mayor precisión, use el panel *Guías* de :menuselection:` Ver -> Caja de herramientas -> ` o seleccionando :guilabel:` Administrar guías para la página … `del menú contextual de la página.

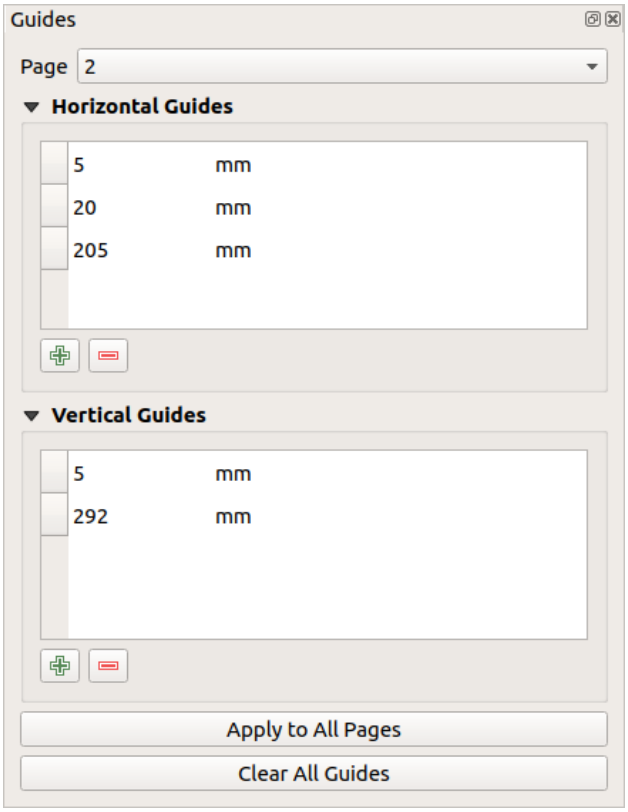


Figura 18.7: El Panel Guías

El panel *Guías* permite la creación de líneas de ajuste en ubicaciones específicas:

1. Seleccione la *Página* a la que le gustaría agregar las guías
2. Haga clic en el botón Agregar nueva guía e ingrese las coordenadas de la línea horizontal o vertical. El origen está en la esquina superior izquierda. Hay diferentes unidades disponibles para esto.

   El panel también permite ajustar la posición de las guías existentes a las coordenadas exactas: haga doble clic y reemplace el valor.
3. El panel *Guías* enumera solo los elementos de la página actual. Permite la creación o eliminación de guías solo en la página actual. Sin embargo, puede usar el botón *Aplicar a todas las páginas* para replicar la configuración de la guía de la página actual en las otras páginas del diseño.
4. Para eliminar una guía, selecciónela y presione el botón Eliminar guía seleccionada. Utilice *Borrar todas las guías* para eliminar todas las guías de la página actual.

Truco

** Ajustar a elementos de diseño existentes **

Además de guías y cuadrículas, puede utilizar elementos existentes como referencias de ajuste al mover, cambiar el tamaño o crear elementos nuevos; estos se llaman **guías inteligentes** y requieren la opción *Ver -> Guías inteligentes* para ser marcada. Cada vez que el puntero del ratón está cerca del límite de un elemento, aparece una cruz de ensamblado.

El Panel Elementos

El panel *Elementos* ofrece algunas opciones para gestionar la selección y visibilidad de los elementos. Todos los elementos añadidos al lienzo de diseño de impresión (incluido *grupo de elementos*) se muestran en una lista y la selección de un elemento hace que se seleccione la fila correspondiente en la lista, al igual que la selección de una fila hace que se seleccione el elemento correspondiente en el lienzo de composición de impresión. Se trata, por tanto, de una forma práctica de seleccionar un elemento situado detrás de otro. Tenga en cuenta que una fila seleccionada se muestra en negrita. También es posible seleccionar varios elementos manteniendo pulsada la tecla `Shift` para los elementos contiguos, y la tecla `Ctrl` para los no contiguos.

Para cualquier artículo disponible, puede:

- hacerlo visible o no;
- bloquear o desbloquear su posición;
- ordenar su posición Z. Puede mover hacia arriba y hacia abajo cada elemento de la lista con un clic y arrastrar. El elemento superior de la lista se colocará en primer plano en el lienzo de la composición de impresión. De forma predeterminada, un elemento recién creado se coloca en primer plano.
- cambie el ID del elemento haciendo doble clic en el texto;
- haga clic con el botón derecho en un elemento y seleccione si desea copiarlo o eliminarlo o abrir su *panel de propiedades*.

Una vez que haya encontrado la posición correcta para un elemento, puede bloquearlo marcando la casilla en la columna. Los elementos bloqueados **no** se pueden seleccionar en el lienzo. Los elementos bloqueados se pueden desbloquear seleccionando el elemento en el panel *Elementos* y desmarcando la casilla de verificación o puede utilizar los iconos en la barra de herramientas.

El panel de historial de deshacer: acciones de reversión y restauración

Durante el proceso de diseño, es posible revertir y restaurar los cambios. Esto se puede hacer con las herramientas de reversión y restauración disponibles en el menú *Editar*, la barra de herramientas :guilabel:` Diseño` o el menú contextual cada vez que haga clic con el botón derecho en el área de diseño de impresión:

- Deshacer último cambio
- Rehacer último cambio

Esto también puede hacerse haciendo clic con el ratón dentro del panel *Deshacer historial* (véase Figura 18.8). El panel Historial muestra las últimas acciones realizadas dentro del diseño de impresión. Seleccione el punto al que desea volver y una vez que realice una nueva acción se eliminarán todas las acciones realizadas después de la seleccionada.

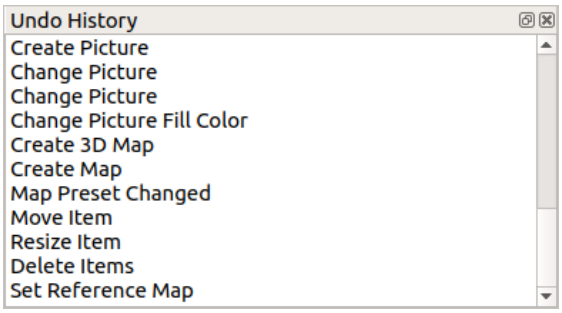


Figura 18.8: Deshacer historial en la composición de impresión

18.2 Elementos del Layout

18.2.1 Opciones de Elementos comunes de Composición

QGIS proporciona un amplio conjunto de elementos para diseñar un mapa. Pueden ser de mapa, leyenda, barra de escala, imagen, tabla, flecha norte, tipo de imagen … Sin embargo, comparten algunas opciones y comportamientos comunes que se exponen a continuación.

Creando un elemento de composición

Los elementos se pueden crear utilizando diferentes herramientas, ya sea desde cero o basándose en elementos existentes.

Para crear un elemento de diseño desde cero:

1. Seleccione la herramienta correspondiente del menú *Añadir elemento* o de la barra de la *Caja de Herramientas*.
2. Entonces:
   - Haga clic en la página y complete la información de tamaño y ubicación solicitada en el cuadro de diálogo emergente *Propiedades del nuevo elemento* (para obtener más detalles, consulte *Posición y tamaño*);

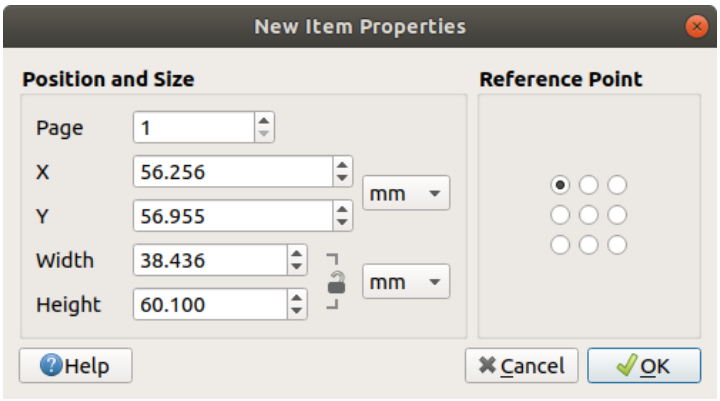


Figura 18.9: Diálogo Propiedades de Nuevo Elemento

   - O haga clic y arrastre para definir el tamaño inicial y la ubicación del elemento. Puede confiar en que las *rejillas y guías* se ajusten para una mejor posición.

> **Nota**
>
> Debido a que pueden tener formas particulares, los elementos de nodo de dibujo o flecha no funcionan con los métodos de un clic ni de hacer clic y arrastrar; debe hacer clic y colocar cada nodo del elemento. Ver *Los elementos de forma basados en nodos* para más detalles.

Además puede:

1. Seleccionar un elemento existente con el botón  Seleccionar / Mover elemento de la barra de herramientas :guilabel:` Caja de herramientas`
2. Utilice el menú contextual o la herramienta del menú *Editar* para copiar/cortar el elemento y pegarlo en la posición del mouse como un elemento nuevo.

También puede utilizar el comando *Pegar en lugar* (`Ctrl + Shift + V`) para duplicar un elemento de una página a otra y colocarlo en la nueva página en las mismas coordenadas que el original.

Además, puede crear elementos utilizando una plantilla de diseño de impresión (para obtener más información, consulte *El Administrador de Composiciones*) a través del comando :menuselection:`Diseño -> Añadir elementos desde plantilla… `.

Truco

Añadir elementos de composición usando el explorador de archivos

Desde su explorador de archivos o usando el Panel *Navegador*, arrastre y suelte una plantilla del compositor de impresión (archivo `.qpt`) en un diálogo de composición de impresión y QGIS agrega automáticamente todos los elementos de esa plantilla al diseño.

Interactuando con los elementos de composición

Cada elemento dentro del diseño de impresión se puede mover y cambiar de tamaño para crear un diseño perfecto.

Para ambas operaciones, el primer paso es activar la herramienta  Seleccionar/Mover elemento y haga clic en el elemento.

Puede seleccionar varios elementos con el botón  Seleccionar/Mover elemento: haga clic y arrastre sobre los elementos o mantenga presionado el botón `Shift` y haga clic en cada uno de los elementos que desee. Para anular la selección de un elemento, haga clic en él manteniendo presionado el botón `Shift`.

Cada vez que hay una selección, el recuento de elementos seleccionados se muestra en la barra de estado. Dentro del menú *Editar*, puede encontrar acciones para seleccionar todos los elementos, borrar todas las selecciones, invertir la selección actual y más …

Moviendo y redimensionando elementos

A menos que la opción *Vista* ► *Mostrar cuadros delimitadores* esté desmarcada, un elemento seleccionado mostrará cuadros en sus límites; al mover uno de ellos con el ratón se redimensionará el elemento en la dirección correspondiente. Mientras redimensiona, si mantiene pulsado `Mayús` se mantendrá la relación de aspecto. Si mantiene pulsado `Alt`, el elemento se redimensionará desde el centro.

Para mover un elemento de la presentación, selecciónelo con el ratón y muévalo manteniendo pulsado el botón izquierdo. Si necesita limitar los movimientos al eje horizontal o vertical, mantenga pulsada la tecla `Mayús` del teclado mientras mueve el ratón. También puede mover un elemento seleccionado utilizando las teclas `Flecha` del teclado; si el movimiento es demasiado lento, puede acelerarlo manteniendo pulsada la tecla `Mayús`. Si necesita mayor precisión, utilice las propiedades *Posición y tamaño*, o el ajuste de cuadrícula/guías como se explicó anteriormente para la creación de elementos.

Cambiar el tamaño o mover varios elementos a la vez se realiza de la misma manera que para un solo elemento. Sin embargo, QGIS proporciona algunas herramientas avanzadas para cambiar automáticamente el tamaño de una selección de elementos siguiendo diferentes reglas:

- cada altura de elemento coincide con el  más alto o el  elemento seleccionado más corto;
- cada ancho de elemento coincide con el  más ancho o el  elemento seleccionado más estrecho;
- cambia el tamaño de los elementos a cuadrados  : cada elemento se amplía para dar forma a un cuadrado.

Asimismo, hay *herramientas* disponible para organizar la posición de múltiples hitos distribuyéndolos equidistantemente:

- bordes (izquierdo, derecho, superior o inferior) de los elementos;
- centros de los elementos horizontal o verticalmente;

- espaciado entre elementos horizontal o verticalmente.

Agrupando elementos

Agrupar elementos le permite manipular un conjunto de elementos como si fuera uno solo: puede cambiar el tamaño, mover, eliminar y copiar fácilmente los elementos como un todo.

Para crear un grupo de elementos, seleccione más de uno y pulse el botón . *Grupo* del menú *Items* o de la barra de herramientas *Acciones*, o desde el menú contextual. Una fila llamada `Grupo` se añade al panel *Items* y puede bloquearse u ocultarse como cualquier otro *objeto del panel de elementos*. Los elementos agrupados **no se pueden seleccionar individualmente** en el lienzo; utilice el panel Elementos para la selección directa y acceda al panel de propiedades del elemento.

Bloqueando elementos

Una vez que haya encontrado la posición correcta para un elemento, puede bloquearlo utilizando el botón **|bloqueado|** *Bloquear elementos seleccionados* del menú *Elementos* o de la barra de herramientas *Acciones*, o marcando la casilla situada junto al elemento en el panel *Elementos*. Los elementos bloqueados **no** se pueden seleccionar en el lienzo.

Los elementos bloqueados se pueden desbloquear seleccionando el elemento en el el panel *Elementos* y desmarcando la casilla de verificación o puede utilizar los iconos en la barra de herramientas.

Alineación y distribución

Las herramientas para subir o bajar la posición Z de los elementos de la maqueta se encuentran dentro del menú desplegable Subir elementos seleccionados. Elija un elemento en el lienzo de la maqueta de impresión y seleccione la funcionalidad correspondiente para subir o bajar el elemento seleccionado sobre los demás elementos. Este orden se muestra en el panel *Elementos*. También puede subir o bajar objetos en el panel *Elementos* haciendo clic y arrastrando la etiqueta de un objeto en esta lista.

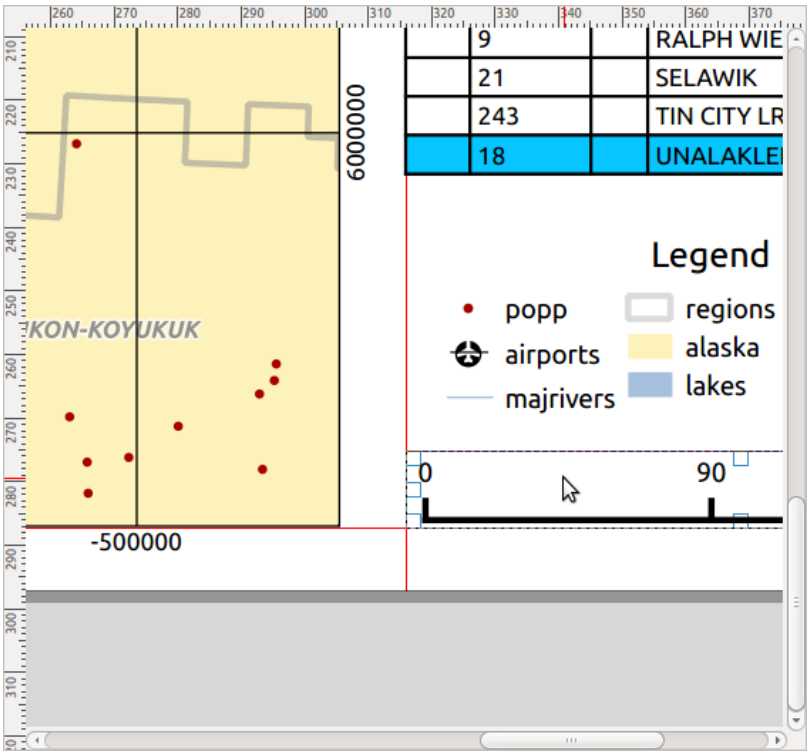


Figura 18.10: Líneas de ayuda de alineación en el diseño de impresión

Hay varias opciones de alineación disponibles dentro del menú desplegable Alinear elementos seleccionados (ver Figura 18.10). Para utilizar una función de alineación, primero seleccione los elementos y luego haga click en uno de los iconos de alineación:

- *Alinear a la ziquierda* o *Alinear a la derecha*;
- *Alinear arriba* o *Alinear abajo*;
- *Alinear al centro* horizontalmente o *Alinear al centro vertical*.

Todos los elementos seleccionados se alinearán con su cuadro delimitador común. Al mover elementos en el lienzo de diseño, aparecen líneas de ayuda de alineación cuando los bordes, los centros o las esquinas están alineados.

Otra forma de mejorar la colocación de los elementos de la composición es ajustando el espaciado entre ellos, sobre la página de la composición. Esto puede hacerse seleccionando los elementos y pulsando el menú desplegable *Distribuir bordes izquierdos* para:

- *Distribuir Bordes Izquierdos* o *Distribuir Bordes Derechos* de los elementos equidistantemente.
- *Distribuir bordes superiores* o *Distribuir bordes inferiores* de los elementos de forma equidistante.
- *Distribuir centros horizontales* o *Distribuir centros verticales* de los elementos de forma equidistante.
- Añade el mismo espacio entre los elementos: **|distribuirHSpace|** *Distribuir equitativamente el espaciado horizontal* o **|distributeVSpace||***Distribuir equitativamente el espaciado vertical*.

Propiedades comunes de elementos

Los elementos de diseño tienen un conjunto de propiedades comunes que encontrará en la parte inferior del panel *Propiedades del elemento*: Posición y tamaño, Rotación, Marco, Fondo, ID de elemento, Variables y Renderizado (véase Figura 18.11).

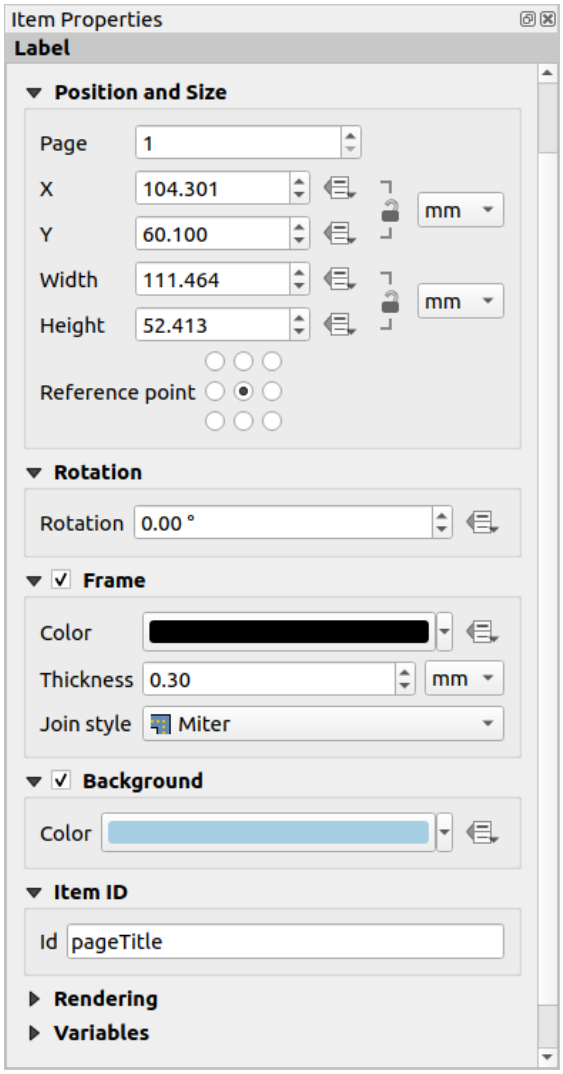


Figura 18.11: Grupos de propiedades de elementos comunes

Nota

El icono Anulación definida por datos junto a la mayoría de las opciones significa que puede asociar esa propiedad con una capa, atributos de objetos, geometría o con cualquier otra propiedad de elemento de diseño, usando expresiones `o :ref:`variables `. Para obtener más información, consulte :ref:`data_defined.

- El grupo *Posición y tamaño* te permite definir el tamaño y la posición del marco que contiene el elemento (ver *Posición y tamaño* para más información).
- La *Rotación* establece la rotación del elemento (en grados).
- El *Marco* muestra u oculta el marco alrededor del elemento. Utilice los widgets *Color*, *Espesor* y *Estilo de unión* para ajustar esas propiedades.
- En el grupo *Fondo* se puede crear o elegir un *Color* de fondo. La transparencia puede ajustarse modificando la configuración del campo alfa.
- Utilice *ID del elemento* para crear una relación con otros elementos de diseño de impresión. Esto se usa con el servidor QGIS y otros clientes web potenciales. Puede establecer una ID en un elemento (por ejemplo, un mapa o una etiqueta), y luego el cliente web puede enviar datos para establecer una propiedad (por ejemplo,

texto de etiqueta) para ese elemento específico. El comando GetProjectSettings enumerará los elementos y los ID que están disponibles en un diseño.

- El modo *Representación* le ayuda a establecer si el elemento se puede mostrar y cómo: puede, por ejemplo, aplicar *modo de fusión*, ajustar la opacidad del elemento o :guilabel:` Excluir elemento de las exportaciones` .

Posición y tamaño

Ampliando las características del diálogo *Propiedades del nuevo elemento* con capacidades definidas por datos, este grupo le permite colocar los elementos con precisión.

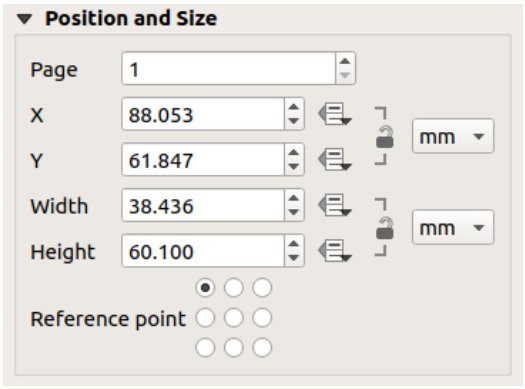


Figura 18.12: Posición y tamaño

- el número actual de la página donde ubicar el elemento;
- el punto de referencia del elemento;
- las coordenadas *X* y *Y* del *Punto de referencia* del elemento en la página elegida. La relación entre estos valores se puede bloquear haciendo clic en el botón . Los cambios realizados en un valor mediante el widget o la herramienta Seleccionar/Mover elemento se reflejará en ambos;
- el *Ancho* y *Alto* del cuadro delimitador del elemento. En cuanto a las coordenadas, la relación entre ancho y alto se puede bloquear.

Modo de representación

QGIS permite la representación avanzada de elementos de diseño como capas vectoriales y ráster.

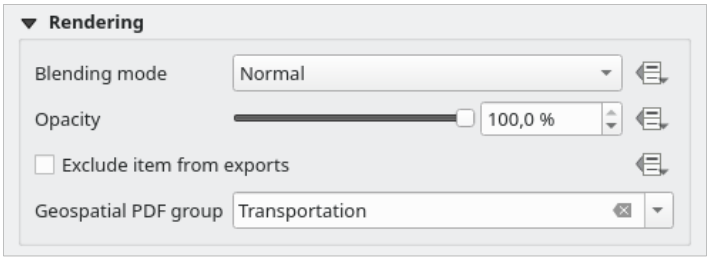


Figura 18.13: Modo de representación

- *Modo de fusión*: Con esta herramienta puede lograr efectos que de otra manera solo se lograrían usando software de renderizado gráfico. Los píxeles de sus elementos superpuestos y subyacentes se pueden mezclar de acuerdo con el modo establecido (consulte *Modos de Mezcla* para obtener una descripción de cada efecto).

- *Opacidad*: Con esta herramienta puede hacer visible el elemento subyacente en el diseño. Utilice el control deslizante para adaptar la visibilidad del elemento a sus necesidades. También puede definir con precisión el porcentaje de visibilidad en el menú situado junto al control deslizante.
- *Excluir elemento de las exportaciones*: Combinado con expresiones definidas por datos, puede decidir que un elemento sea invisible en algunas exportaciones. Tras activar esta casilla, el elemento no se incluirá en la exportación a PDF, impresión, etc.
- *Grupo de PDF geoespacial*: Cuando se activa, se creará un grupo de árbol de capas coincidente en el PDF geoespacial exportado y el elemento solo será visible cuando se marque este grupo. Esto permite que los espectadores del PDF geoespacial puedan mostrar el contenido de forma selectiva como un grupo. Por ejemplo, puede permitir que el contenido adicional del diseño, como etiquetas descriptivas o leyendas, solo se muestre cuando las capas del grupo estén visibles, lo que hace que la exportación del archivo sea mucho más flexible.

Variables

La sección *Variables* enumera todas las variables disponibles a nivel del elemento de diseño (que incluye todas las variables globales, de proyecto y de diseño). Los elementos de diseño de mapa tienen una sección *Configuración del mapa* adicional para variables que facilitan el acceso a valores como la escala del mapa, su extensión, etc.

En *Variables*, también es posible gestionar variables a nivel de elemento de diseño. Haga clic en el botón para añadir una nueva variable personalizada. Del mismo modo, seleccione cualquier variable personalizada a nivel de elemento de la lista y haga clic en el botón para eliminarla.

Más información sobre el uso de variables en la sección *Almacenando valores en variables*.

18.2.2 El elemento del mapa

El elemento del mapa es el marco principal que muestra el mapa que ha diseñado en el lienzo del mapa. Utilice la herramienta *Añade nuevo mapa a la composición* como sigue en *instrucciones de creación de elementos* para agregar un nuevo elemento de mapa que luego puede manipular de la misma manera que se expone en *Interactuando con los elementos de composición*.

De forma predeterminada, un nuevo elemento de mapa muestra el estado actual de *vista del mapa* con su extensión y capas visibles. Puede personalizarlo gracias al panel *Propiedades del elemento*. Aparte de *propiedades comunes de los elementos*, este objeto tiene las siguientes funcionalidades:

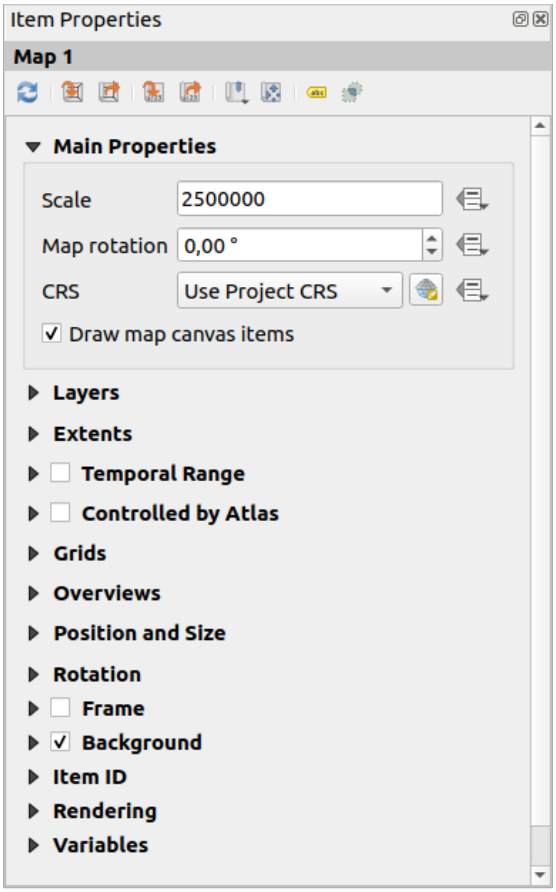


Figura 18.14: Panel de Propiedades del Elemento Mapa

La Barra de Herramientas

El panel Mapa *Propiedades del elemento* incluye una barra de herramientas con las siguientes funcionalidades:

- Update map preview
- Set map canvas to match main canvas extent
- View current map extent in main canvas
- Set map scale to match main canvas scale
- Set main canvas to match current map scale
- Marcadores: establecer la extensión del elemento del mapa para que coincida con un marcador espacial existente
- Interactively edit map extent: desplazarse y hacer zoom de forma interactiva dentro del elemento del mapa
- Configuración de etiquetado: controla el comportamiento de la etiqueta de entidad (ubicación, visibilidad …) en la extensión del elemento del mapa de diseño:
  - establecer un *Margen desde los bordes del mapa*, una distancia de datos definible desde los límites del elemento del mapa dentro de la cual no se debe mostrar ninguna etiqueta
  - *Permitir etiquetas truncadas en los bordes del mapa*: controla si las etiquetas que caen parcialmente fuera de la extensión permitida del elemento del mapa deben ser renderizadas. Si está marcada, estas

etiquetas se mostrarán (cuando no haya forma de colocarlas completamente dentro del área visible). Si no se marca, se omitirán las etiquetas parcialmente visibles.

- *Elementos de bloqueo de etiquetas*: permite que otros elementos de diseño (como barras de escala, flechas de norte, mapas insertados, etc.) se marquen como bloqueadores para las etiquetas del mapa en el elemento de mapa **activo**. Esto evita que se coloquen etiquetas de mapa debajo de esos elementos, lo que hace que el motor de etiquetado pruebe una ubicación alternativa para estas etiquetas o las descarte por completo.

  Si Se establece *Margen desde los bordes del mapa*, las etiquetas del mapa no se colocan más cerca de la distancia especificada de los elementos de diseño marcados.

- *Mostrar etiquetas no colocadas*: se puede usar para determinar si faltan etiquetas en el mapa de diseño (por ejemplo, debido a conflictos con otras etiquetas de mapa o debido a espacio insuficiente para colocar la etiqueta) resaltándolas en a *color predefinido*.

- Configuración de recorte: permite recortar el elemento de mapa al objeto del atlas y dar forma a elementos poligonales:

  - *Recortar a objeto atlas*: puede determinar que el elemento del mapa de diseño se recortará automáticamente al actual *objeto atlas*.

    Hay diferentes modos de recorte disponnibles:

    * *Cortar solo durante el renderizado*: aplica un clip basado en pintor, de modo que las partes de las entidades vectoriales que se encuentran fuera del objeto del atlas se vuelven invisibles
    * *Cortar Objeto Antes de Renderizar*: aplica el corte antes de renderizar las entidades, por lo que los bordes de las entidades que quedan parcialmente fuera de la entidad de atlas seguirán siendo visibles en el límite de la entidad de atlas
    * *Representar Objetos Intersecantes Sin Cambios*: renderiza todas las entidades que se cruzan con la entidad del atlas actual, pero sin recortar su geometría.

    Puede *Forzar etiquetas dentro del atlas*. Si no desea *Recortar todas las capas* a la característica atlas puede utilizar la opción *Recortar las capas seleccionadas*.

  - 8|checkbox| *Cortar a elemento*: es posible cambiar la forma del elemento del mapa utilizando un elemento de *forma* o *polígono* del diseño de impresión. Cuando habilita esta opción, el mapa se recortará automáticamente a la forma seleccionada en el cuadro combinado. Nuevamente, los modos de recorte mencionados anteriormente están disponibles y las etiquetas pueden verse forzadas a mostrarse solo dentro de la forma de recorte.

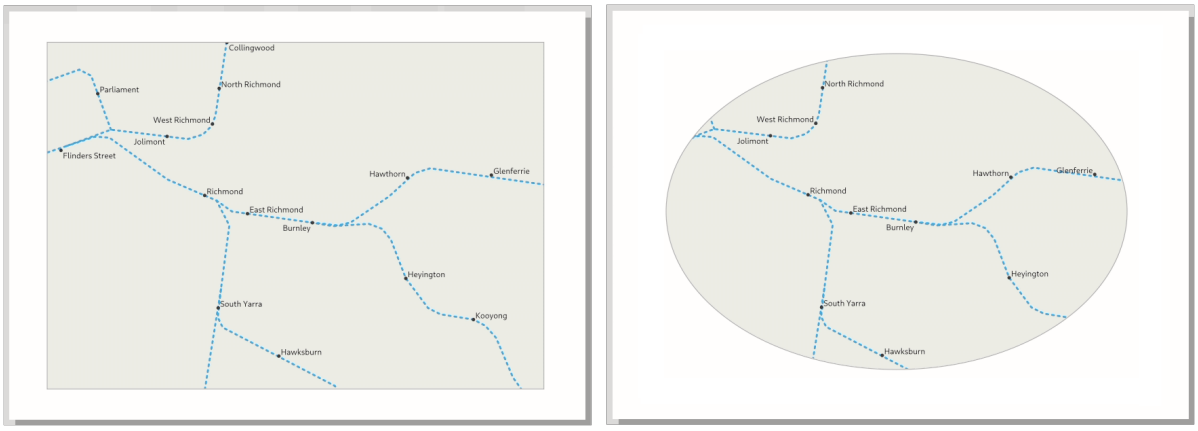


Figura 18.15: Recortar un elemento de mapa de diseño de formas

Propiedades principales

En el grupo *propiedades principales* (ver Figura 18.14) del panel de mapa *Propiedades del Elemento*, estan disponibles las opciones:

- El botón *Update Map Preview* para actualizar la representación del elemento del mapa si se ha modificado la vista en el lienzo del mapa. Tenga en cuenta que la mayoría de las veces, los cambios activan automáticamente la actualización del elemento del mapa;
- La *Escala* para configurar manualmente la escala del elemento del mapa;
- La *Rotación del mapa* le permite rotar el contenido del elemento del mapa en el sentido de las agujas del reloj en grados. Aquí se puede imitar la rotación del lienzo del mapa;
- El *SRC* le permite mostrar el contenido del elemento del mapa en cualquier SRC. Por defecto es `Usar SRC del proyecto`;
- *Dibujar elementos del lienzo del mapa* le permite mostrar en el diseño de impresión *anotaciones* que se colocan en el lienzo del mapa principal.

Capas

De forma predeterminada, la apariencia del elemento del mapa se sincroniza con la representación del lienzo del mapa, lo que significa que alternar la visibilidad de las capas o modificar su estilo en *Panel de capas* se aplica automáticamente al elemento del mapa. Debido a que, como cualquier otro elemento, es posible que desee agregar varios elementos de mapa a un diseño de impresión, es necesario romper esta sincronización para permitir mostrar diferentes áreas, combinaciones de capas, a diferentes escalas … El grupo de propiedades de *Capas* (ver Figura 18.16) le ayuda a hacer eso.

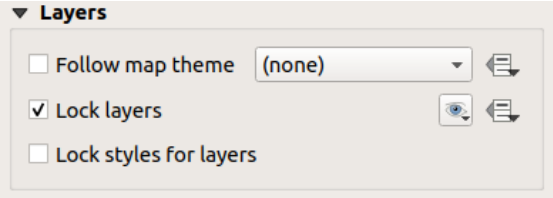


Figura 18.16: Grupo Capas del mapa

Si desea mantener el elemento del mapa consistente con un *tema de mapa* existente , marque *Seguir el tema del mapa* y seleccionar el tema deseado en la lista desplegable. Cualquier cambio aplicado al tema en la ventana principal de QGIS (usando la función de reemplazo de tema) afectará automáticamente al elemento del mapa. Si se selecciona un tema de mapa, la opción *Bloquear estilos para capas* está deshabilitada porque :guilabel:` Seguir tema de mapa` también actualiza el estilo (simbología, etiquetas, diagramas) de las capas.

Para bloquear las capas que se muestran en un elemento de mapa a la visibilidad del lienzo del mapa actual, marque *Bloquear capas*. Cuando esta opción está habilitada, cualquier cambio en la visibilidad de las capas en la ventana principal de QGIS no afectará el elemento del mapa del diseño. Sin embargo, el estilo y las etiquetas de las capas bloqueadas todavía se actualizan de acuerdo con la ventana principal de QGIS. Puede evitar esto usando *Bloquear estilos para capas*.

En lugar de utilizar el lienzo del mapa actual, también puede bloquear las capas del elemento del mapa a las de un tema de mapa existente: seleccione un tema de mapa en el botón desplegable Establecer lista de capas a partir de un tema de mapa, y *Bloquear capas* está activado. El conjunto de capas visibles en el tema del mapa se utiliza a partir de ahora para el elemento del mapa hasta que seleccione otro tema del mapa o desmarque la opción *Bloquear capas*. A continuación, es posible que deba actualizar la vista con el botón Actualizar vista de mapa de la barra de herramientas *Navegación* o el botón *Actualizar vista previa* visto arriba.

Tenga en cuenta que, a diferencia de la opción *Seguir tema del mapa*, si la opción :guilabel:` Bloquear capas` está habilitada y configurada en un tema de mapa, las capas del elemento del mapa no se actualizarán incluso si el tema del mapa es actualizado (usando la función reemplazar tema) en la ventana principal de QGIS.

Las capas bloqueadas en el elemento del mapa también pueden ser *data-defined*, usando el icono junto a la opción. Cuando se usa, esto anula la selección establecida en la lista desplegable. Debe pasar una lista de capas separadas por el carácter `|`. El siguiente ejemplo bloquea el elemento del mapa para usar solo las capas `capa 1` y `capa 2`:

```
concat ('layer 1', '|', 'layer 2')
```

Extensión

El grupo *Extensión* del panel de propiedades del elemento de mapa proporciona las siguientes funcionalidades (ver Figura 18.17):

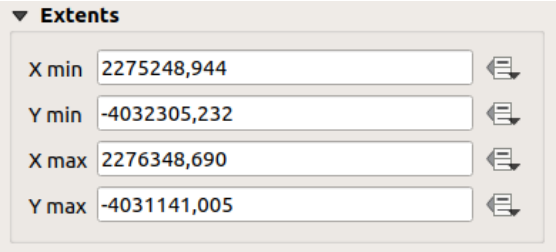


Figura 18.17: Grupo Extensión del mapa

El área **Extensiones** muestra las coordenadas `X` e `Y` del área que se muestra en el elemento del mapa. Cada uno de estos valores se puede reemplazar manualmente, modificando el área del lienzo del mapa que se muestra y/o el tamaño del elemento del mapa. La extensión también se puede modificar utilizando herramientas en la parte superior del panel de elementos del mapa, como:

- Set map canvas to match main canvas extent
- Set map scale to match main canvas scale

También puede modificar la extensión de un elemento de mapa mediante la herramienta Mover contenido del elemento: haga clic y arrastre dentro del elemento del mapa para modificar su vista actual, manteniendo la misma escala. Con la herramienta habilitada, use la rueda del mouse para acercar o alejar, modificando la escala del mapa mostrado. Combine el movimiento con la tecla `Ctrl` presionada para tener un zoom más pequeño.

Rango de elevación

La configuración *Rango de elevación* en las propiedades del elemento del mapa le permite filtrar el contenido de determinadas capas en función de un rango de elevación específico. Aunque todas las capas permanecen visibles, aquellas que admiten el filtrado por elevación (actualmente nubes de puntos y DEM ráster) verán filtrados sus datos. Solo se mostrarán las partes de estas capas que se encuentren dentro del rango de elevación establecido por los valores *Inferior* y *Superior*.

El *Rango de elevación* puede definirse mediante datos. Esto significa que puede establecer diferentes rangos de elevación para diferentes características en el *atlas* o *informe*.

Rango temporal

El grupo *Temporal range* del panel de propiedades del elemento del mapa proporciona las opciones para controlar la representación de capas en el elemento del mapa en función de un rango temporal. Solo las capas cuyas propiedades temporales se superponen con el rango de tiempo establecido por las fechas *Start* y *End* se muestran en el elemento del mapa.

Los widgets asociados definidos por datos ayudan a que el rango de tiempo sea dinámico y permiten la salida de atlas temporales *atlases*, es decir, mapas automatizados con una extensión espacial fija y cuyo contenido varía según el tiempo. Por ejemplo, si usa como capa de cobertura un archivo csv con un par de campos de inicio y final y una cantidad de filas que representan rangos de fechas, habilite tanto el rango temporal como el control por atlas en las propiedades del elemento del mapa y presione la exportación de atlas.

Controlado por Atlas

Las propiedades del grupo *Controlado por atlas* está disponible solo si un :ref:`atlas<atlas_generation>`está activo en el diseño de impresión. Marque esta opción si desea que el elemento del mapa sea gobernado por el atlas; al iterar sobre la capa de cobertura, la extensión del elemento del mapa se panoramiza / acerca a la función del atlas siguiente:

- *Margen alrededor de entidades*: acerca la entidad a la mejor escala, manteniendo alrededor de cada uno un margen que representa un porcentaje del ancho o alto del elemento del mapa. El margen puede ser el mismo para todos los objetos o *set variable*, por ejemplo, dependiendo de la escala del mapa;
- *Escala predefinida (mejor ajuste)*: acerca la función en el proyecto *escala predefinida* donde la función del atlas encaja mejor;
- *Escala fija*: las entidades del atlas se desplazan de una a otra, manteniendo la misma escala del elemento del mapa. Ideal cuando se trabaja con objetos del mismo tamaño (por ejemplo, una cuadrícula) o cuando se desea resaltar las diferencias de tamaño entre las características del atlas.

Cuadrículas

Con las cuadrículas, puede agregar, sobre su mapa, información relativa a su extensión o coordenadas, ya sea en la proyección del elemento del mapa o en una diferente. El grupo *Cuadrículas* brinda la posibilidad de agregar varias cuadrículas a un elemento del mapa.

- Con los botones y puede agregar o eliminar una cuadrícula seleccionada;
- Con los botones y puede mover hacia arriba y hacia abajo en una cuadrícula en la lista, por lo tanto, moverla arriba o abajo de otra, sobre el elemento del mapa.

Haga doble clic en la cuadrícula agregada para cambiarle el nombre.

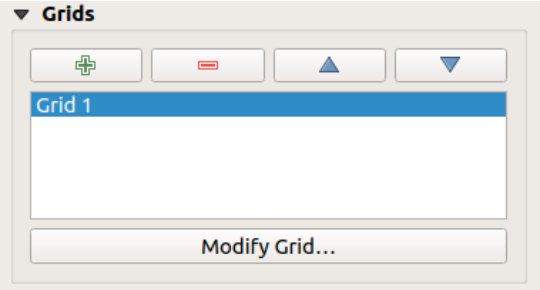


Figura 18.18: Diálogo de cuadrículas del mapa

Para modificar una cuadrícula, selecciónela y presione el botón *Modificar cuadrícula…* para abrir el panel *Propiedades de cuadrícula de mapa* y acceder a sus opciones de configuración.

Apariencia de cuadrícula

En el panel *Propiedades de la cuadrícula de mapa*, marque *Cuadrícula habilitada* para mostrar la cuadrícula en el elemento del mapa.

Como tipo de cuadrícula, puede especificar usar un:

- **Sólido**: muestra una línea a través del marco de la cuadrícula. El *Estilo de línea* puede personalizarse utilizando *color* y *símbolo* widget selector;
- **Cruz**: muestra el segmento en la intersección de las líneas de la rejilla para el que puede establecer el *Estilo de línea* y el *Ancho de cruz*;
- **Marcadores**: sólo muestra el símbolo de los marcadores personalizables en la intersección de las líneas de la cuadrícula;
- o **Sólo marco y anotaciones**.

Aparte del tipo de cuadrícula, puede definir:

- el *SRC* de la cuadrícula: por defecto, seguirá el SRC del elemento del mapa. Pulse el botón Seleccionar SRC para establecer un SRC diferente.
- el tipo *Intervalo* a utilizar para las referencias de cuadrícula:
  - **Unidades del mapa**: se establece una distancia dentro del mapa (en la unidad de la cuadrícula SRC) entre referencias de cuadrícula consecutivas en las direcciones *X* y *Y*. El número de marcas de cuadrícula variará en función de la escala del mapa.
  - Al elegir **Ajustar anchura de segmento** se seleccionará dinámicamente el intervalo de la cuadrícula en función de la extensión del mapa a un intervalo «bonito». Ese intervalo óptimo se calcula dentro de un rango de distancias cuyos valores *Mínimo* y *Máximo* pueden personalizarse.
  - Con **Milímetros** o **Centímetros**, se establece una distancia en el papel entre referencias de cuadrícula consecutivas en las direcciones *X* y *Y*. El número de marcas de cuadrícula será el mismo sea cual sea la escala del mapa.
- el *Desplazamiento* desde los bordes del elemento del mapa, en la dirección *X* y/o en la dirección *Y*
- y el *modo de mezcla* de la cuadrícula (ver modos de mezcla) cuando sea compatible.

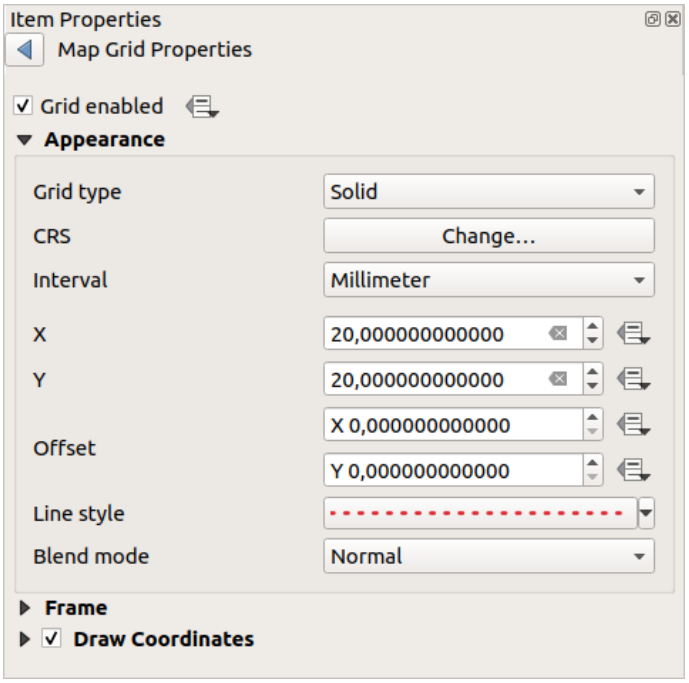


Figura 18.19: Diálogo Apariencia de Cuadrícula

Marco Cuadrícula

Hay diferentes opciones para diseñar el marco que contiene el mapa. Están disponibles las siguientes opciones: `Sin marco`, `` Cebra``, `` Zebra (náutica)``, `Marcas interiores`, `Marcas exteriores`, `Marcas interiores y exteriores`, `Borde de línea` y `Borde de línea (náutico)`.

Cuando sea compatible, es posible establecer *Tamaño del marco*, *Margen del marco*, *Grosor de la línea del marco* con el color asociado y *Colores de relleno del marco*.

Usando los valores de `Latitud/Solo Y` y `Longitud / Solo X` en la sección de divisiones, puede evitar que se muestre una combinación de coordenadas de latitud/Y y longitud/X en cada lado cuando se trabaja con mapas rotados o cuadrículas reproyectadas. También puede optar por establecer visibles o no cada lado del marco de la cuadrícula.

Cuando se gira la extensión del elemento del mapa (desde el grupo *Propiedades principales*) o se aplica un SRC diferente a la cuadrícula, es posible que las líneas de la cuadrícula no sean ortogonales a los lados del elemento del mapa. Esto puede dar lugar a un mal aspecto de la rejilla cuando se aplica estilo con ticks interiores y/o exteriores. Activando *Seguir la rotación de la rejilla* alineará las marcas con las líneas de la rejilla. Además, puede ajustar algunas propiedades más:

- *Alineación de los ticks*: Los ticks interiores y/o exteriores serán paralelos a su correspondiente línea de rejilla. Su alineación puede ser:
  - **Ortogonales**: los ticks del mismo lado terminan en una línea, paralelos al lado. Esto puede provocar, por ejemplo, que algunos ticks se alarguen cuando tienen un ángulo bajo respecto al cuadro.
  - **Longitud fija**: todas las marcas tienen la misma longitud, por lo que es posible que no se alineen.
- *Omitir por debajo del ángulo*: evita que se muestren las marcas de las líneas de la cuadrícula que se cruzan con el borde del marco por debajo de un umbral especificado.
- *Margen desde la esquina del mapa*: evita mostrar ticks demasiado cerca de las esquinas del mapa, ya que podrían solaparse y/o estar fuera de los límites.

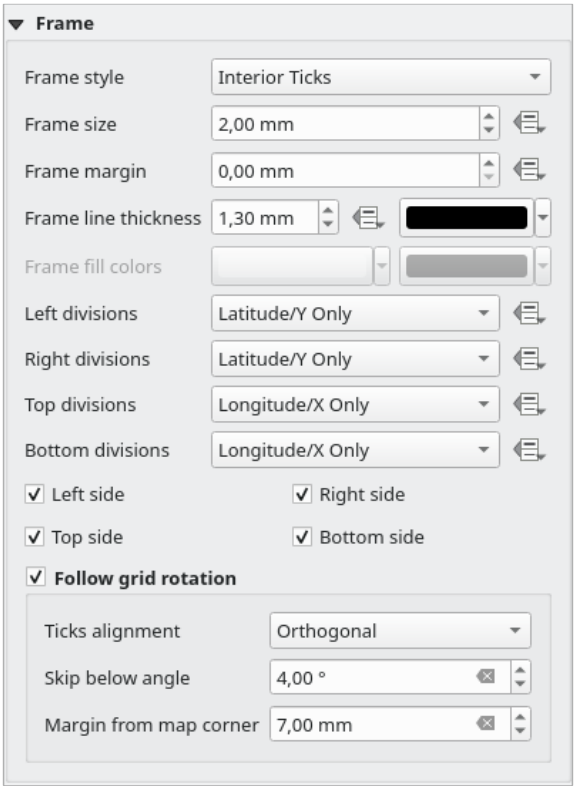


Figura 18.20: Diálogo de marco de cuadrícula

Coordenadas

La casilla *Dibujar coordenadas* permite añadir coordenadas al marco del mapa. Los valores mostrados se refieren a la unidad de *intervalo de cuadrícula* elegida. Puede elegir el formato numérico de la anotación, las opciones van desde decimal a grados, minutos y segundos, con o sin sufijo, alineado o no y un formato personalizado utilizando el diálogo de expresión.

Para cada uno de los lados *Izquierda*, *Derecha*, *Superior* e *Inferior* del marco de la cuadrícula, puede indicar:

- si se muestran las coordenadas: **Mostrar todo**, **Mostrar sólo latitud/Y**, **Mostrar sólo longitud/X**, **Desactivado**. Mostrar sólo los valores de Latitud/Y o Longitud/X en las divisiones ayuda a prevenir una mezcla de coordenadas de latitud/Y y longitud/X mostradas en cada lado cuando se trabaja con mapas rotados o cuadrículas reproyectadas.
- la posición relativa del texto respecto al marco de la cuadrícula: **Marco exterior** o **Marco interior**.
- la colocación y orientación de la anotación:
  - **Horizontal**
  - **Vertical ascendente**, **Vertical descendente**
  - **Dirección de los límites**
  - **Por encima del tick**, Sobre el tick**, **Por debajo del tick** cuando se utiliza un marco basado en el tick.

También puede definir las propiedades de la :guilabel:`Fuente <text_format>` (fuente, tamaño, color, búfer,…) la *Distancia al marco del mapa* y la *Precisión de las coordenadas* (número de decimales) para las anotaciones dibujadas.

Seguir la rotación de la rejilla: disponible cuando se rota la extensión del mapa o se reproyecta la cuadrícula, le ayuda a ajustar la colocación de las anotaciones. Dependiendo del modo de colocación seleccionado, las anotaciones también se rotan:

- *Alineación de anotaciones*: puede ser **Ortogonal** o de **Longitud fija**.

- *Omitir por debajo del ángulo*: impide mostrar anotaciones para las líneas de cuadrícula que intersecan el borde del marco por debajo de un umbral especificado
- *Margen desde la esquina del mapa*: impide mostrar anotaciones demasiado cerca de las esquinas del mapa, ya que podrían solaparse y/o estar fuera de los límites.

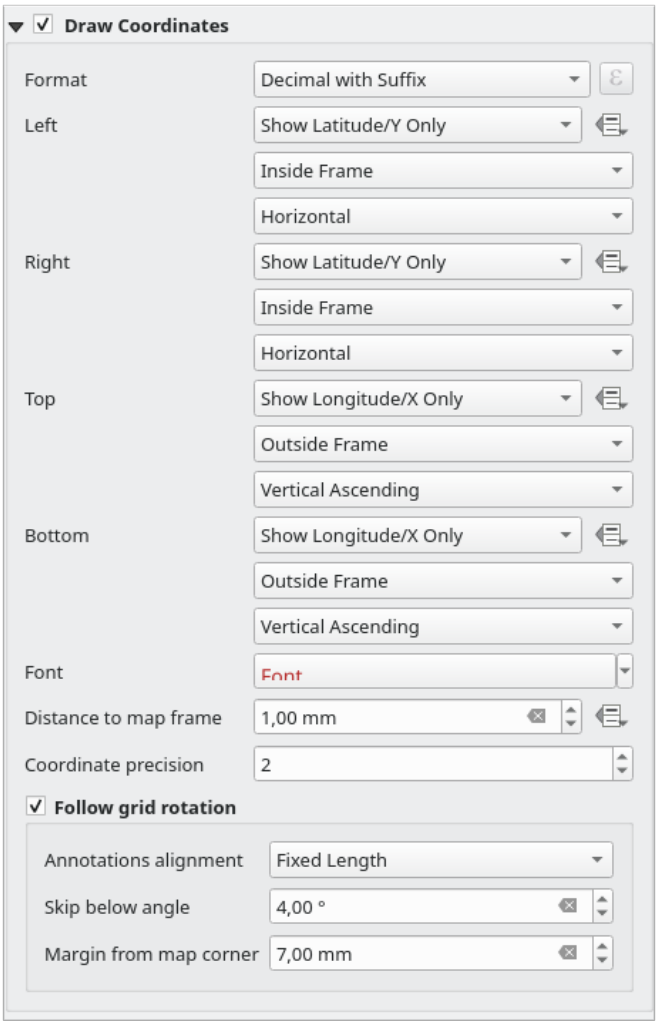


Figura 18.21: Diálogo Coordenadas para dibujo de rejilla

Vistas generales

A veces puede tener más de un mapa en el diseño de impresión y le gustaría ubicar el área de estudio de un elemento del mapa en otro. Esto podría ser, por ejemplo, para ayudar a los lectores de mapas a identificar el área en relación con su contexto geográfico más amplio que se muestra en el segundo mapa.

El grupo *Vistas generales* del panel de mapa le ayuda a crear el vínculo entre dos extensiones de mapas diferentes y proporciona las siguientes funcionalidades:

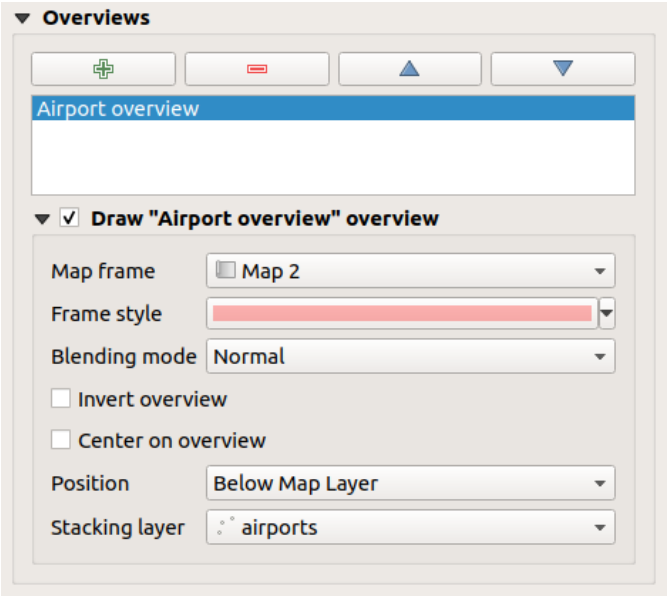


Figura 18.22: Grupo vista general del mapa

Para crear una descripción general, seleccione el elemento del mapa en el que desea mostrar la extensión del otro elemento del mapa y expanda la opción *Vistas generales* en el panel *Propiedades del elemento*. Luego presione el botón para agregar una vista general.

Inicialmente esta vista rápida es llamada “Overview 1” (ver Figura 18.22). Puede:

- Cambiar el nombre con un doble clic
- Con los botones y , agregar o eliminar vistas generales
- Con los botones y , mueven una vista general hacia arriba y hacia abajo en la lista, colocándola encima o debajo de otras vistas generales en el elemento del mapa (cuando están en la misma *posición en la pila*).

Luego, seleccione el elemento de descripción general en la lista y marque *Dibujar « <name_overview>» vista general* para habilitar el dibujo de la vista general en el marco del mapa seleccionado. Puedes personalizarlo con:

- El *marco de mapa* selecciona el elemento del mapa cuyas extensiones se mostrarán en el elemento del mapa actual.
- El *Estilo de marco* usa :ref:`propiedades del símbolo <symbol-selector>`para representar el marco de vista general.
- El *Modo de mezcla* le permite establecer diferentes modos de fusión de transparencia.
- El *Invertir vista general* crea una máscara alrededor de las extensiones cuando se activa: las extensiones del mapa referenciadas se muestran claramente, mientras que el resto del elemento del mapa se mezcla con el color de relleno del marco (si se usa un color de relleno).
- El *Centrar en vista general* desplaza el contenido del elemento del mapa para que el marco de vista general se muestre en el centro del mapa. Solo puede utilizar un artículo de resumen para centrar cuando tenga varios resúmenes.
- La *Posición* controla exactamente en qué lugar de la pila de capas del elemento del mapa se colocará la vista general, p. Ej. permitiendo dibujar una extensión de vista general debajo de algunas capas de entidades, como carreteras, mientras se dibuja sobre otras capas de fondo. Las opciones disponibles son:
  - *Debajo del mapa*

- *Debajo de la capa del mapa* y :guilabel:` Encima de la capa del mapa`: coloque el marco de vista general debajo y encima de las geometrías de una capa, respectivamente. La capa se selecciona en la opción *Capa de apilamiento*.
- *Debajo de las etiquetas del mapa*: dado que las etiquetas siempre se representan sobre todas las geometrías de entidades en un elemento de mapa, coloca el marco de vista general sobre todas las geometrías y debajo de cualquier etiqueta.
- *Encima de las etiquetas del mapa*: coloca el marco de vista general sobre todas las geometrías y etiquetas en el elemento del mapa.

18.2.3 El Elemento Mapa 3D

El elemento Mapa 3D se utiliza para mostrar una *Vista de mapa 3D*. Utilice el botón  *Añadir nuevo mapa 3D a la composición*, y siga :ref:` instrucciones de creación de elementos <create_layout_item>` para agregar un nuevo elemento de mapa 3D que luego puede manipular de la misma manera que se muestra en *Interactuando con los elementos de composición*.

De forma predeterminada, un nuevo elemento de mapa 3D está vacío. Puede establecer las propiedades de la vista 3D y personalizarla en el panel *Propiedades del elemento*. Además de *propiedades generales*, este objeto tiene las siguientes funcionalidades (Figura 18.23):

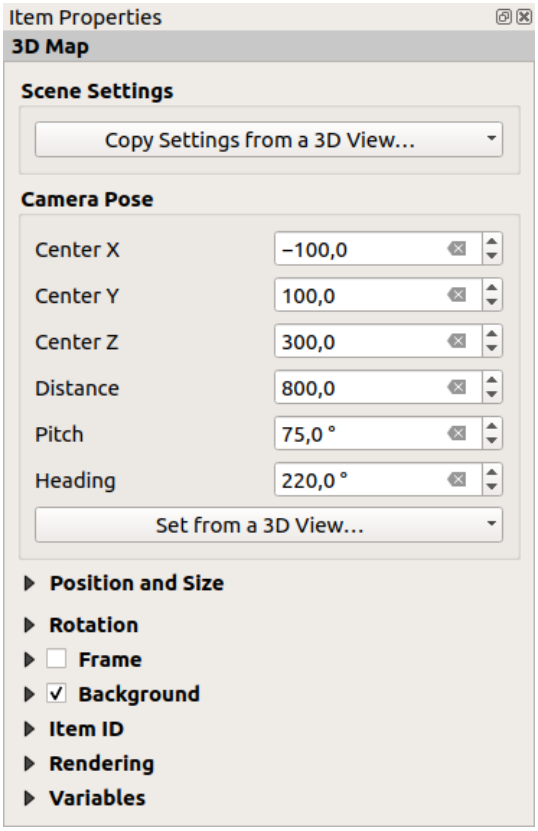


Figura 18.23: Propiedades del elemento mapa 3D

Configuración de escena

Presione *Copiar configuraciones desde una vista 3D…* para elegir la vista de mapa 3D que se mostrará.

La vista del mapa 3D se renderiza con su configuración actual (capas, terreno, luces, posición y ángulo de la cámara …).

Camera pose

- *Centro X* establece la coordenada X del punto al que apunta la cámara
- *Centro Y* establece la coordenada Y del punto al que apunta la cámara
- *Centro Z* establece la coordenada Z del punto al que apunta la cámara
- *Distancia* establece la distancia desde el centro de la cámara hasta el punto al que apunta la cámara
- *Inclinación* establece la rotación de la cámara alrededor del eje X (rotación vertical). Valores de 0 a 360(grados). 0 °: terreno visto directamente desde arriba; 90 °: horizontal (desde el lateral); 180 °: recto desde abajo; 270 °: horizontal, boca abajo; 360 °: recto desde arriba.
- *Encabezado* establece la rotación de la cámara alrededor del eje Y (rotación horizontal: 0 a 360 grados). 0 °/ 360 °: norte; 90 °: oeste; 180 °: sur; 270 °: este.

El menú desplegable *Set from a 3D View…* le permite completar los elementos con los parámetros de una vista 3D.

18.2.4 El elemento etiqueta

El elemento *Etiqueta* es una herramienta que ayuda a decorar tu mapa con textos que ayudarían a entenderlo; puede ser el título, autor, fuentes de datos o cualquier otra información … Puede agregar una etiqueta con la herramienta  *Agregar etiqueta* siguiendo las *instrucciones de creación de elementos* y manipularla de la misma manera que se expone en *Interactuando con los elementos de composición*

De forma predeterminada, el elemento de etiqueta proporciona un texto predeterminado que puede personalizar utilizando su panel *Propiedades del elemento*. Aparte de las propiedades comunes de *elementos*, esta prestación tiene las siguientes funcionalidades (ver Figura 18.24):

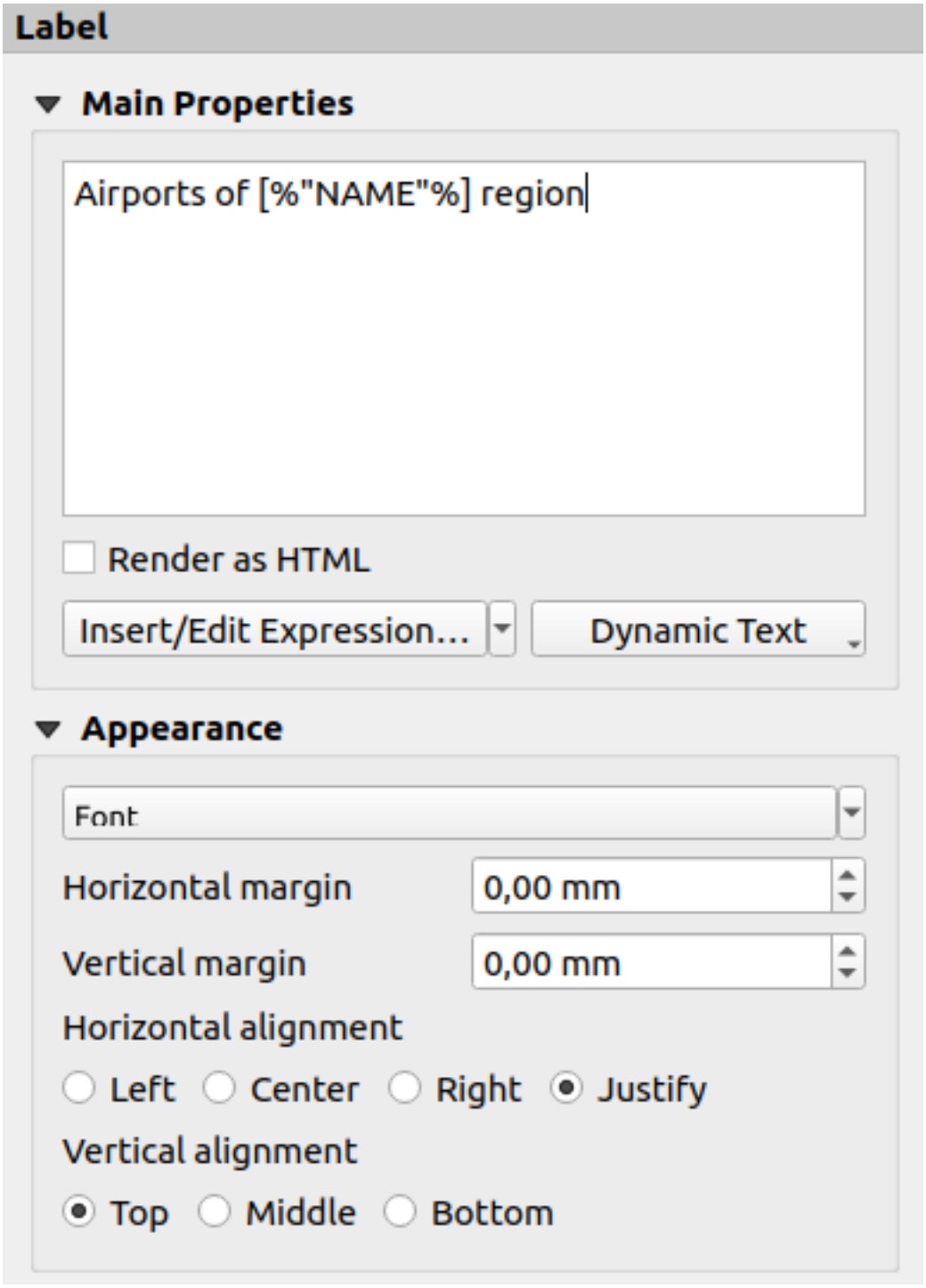


Figura 18.24: Panel de propiedades de elemento etiqueta

Propiedades principales

El grupo *Propiedades principales* es el lugar para proporcionar el texto de la etiqueta. El texto puede ser estático, dinámico con funciones y variables expression <expression_builder> ` y/o formateado con HTML. Las partes dinámicas de una etiqueta deben estar rodeadas por ``[%` y `%]` para que se interpreten y evalúen como tales.

- Para usar expresiones en las etiquetas, puede hacer clic en el botón *Insertar/Editar expresión…*, escribir su fórmula como de costumbre y cuando se aplica el cuadro de diálogo, QGIS agrega automáticamente los caracteres circundantes.

  > **Consejo**
  >
  > Cuando haces clic en el botón *Insertar/Editar expresión…* sin haber seleccionado nada en el cuadro de texto, la nueva expresión se añadirá al texto existente. Sin embargo, si desea modificar una expresión existente, el comportamiento cambia en función de su selección: si se selecciona algún texto y la selección se encuentra dentro de una expresión (entre "[%" y "%]"), o si no se selecciona ningún texto pero el cursor se encuentra dentro de una expresión, se seleccionará toda la expresión. En todos los demás casos, la selección permanecerá tal cual.

  Debido a que los mapas suelen estar llenos de alguna información textual común (fecha, autor, título, número de página, …), QGIS proporciona un acceso directo a las expresiones o variables correspondientes: presione el botón *Texto dinámico* para seleccionar y insértelos en su etiqueta.

  > **Truco**
  >
  > El menú superior *Agregar elemento* ► *Agregar texto dinámico* ► se puede usar para crear un nuevo elemento de etiqueta con la expresión predefinida seleccionada.

  Es posible convertir una etiqueta dinámica en estática: presione la flecha desplegable junto al botón *Insertar/Editar expresión…* y seleccione *Convertir a estática*. Cualquier parte dinámica del contenido de la etiqueta será evaluada y reemplazada con sus valores actuales. Luego puede modificar manualmente el texto resultante cuando sea necesario.

- Las etiquetas se pueden interpretar como código HTML: marque **|casilla de verificación|** *Renderizar como HTML*. Ahora puede insertar etiquetas o estilos HTML, URL, una imagen en la que se puede hacer clic que se vincula a una página web o algo más complejo…

El siguiente código combina la representación HTML con expresiones, para un etiquetado avanzado y generará Figura 18.25:

```
<html>
 <head>
   <style>
      /* Define some custom styles, with attribute-based size */
      name {color:red; font-size: [% ID %]px; font-family: Verdana; text-shadow:␣
↪grey 1px 0 10px;}
      use {color:blue;}
   </style>
 </head>

 <body>
   <!-- Information to display -->
   <u>Feature Information</u>
   <ul style="list-style-type:disc">
     <li>Feature Id: [% ID %]</li>
     <li>Airport: <name>[% NAME %]</name></li>
     <li>Main use: <use>[% USE %]</use></li>
   </ul>
```

(continúe en la próxima página)

(proviene de la página anterior)

```
   Last check: [% concat( format_date( "control_date", 'yyyy-MM-dd'), ' by <b><i>',
↪ @user_full_name, '</i></b>' ) %]

   <!-- Insert an image -->
   <p align=center><img src="path/to/logos/qgis-logo-made-with-color.svg" alt=
↪"QGIS icon" style="width:80px;height:50px;"</p>
 </body>
</html>
```



Figura 18.25: Aprovechando una etiqueta con estilo HTML

Apariencia

- Defina la fuente y el estilo del texto haciendo clic en el botón *Font*. En el menú *Fuente de la etiqueta* puede utilizar algunas de las opciones para *Formatear el texto de la etiqueta*.
- Puede especificar diferentes márgenes horizontales y verticales en mm. Este es el margen desde el borde del elemento de diseño. La etiqueta se puede colocar fuera de los límites de la etiqueta, p. Ej. para alinear los elementos de la etiqueta con otros elementos. En este caso, debe utilizar valores negativos para el margen.
- Usar la alineación del texto es otra forma de colocar su etiqueta. Puede ser:
  - *Izquierda*, *Centro*, *Derecha* o *Justificar* para *Alineación horizontal*
  - y *Arriba*, *Centro vertical*, *Abajo* para *Alineación vertical*.

Truco

Resize the layout label item to fit its contents

By default, the size of a layout label item does not depend on the contained text. It may be too large with useless blank space or too small, hiding part of the text. To automatically resize a label so it fits the text inside it perfectly, double-click one of the square handles on the label. The label will adjust its size to match the text, either expanding or shrinking as needed, while the opposite handle of the label remains fixed in place. This allows you to quickly ensure the label fits the text without manually resizing it. So e.g., double-clicking the left-middle handle causes the left side of the label to move.

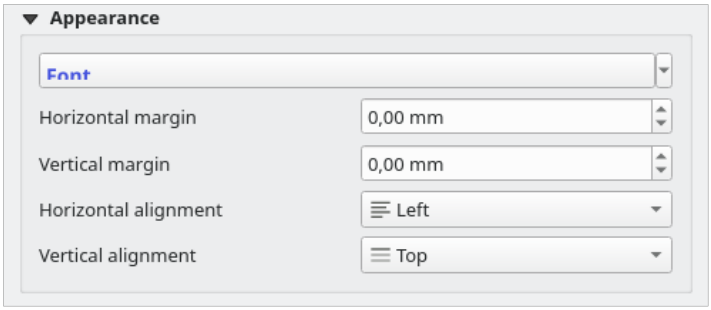


Figura 18.26: Propiedades de apariencia del elemento de etiqueta

Explorando expresiones en un elemento de etiqueta

A continuación, se muestran algunos ejemplos de expresiones que puede utilizar para completar el elemento de la etiqueta con información interesante: recuerde que el código, o al menos la parte calculada, debe estar rodeada por `[%` y `%]` en el cuadro *Propiedades principales*:

- Muestra un título con el valor actual de la característica del atlas en «field1»:

```
'This is the map for ' || "field1"
```

  o, escrito en la sección *Propiedades principales*:

```
This is the map for [% "field1" %]
```

- Agregue una paginación para los objetos procesadas del atlas (por ejemplo, `Page 1/10`):

```
concat( 'Page ', @atlas_featurenumber, '/', @atlas_totalfeatures )
```

- Devuelve el nombre de los aeropuertos de la entidad de la región del atlas actual, según sus atributos comunes:

```
aggregate( layer := 'airports',
           aggregate := 'concatenate',
           expression := "NAME",
           filter := fk_regionId = attribute( @atlas_feature, 'ID' ),
           concatenator := ', '
         )
```

  O, si una *relación de atributos* está establecida:

```
relation_aggregate( relation := 'airports_in_region_relation',
                    aggregate := 'concatenate',
                    expression := "NAME",
                    concatenator := ', '
                  )
```

- Devuelve el nombre de los aeropuertos incluidos en la entidad de región del atlas actual, según su relación espacial:

```
aggregate( layer := 'airports',
           aggregate := 'concatenate',
           expression := "NAME",
           filter := contains( geometry( @parent ), $geometry ),
           concatenator := ', '
         )
```

  O:

```
array_to_string( array:= overlay_contains( layer := 'airports',
                                           expression := "NAME" ),
                 delimiter:= ', '
               )
```

- Devuelve la coordenada X inferior de la extensión del elemento `Map 1`:

```
x_min( map_get( item_variables( 'Map 1' ), 'map_extent' ) )
```

- Recupere el nombre de las capas en el elemento de diseño actual `Map1` y en formato de un nombre por línea:

```
array_to_string(
 array_foreach(
  map_get( item_variables( 'Map 1' ), 'map_layers' ), -- retrieve the layers␣
↪list
  layer_property( @element, 'name' ) -- retrieve each layer name
 ),
 '\n' -- converts the list to string separated by breaklines
)
```

- Muestre la lista de capas con sus cadenas de licencia (derechos de uso) en un elemento de diseño `Mapa 1`. Necesitas llenar las propiedades de capas primero " *Access metadata*

```
array_to_string( map_credits( 'Map 1', true ) )
```

18.2.5 El elemento leyenda

El elemento *Leyenda* es un cuadro o una tabla que explica el significado de los símbolos usados en el mapa. Luego, una leyenda se vincula a un elemento del mapa. Puede agregar un elemento de leyenda con la Herramienta  *Agregar leyenda* siguiendo las instrucciones de creación de elementos <create_layout_item>`y manipularla de la misma manera que se expone en :ref: interact_layout_item`.

De forma predeterminada, el elemento de la leyenda muestra todas las capas disponibles y se puede refinar usando su panel *Propiedades del elemento*. Aparte de las propiedades comunes de *elementos*, esta prestación tiene las siguintes funcionalidades (ver Figura 18.27):

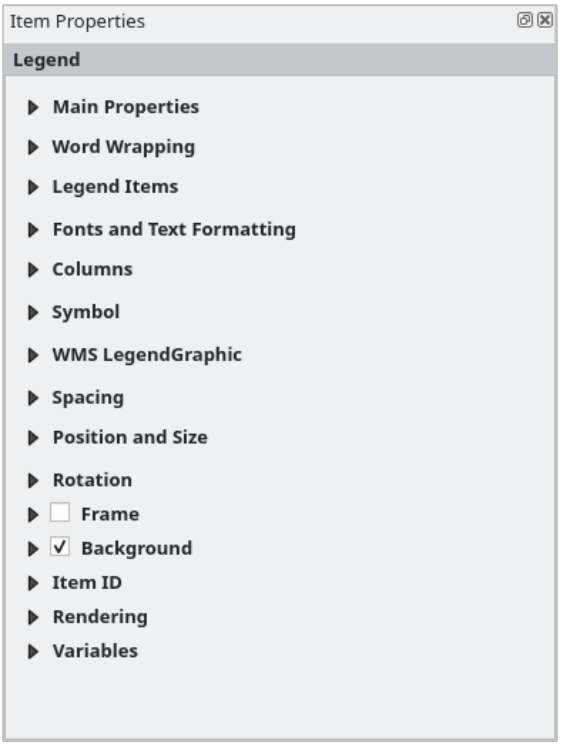


Figura 18.27: Panel de Propiedades del elemento Leyenda

Propiedades principales

El grupo *propiedades Principales* del panel de la leyenda *Propiedades del Elemento* prporciona las siguientes funcionalidades (ver Figura 18.28):

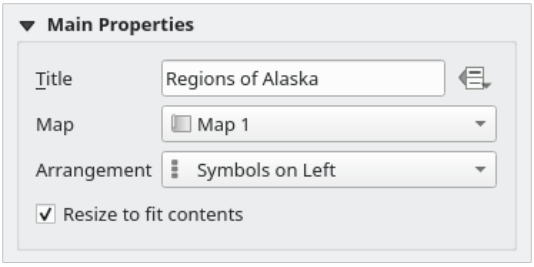


Figura 18.28: Grupo de propiedades Principales de Leyenda

En propiedades principales se puede:

- Cambiar el *Título* de la leyenda. Se puede dinamizar usando la configuración *suplantación definida por datos*, útil por ejemplo al generar un atlas;
- Elegir a qué elemento *Mapa* se referirá la leyenda actual. De forma predeterminada, se selecciona el mapa sobre el que se dibuja el elemento de leyenda. Si no hay ninguno, acude al *mapa de referencia*.

> **Nota**
>
> *Variables* del elemento de mapa vinculado (@map_id, @map_scale, @map_extent…) también son accesibles desde las propiedades definidas por datos de la leyenda.

- Establecer los símbolos y la ubicación del texto en la leyenda: la *Disposición* puede ser *Símbolos a la izquierda* o *Símbolos a la derecha*. El valor predeterminado depende de la configuración regional en uso (de derecha a izquierda o no).
- Utilice *Redimensionar para ajustar el contenido* para controlar si una leyenda debe redimensionarse automáticamente para ajustarse a su contenido. Si no se marca, la leyenda nunca cambiará de tamaño y, en su lugar, se mantendrá en el tamaño que el usuario haya establecido. Cualquier contenido que no se ajuste al tamaño se recorta.

Word wrapping

The *Word wrapping* group of the legend *Item Properties* panel provides the following functionalities:

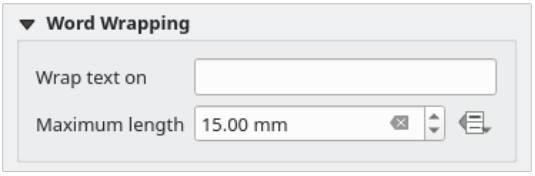


Figura 18.29: Legend's word wrapping group

- Envolver el texto de la leyenda en un carácter determinado: cada vez que aparece el carácter, se reemplaza con un salto de línea;
- Allow legend text to be automatically wrapped after a set line length (in millimeters), preventing very wide auto-generated legends. The *Maximum length* can be set as a static or data-defined value, convenient for dynamic layouts which adjust legend size based on for example page orientation or displayed features, or atlases where you may want to tweak the legend appearance on different pages.

  Examples:

  – Control legend width based on layout page orientation

  ```
  IF( @layout_pageheight >= @layout_pagewidth, 35, 80 )
  ```

  – Set the legend column width when a specific layer is displayed in the linked map

  ```
  CASE WHEN
    array_contains(
      map_get( item_variables('legend_1'), 'map_layer_ids' ),
      'a_specific_layer_id' )
  THEN 60
  END
  ```

Elementos de la leyenda

El grupo *elementos de la leyenda* del panel *Propiedades del Elemento* de la leyenda proporciona las siguientes funcionalidades (ver Figura 18.30):

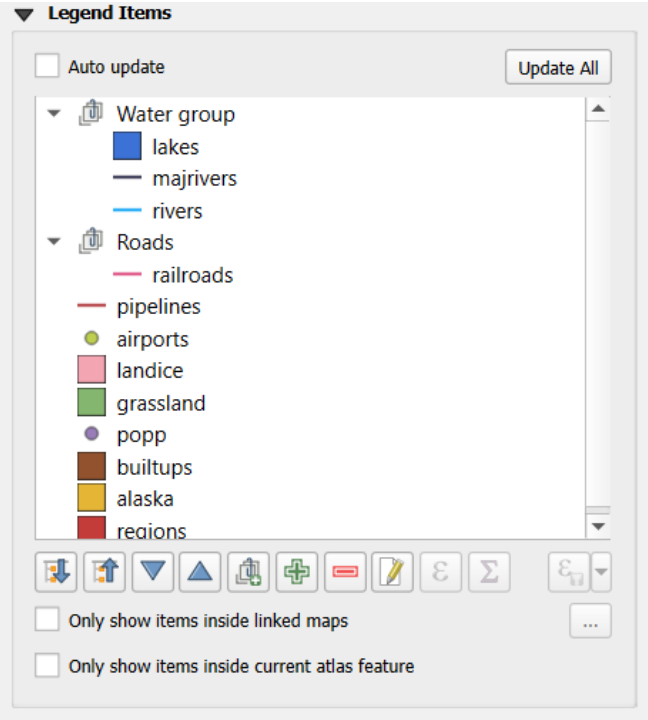


Figura 18.30: Grupo elementos de Leyenda

- La leyenda se actualizará de manera automática si *Autoactualizar* está marcado. Cuando *Autoactualizar* está desmarcado este le brinda mas control sobre los elementos de leyenda. Todos los iconos debajo de la lista de elementos de la leyenda serán activados.
- La ventana de elementos de la leyenda enumera todos los elementos de la leyenda y permite cambiar el orden de los elementos, agrupar capas, eliminar y restaurar elementos de la lista, editar los nombres de las capas y la simbología y añadir un filtro.
  - Utilice los botones Ampliar todo y Contraer todo para ampliar o contraer respectivamente todos los grupos y subgrupos del árbol de leyendas. Asegúrese de que la etiqueta :guilabel:` Auto actualizar` está desmarcada para utilizar estos botones.
  - El orden de los artículos se puede cambiar con los botones y o con la función de «arrastrar y soltar». El orden no se puede cambiar para los gráficos de leyenda de WMS.
  - Utilice para añadir un grupo a la leyenda.
  - Utilice el botón para agregar capas y el botón para eliminar grupos, capas o clases de símbolos.
  - El botón es usado para editar la capa, nombre de grupo o título. Primero necesita seleccionar el elemento de la leyenda. Doble click en el elemento también abre el cuadro de texto para renombrarlo.
  - El botón usa expresiones para personalizar cada etiqueta de símbolo de la capa seleccionada (ver *Definir por datos las etiquetas de leyenda*)
  - El botón agrega un recuento de objetos espaciales para cada clase de la capa vectorial.
  - Filtrar leyenda por expresión le ayuda a filtrar cuáles de los elementos de leyenda de una capa se mostrarán, es decir, utilizando una capa que tiene diferentes elementos de leyenda (por ejemplo, de una simbología basada en reglas o categorizada), puede especificar un expresión booleana para eliminar del árbol de leyendas, estilos que no tienen ningun objeto espacial que satisfaga una condición. Tenga en cuenta que, no obstante, los objetos espaciales se mantienen y se muestran en el elemento del mapa de diseño.

Si bien el comportamiento predeterminado del elemento de leyenda es imitar el árbol del panel *Capas*, que muestra los mismos grupos, capas y clases de simbología, hacer clic con el botón derecho en cualquier elemento le ofrece opciones para ocultar el nombre de la capa o aumentarlo como un grupo o subgrupo. En caso de que haya realizado algunos cambios en una capa, puede revertirlos eligiendo *Restablecer valores predeterminados* en el menú contextual de la entrada de leyenda.

Después de cambiar la simbología en la ventana principal de QGIS, puede hacer clic en *Actualizar todo* para adaptar los cambios en el elemento de leyenda del diseño de impresión.

- Con la casilla de verificación *Mostrar sólo elementos dentro de mapas enlazados*, sólo los elementos de la leyenda visibles en el mapa enlazado aparecerán en la leyenda. Si tiene más de un mapa, puede hacer clic en … y seleccionar otros mapas de su disposición. Esta herramienta permanece disponible cuando *Auto-actualizar* está activo.
- Al generar un atlas con entidades poligonales, puede filtrar los elementos de leyenda que se encuentran fuera de la entidad atlas actual. Para hacer eso, marque el *Mostrar solo elementos dentro del objeto atlas actual*.

Definir por datos las etiquetas de leyenda

ε le permite añadir *expresiones* para cada etiqueta de símbolo de una capa dada. Nuevas variables (`@symbol_label`, `@symbol_id` y `@symbol_count`) le ayudarán a interactuar con la entrada de leyenda.

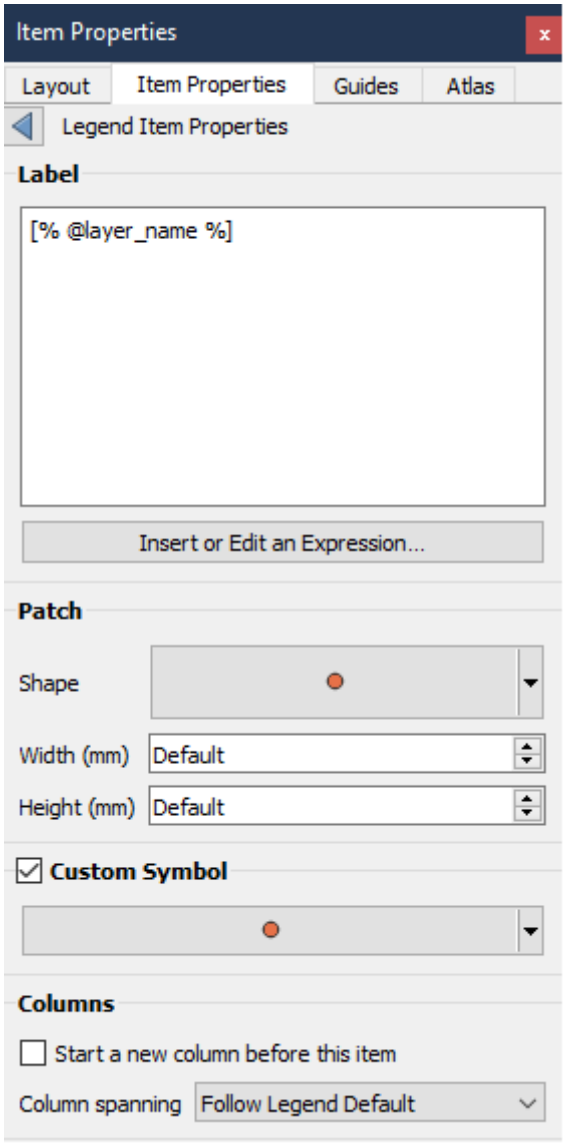
Por ejemplo, dada una capa `regions` categorizada por su campo `type`, puede agregar a cada clase en la leyenda su número de objetos y área total, p.ej. `Borough (3) - 850ha`:

1. Seleccione la capa de entrada en el árbol de leyenda
2. Presione el botón ε , abriendo el Diálogo *Constructor de cadenas de expresión*
3. Introduzca la siguiente expresión (*asumiendo que las etiquetas de los símbolos no han sido editadas*):

```
format( '%1 (%2) - %3ha',
        @symbol_label,
        @symbol_count,
        round( aggregate(@layer, 'sum', $area, filter:= "type"=@symbol_label)/
↪10000 )
      )
```

4. Presiona *Aceptar*

Personalización de elementos de leyenda



Los elementos de leyenda también se pueden personalizar individualmente en *Propiedades de elementos de leyenda*. Pero estas personalizaciones solo se pueden hacer con *Actualización automática* deshabilitada.

Haciendo doble clic en un elemento o presionando Edit selected item properties permite una mayor personalización.

Etiqueta

Para todos los tipos de elementos, permite modificar el texto de la etiqueta escribiendo o insertando expresiones usando la **|expresión|** *Insertar o Editar una Expresión*. Las expresiones también se pueden agregar directamente en cualquier parte de la etiqueta del elemento mediante la notación [% expresión %].

Columnas

La Propiedad del elemento de leyenda también le permite controlar el comportamiento de división de la columna forzando que la división de la columna ocurra después de un elemento específico o todos los símbolos de una capa. La división automática de una capa y su elemento secundario también se puede permitir o bloquear por capas en este widget.

Parche

Para elementos con un símbolo, la propiedad de elemento de leyenda le permite especificar la altura y el ancho máximos que puede ocupar un símbolo.

Para símbolos vectoriales, puede especificar una forma personalizada para el símbolo. Las formas generalmente se definen mediante una expresión para representar la geometría en un plano simple, pero esos símbolos también se pueden guardar en el administrador de estilos e importarlos más tarde. El símbolo predeterminado para cada tipo de geometría también se puede controlar a través del administrador de estilos.

Símbolo personalizado

También se puede especificar un símbolo personalizado para símbolos vectoriales. Esto puede ser útil para modificar la representación de un símbolo específico, para mejorarlo en la leyenda o tener un símbolo independiente de su vista previa de símbolo real. Este símbolo personalizado anulará el símbolo de la leyenda, pero tendrá en cuenta el símbolo *Parche* especificado.

Formateo de fuentes y texto

El grupo *Formateo de fuentes y texto* del panel *Propiedades del elemento* de la leyenda proporciona las siguientes funcionalidades:

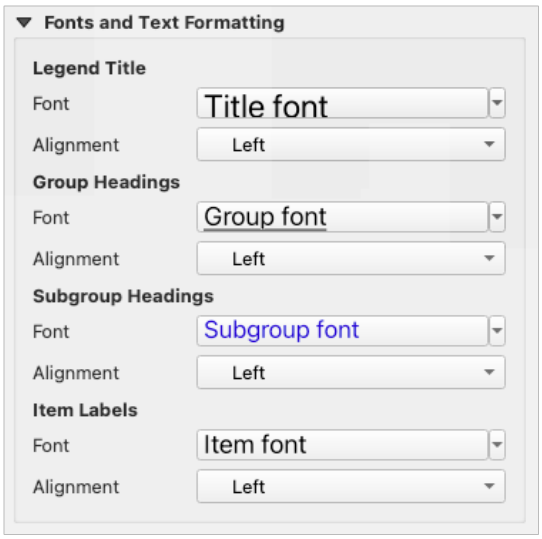


Figura 18.31: Propiedades de Fuente de Leyenda

- Puede cambiar el tipo de letra del título, grupo, subgrupo y objeto espacial de la leyenda utilizando el *selector de fuente* widget, que ofrece todas las funciones de *formateo de texto* (control del espaciado entre fuentes, formato HTML mixto, coloreado, difuminado, fondo, búfer de texto, sombra, …)
- Para cada uno de estos niveles, puede establecer la *Alineación* del texto: puede ser :guilabel:` Izquierda` (predeterminado para locales basados en izquierda a derecha), *Centro* o :guilabel:` Derecha `(predeterminado para locales basados en derecha a izquierda).

Columnas

Bajo el grupo *Columnas* de la leyenda del panel *Propiedades de elemento*, los elementos de la leyenda se pueden organizar en varias columnas:

- Establezca el número de columnas en el campo *Número* 1,00. Este valor se puede hacer dinámico, por ejemplo, siguiendo los objetos del atlas, el contenido de la leyenda, el tamaño del marco…
- *Anchura de columnas igual* controla cómo deberían ser ajustadas las columnas de la leyenda.
- La opción *Dividir capas* permite a una capa categorizada o graduada de la leyenda repartirse en columnas.

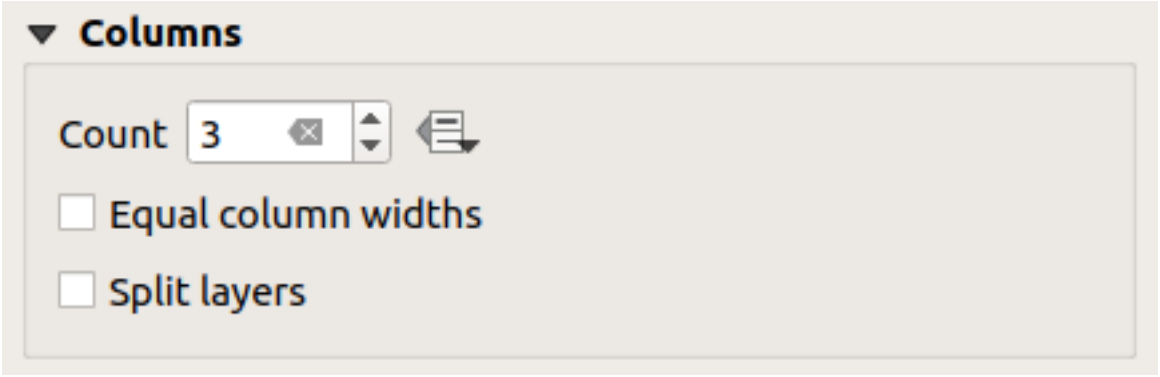


Figura 18.32: Ajustes de Columnas de Leyenda

Símbolo

El grupo *Símbolo* de la leyenda del panel *Propiedades del elemento* configura el tamaño de los símbolos que se muestran junto a las etiquetas de la leyenda. Usted puede:

- Establezca la *Anchura del símbolo* y la :guilabel:` Altura del símbolo`
- Establecer los marcadores" *Tamaño mínimo de símbolo* y *Tamaño máximo de símbolo*: `0.00mm` significa que no hay ningún valor establecido.
- *Dibujar delimitador para símbolos raster*: esto agrega un contorno al símbolo que representa el color de la banda de la capa raster; puede establecer tanto el *Color de marca* como la *Altura*.

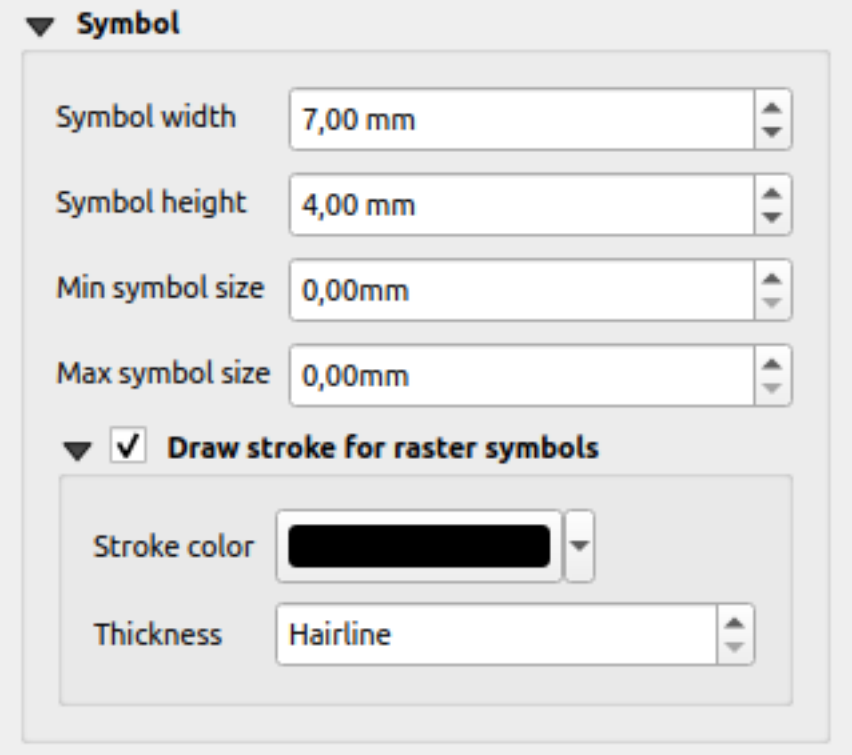


Figura 18.33: Configuración de Símbolo de Leyenda

Gráfico de leyenda de WMS

La sección *WMS LegendGraphic* del panel de leyenda *Item Properties* proporciona las siguientes funcionalidades (ver Figura 18.34):

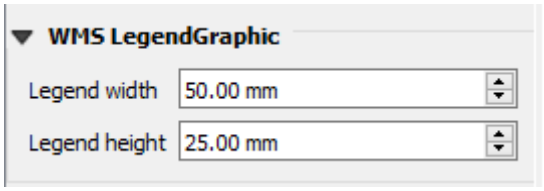
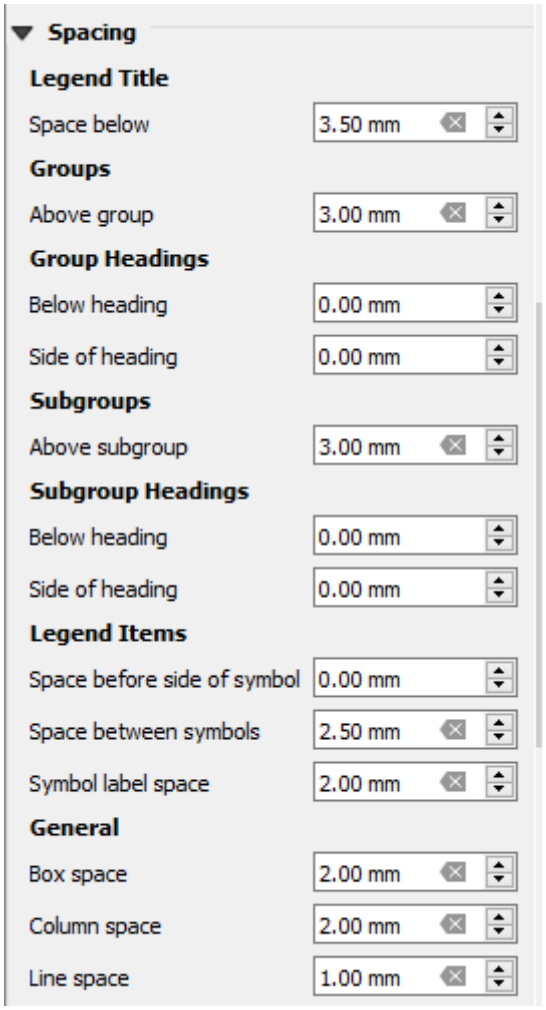


Figura 18.34: Gráfico de leyenda de WMS

Cuando haya agregado una capa WMS e insertado un elemento de leyenda, se enviará una solicitud al servidor WMS para proporcionar una leyenda WMS. Esta leyenda solo se mostrará si el servidor WMS tiene la capacidad GetLegendGraphic. El contenido de la leyenda de WMS se proporcionará como una imagen rasterizada.

GráficoDeLeyenda WMS se utiliza para poder ajustar el *Anchura de la leyenda* y la *Altura de la leyenda* de la imagen ráster de la leyenda WMS.

Espaciado



La sección *Espaciado* te permite personalizar el espaciado dentro de la leyenda. El espaciado puede ser de gran ayuda para indicar la agrupación de elementos en la leyenda y su relación.

Espaciado alrededor y antes del título, grupos, subgrupos, símbolos, etiquetas, cuadros, columnas y líneas se pueden personalizar a través de este diálogo.

18.2.6 El elemento de barra de escala

Las barras de escala proporcionan una indicación visual del tamaño de las entidades y la distancia entre ellas en el elemento del mapa. Un elemento de barra de escala requiere un elemento de mapa. Utilice la herramienta *Agregar barra de escala* como sigue *instrucciones de creación de elementos* para agregar un nuevo elemento de barra de escala que luego pueda manipular de la misma manera que se expone en *Interactuando con los elementos de composición*.

De forma predeterminada, un nuevo elemento de barra de escala muestra la escala del elemento del mapa sobre el que se dibuja. Si no hay ningún elemento de mapa debajo, se utiliza *mapa de referencia*. Puede personalizarlo en el panel *Propiedades del elemento*. Aparte de *propiedades comunes de los elementos*, esta prestación tiene las siguientes funcionalidades (ver Figura 18.35):

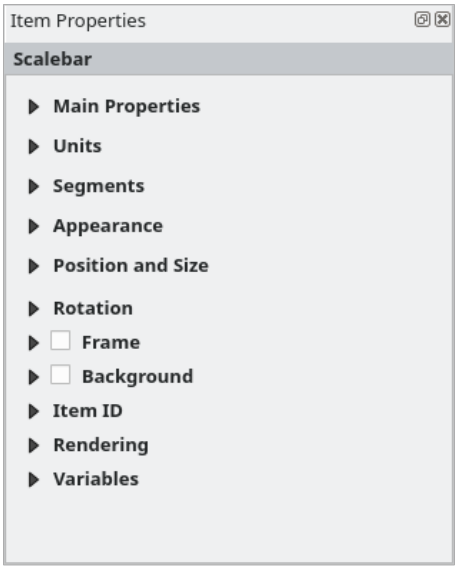


Figura 18.35: Panel de Propiedades del elemento de Barra de Escala

Propiedades principales

El grupo *propiedades Principales* del panel de la barra de escala *propiedades del elemento* proporciona las siguientes funcionalidades (ver Figura 18.36):

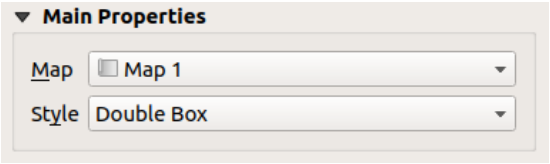


Figura 18.36: Grupo de propiedades principales de la Barra de Escala

1. Primero, selecciona el mapa al que la barra de escala se adjuntará
2. Luego, escoja el estilo de la barra de escala. Estilos disponibles son:
   - Estilos **Recuadro simple** y **recuadro doble**, que contienen una o dos líneas de recuadros de colores alternos;
   - **Líneas con marcas en el centro**, **Líneas con marcas arriba** o **Líneas con marcas abajo**;
   - Estilo **línea escalonada** que dibuja una representación de línea escalonada de la barra de escala
   - Estilo **hueco** que dibuja un solo cuadro con color alterno para los segmentos, con líneas horizontales a través de segmentos alternos
   - **Numérico**, donde se imprime la relación de escala (por ejemplo, `1:50000`).
3. Establecer propiedades según corresponda

Unidades

El grupo *Unidades* de la barra de escala:guilabel:El panel` Propiedades del elemento` proporciona las funcionalidades para establecer las unidades de visualización y algunos formatos de texto (ver Figura 18.37):

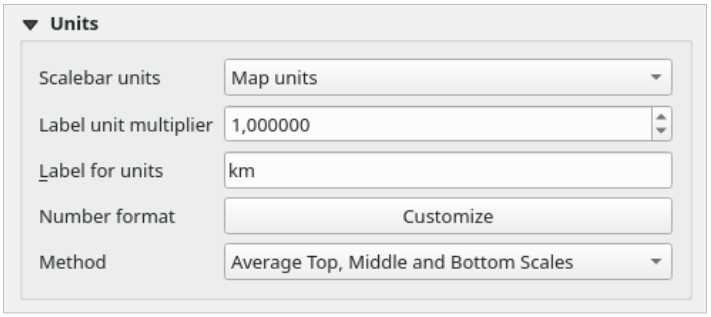


Figura 18.37: Grupo Unidades de la Barra de Escala

- Seleccione las unidades a usar con *unidades de la barra de escala*. Hay varias elecciones posibles: **Unidades de mapa** (la predeterminada), **Metros**, **Pies**, **Millas** o **Millas Nauticas**… y algunas derivadas. La conversión de unidades se gestiona automáticamente.
- El *Multiplicador de unidad de etiqueta* especifica cuántas unidades de barra de escala por unidad etiquetada. Por ejemplo, si las unidades de la barra de escala se establecen en «metros», un multiplicador de 1000 dará como resultado las etiquetas de la barra de escala en «kilómetros».
- El campo *Etiqueta para unidades* define el texto utilizado para describir las unidades de la barra de escala, p. Ej. `m` or `km`. Esto debe coincidir para reflejar el multiplicador anterior.
- Presione *Personalizar* junto a *Formato de número* para tener control sobre todas las propiedades de formato de los números en la barra de escala, incluidos miles de separadores, lugares decimales, notación científica, etc. (ver *Formato de Números* para más detalles). Muy útil en el caso de hacer mapas para audiencias fuera de la configuración regional actual de QGIS, o cuando le gustaría variar el estilo de los valores predeterminados de la configuración regional (por ejemplo, agregar separadores de miles cuando la configuración regional predeterminada es ocultarlos).
- Select the *Method* for scale bar calculation: Depending on the map CRS and extent, the distance reported in the scale bar may not be the same measured over the map. This option, which defaults to the *project's scale calculation method*, helps you specify the one you would like to use for this particular scale bar.

Segmentos

El grupo *Segmentos* del panel de la barra de escala *Propiedades del Elemento* proporciona las funcionalidades para configurar el número y tamaño de segmentos y subdivisiones (ver Figura 18.38):

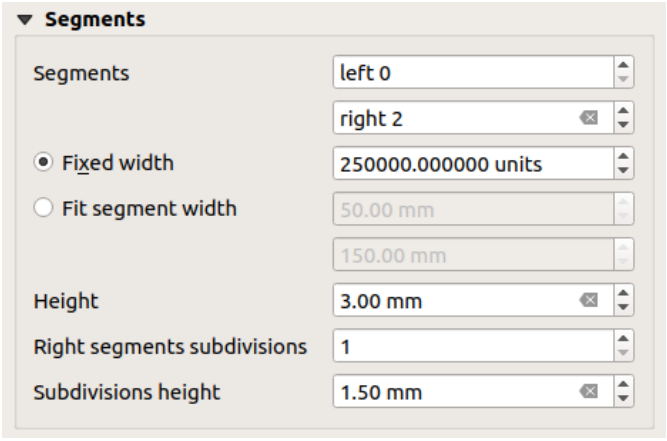


Figura 18.38: Grupo Segmentos de la Barra de Escala

- Puede definir el número de *Segmentos* que se dibujará a los lados izquierdo y derecho del `0` de la barra de escala:
  - número de subdivisiones de un segmento único en el lado *Izquierdo*
  - número de segmentos en el lado *Derecho*
- Puede establecer la anchura de un segmento o un intervalo para la longitud total de la barra de escala:
  - Establezca la longitud de un segmento en las unidades de la barra de escala (*Ancho fijo*)
  - O limitar el tamaño total de la barra de escala en `mm` con la opción *Ajustar el ancho del segmento*. En este último caso, cada vez que cambia la escala del mapa, la barra de escala se redimensiona (y su etiqueta se actualiza) para ajustarse entre el rango superior e inferior establecidos.
- *Altura* se usa para definir la altura de la barra.
- *Subdivisiones del segmento derecho* se utiliza para definir el número de secciones que pueden tener los segmentos del lado derecho de la barra de escala (para los estilos de barra de escala *línea con marcas abajo*, *línea con marcas en el centro* y *líneas con marcas arriba*) .
- *altura de Subdivisión* es usada para definir la altura del segmento de subdivisión.

Apariencia

El grupo *Apariencia* del panel *Propiedades del elemento* de la barra de escala proporciona las siguientes funcionalidades:

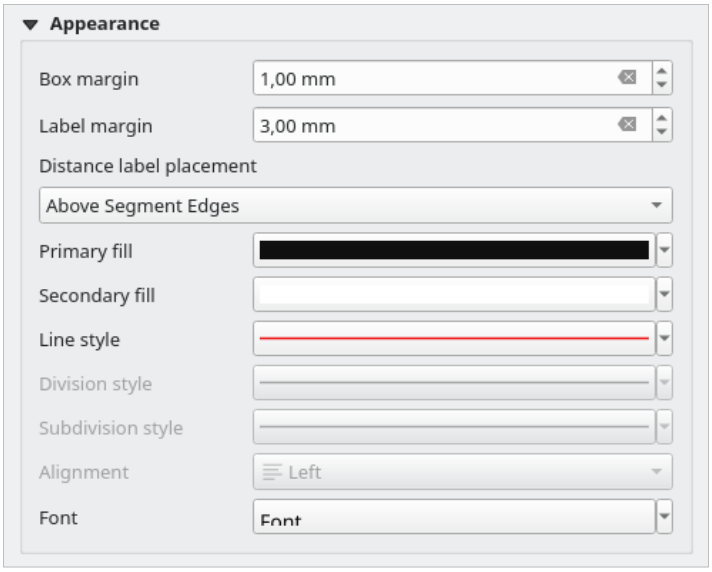


Figura 18.39: Grupo Apariencia de la barra de escala

Se puede definir como la barra de escala se mostrará en un marco.

- *Margen de la caja* : espacio entre texto y bordes del marco
- *margen de las etiquetas*: espacio entre el texto y el dibujo de la barra de escala
- *Colocación de la etiqueta de distancia* define la posición del texto en relación con los segmentos de la barra de escala: *Sobre los bordes de los segmentos*, *Sobre los centros de los segmentos*, *Bajo los bordes de los segmentos* o *Bajo los centros de los segmentos*.
- *Relleno primario* y *relleno secundario* del trazado de la barra de escala usando *propiedades de relleno de símbolo* (color, opacidad, patrones, efectos…) — para estilos *Recuadro Simple*, *recuadro Doble* y *Hueco*
- *Estilo de línea* del trazado de la barra de escala usando *propiedades de línea de símbolos* (color, anchura, unión, estilo extremos, patrones, efectos…) — para todos menos el estilo *Numérico*color
- *Estilo de División* y *Estilo de Subdivisión* respectivamente para segmentos de división y subdivisión en estilos de barra de escala *Línea con marcas arriba*, *Línea con marcas al centro* y *Línea con marcas abajo* usando *propiedades de símbolo de línea* (color, anchura, unión, estilo en extremos, patrones, efectos…)
- *Alineación* pone el texto en el lado izquierdo, centro o derecho del marco (solo para estilos de barras de escalas *Numérico*)
- *Fuente* para establecer las *propiedades* (tamaño, fuente, color, espaciado de letra, sombra, fondo…) de la etiqueta de barra de escala.

Dado que la mayoría de las propiedades de visualización de la barra de escala se basan en símbolos cuyas propiedades pueden definirse por datos, es posible representar barras de escala definidas por datos.

Ejemplo: El siguiente código aplicado a la propiedad en negrita de las etiquetas de escala mostrará los números en negrita cuando sean múltiplos de 500:

```
-- returns True (or 1) if the value displayed on the bar
-- is a multiple of 500

@scale_value % 500 = 0
```

18.2.7 Los elementos de la tabla

Puede usar los elementos de la tabla para decorar y explicar su mapa:

- *Tabla de atributos*: extrae automáticamente un subconjunto de los atributos de una capa, basándose en reglas predefinidas.
- *Tabla fija*: permite crear tablas con contenido introducido manualmente (es decir, al estilo de una hoja de cálculo), de modo que se pueden crear tablas totalmente personalizadas.

El elemento de la tabla de atributos

Los atributos de cualquier capa del proyecto pueden ser mostrados en el diseño de impresión. Utilice la herramienta  *Añadir tabla de atributos* siguiendo las *instrucciones de creación de elementos* para añadir un nuevo elemento de la tabla que luego pueda manipular de la misma manera que se expone en *Interactuando con los elementos de composición*.

Por defecto, un elemento nueva tabla de atributos carga primero filas de la primera capa (ordenada alfabeticamente), con todos los campos. No obstante puedes personalizar la tabla gracias a su panel *Propiedades del elemento*. Aparte de las *propiedades principales del elemento*. esta prestación tiene las siguientes funcionalidades Figura 18.40):

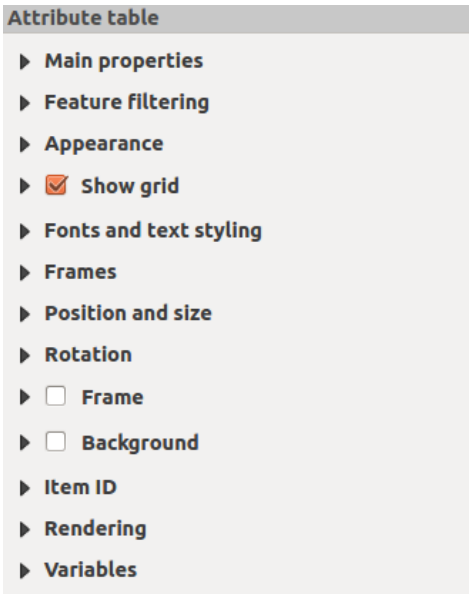


Figura 18.40: Panel de Propiedades del elemento Tabla de Atributos

Propiedades principales

El grupo *Propiedades principales`de la tabla de atributos proporciona las siguientes funcionalidades (ver :numref:`figure_layout_table_ppt*):

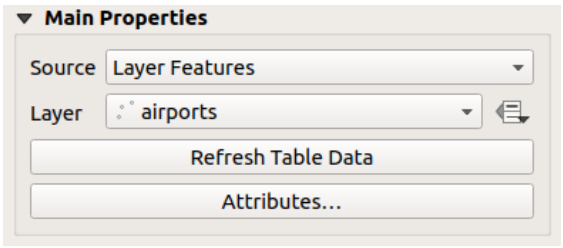


Figura 18.41: Grupo de Propiedades Principales de la Tabla de Atributos

- Para *Fuente*, de forma predeterminada, solo puede seleccionar **Objetos espaciales de la capa**, lo que le permite seleccionar una :guilabel:` Capa` de las capas vectoriales cargadas en el proyecto.

  El botón suplantación definida por datos cerca de la lista de capas le permite cambiar dinámicamente la capa que se utiliza para completar la tabla, p. ej. podría llenar la tabla de atributos con diferentes atributos de capa por página de atlas. Tenga en cuenta que la estructura de la tabla utilizada (Figura 18.44) es la de la capa que se muestra en la lista desplegable :guilabel:` Capa` y se deja intacta, lo que significa que establecer una tabla definida por datos en una capa con campo diferente(s) darán como resultado columna vacía(s) en la tabla.

  En caso de que active *Generar un atlas* en el panel *Atlas* (ver *Generar un Atlas*), hay dos *Fuentes* adicionales posibles:

  - **Objeto de atlas actual** (ver Figura 18.42): no verá ninguna opción para seleccionar la capa, y el elemento de tabla solo mostrará una fila con los atributos del objeto actual de la capa de cobertura del atlas.
  - y **Relación de hijos** (ver Figura 18.43): se mostrará una opción con los nombres relacionados. Esta funcionalidad solo puede ser usada si has definido una *relacion* al usar tu capa de cobertura del atlas como padre, y la tabla mostrará las filas con las capas de hijos de la capa de cobertura del atlas del elemento actual.

- El botón *Refrescar Tabla de Datos* se puede usar para actualizar la tabla cuando el contenido real de la tabla ha cambiado.

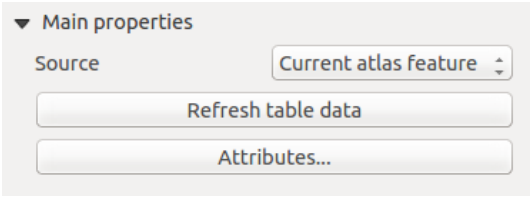


Figura 18.42: Propiedades principales de Tabla de atributos para “Objeto del atlas actual”

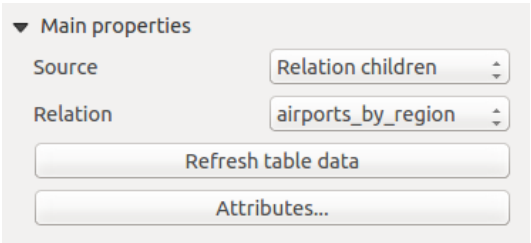


Figura 18.43: Propiedades Principales de Tabla de Atributos para “Hijo de Relación”

- El botón *Atributos…* carga el diálogo *Seleccionar Atributos*, (ver Figura 18.44) que puede ser usado para cambiar los contenidos visibles de la tabla. La parte superior de la ventana muestra la lista de los atributos a mostrar y la parte inferior le ayuda a ordenar los datos.

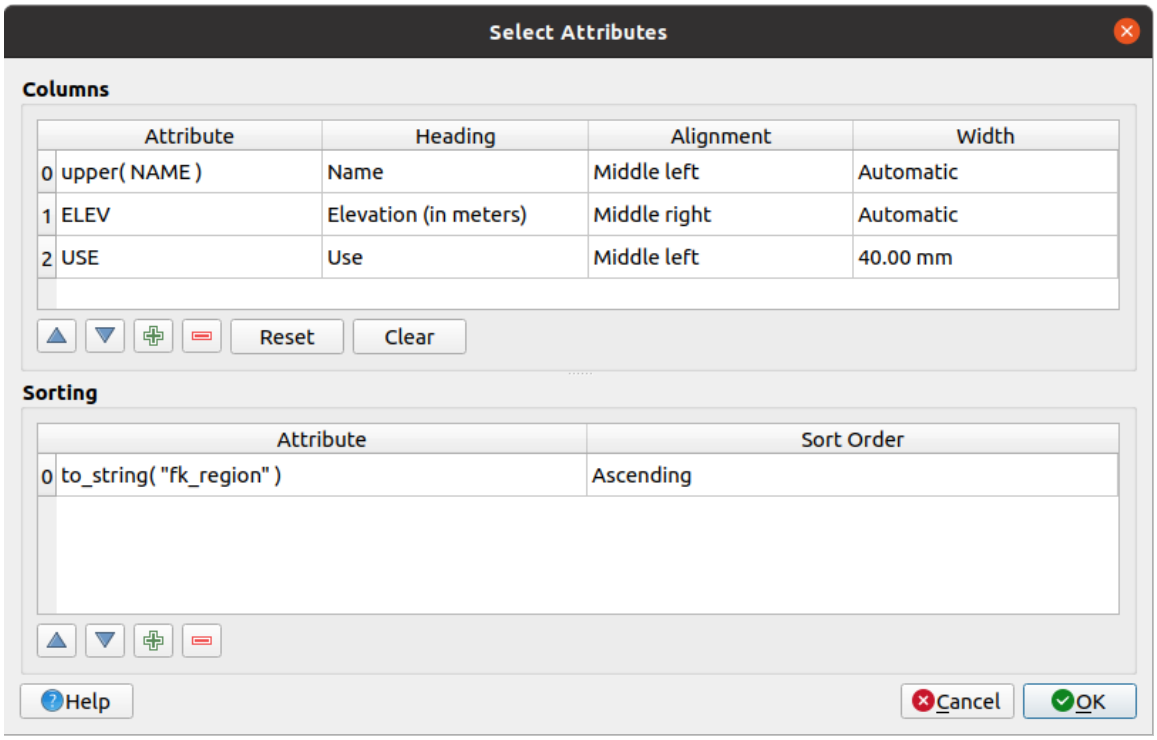


Figura 18.44: Diálogo Seleccionar Atributos de la Tabla de Atributos

En la sección *Columnas* se puede:

- Mueva los atributos hacia arriba o hacia abajo en la lista seleccionando las filas y luego usando los botones y para cambiar las filas. Se pueden seleccionar y mover varias filas a la vez.
- Agregue un atributo con el botón . Esto agregará una fila vacía en la parte inferior de la tabla donde puede seleccionar un campo para que sea el valor del atributo o crear un atributo a través de una expresión regular.
- Elimine un atributo con el botón . Se pueden seleccionar y eliminar varias filas al mismo tiempo.
- Restablezca la tabla de atributos a su estado predeterminado con el botón *Restablecer*.
- Limpia la tabla usando el botón *Limpiar*. Esto es útil cuando tiene una tabla grande pero solo desea mostrar una pequeña cantidad de atributos. En lugar de eliminar manualmente cada fila, puede ser más rápido borrar la tabla y agregar las filas necesarias.
- Los encabezados de las celdas se pueden modificar agregando el texto personalizado en la columna *Encabezado*.
- La alineación de las celdas se puede gestionar con la columna *Alineación* que dictará la posición del texto dentro de la celda de la tabla.
- El ancho de celda se puede administrar manualmente agregando valores personalizados a la columna *Anchura*.

En la sección *Ordenar* se puede:

- Añadir un atributo con el cuál ordenar la tabla: presiona el botón y se añadirá una nueva fila vacía. Inserta un campo o una expresión en la columan *Atributos* y selecciones el *Orden de ordenación* a **Ascendente** o **Descendente**.
- Seleccione una fila en la lista y use los botones y para cambiar la prioridad del orden en el nivel del atributo. Seleccionando una celda en la columna *Orden de ordenación* le ayuda a cambiar el orden de ordenación de un atributo de campo.

- Use el botón para borrar un atributo de la lista de ordenación.

Filtrado de objetos espaciales

El grupo *Filtrado de Objetos Espaciales* de la tabla de atributos proporciona las siguientes funcionalidades (ver Figura 18.45):

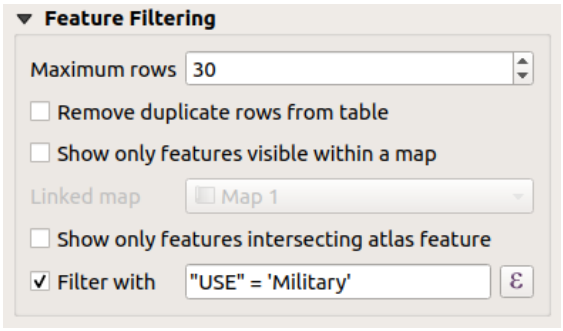


Figura 18.45: Grupo Filtrado de Objetos Espaciales de la Tabla de Atributos

Puede:

- Defina *Máximo de filas* que se mostrarán.
- Activar *Eliminar filas duplicadas de la tabla* para mostrar sólo registros únicos.
- Activar *Mostrar solo objetos espaciales visibles dentro de un mapa* y seleccione el correspondiente :guilabel:` Mapa enlazado` cuyos atributos de entidades visibles se mostrarán.
- Activar *Mostrar solo objetos que intersecan con el objeto Atlas* sólo está disponible cuando *Generar un atlas* está activado. Cuando se activa, mostrará una tabla con solo los bojetos que intersecan con el objeto atlas actual.
- Activar *Filtrar con* y proporciona un filtro al escribir en la línea de entrada o insertar una expresión regular utilizando el botón de expresión dado. Unos pocos ejemplos de declaraciones de filtrado que puede utilizar cuando se han cargado la capa de aeropuertos a partir del conjunto de datos de la muestra:
  - `ELEV > 500`
  - `NAME = 'ANIAK'`
  - `NAME NOT LIKE 'AN%'`
  - `regexp_match( attribute( $currentfeature, 'USE' ) , '[i]')`

  La ultima expresión regular incluirá sólo los aeropuertos que tienen una letra “i” en el campo de atributos “USE”.

Apariencia

El grupo *Apariencia`de la tabla de atributos proporciona las siguientes funcionalidades (ver :numref:`figure_layout_table_appearance*):

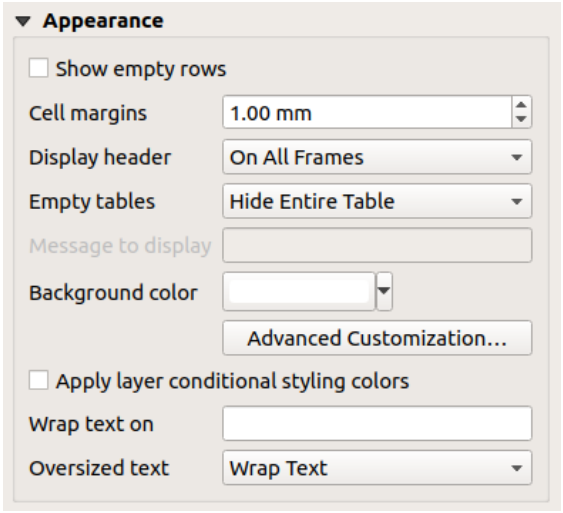


Figura 18.46: Grupo Apariencia de la Tabla de Atributos

- Haga clic en *Mostrar filas vacías* para llenar la tabla de atributos con celdas vacías. ¡Esta opción también se puede usar para proporcionar celdas vacías adicionales cuando tenga un resultado para mostrar!
- Con *Márgenes de celda* puede definir el margen alrededor del texto en cada celda de la tabla.
- Con *Mostrar cabecera* puede seleccionar de una lista una de las opciones “En el primer marco”, “En todos los marcos” o “Sin encabezado”.
- La opción *Tabla vacía* controla lo que se mostrará cuando la selección de resultados esté vacía.
  - **Dibujar cabeceras solamente**, sólo dibuja la cabecera excepto si se tiene elegido “Sin encabezado” por *Mostrar encabezado*.
  - **Ocultar tabla entera**, sólo dibujará el fondo de la tabla. Se puede activar *No dibujar el fondo si el marco esta vacío* en *Marcos* para ocultar la tabla completamente.
  - **Mostrar mensaje del conjunto** dibujará el encabezado y añadirá un espaciado a la celda a todas las columnas y mostrará un mensaje como “Sin resultados” que puede ser proveído en la opción *Mensaje a mostrar*
- La opción *Mensaje a mostrar* sólo se activa cuando se ha seleccionado **Mostrar mensaje establecido** para *Tabla Vacía*. El mensaje proporcionado se mostrará en la tabla en la primer fila, cuando el resultado sea una tabla vacía.
- Con *Color de fondo* puede establecer el color de fondo de la tabla usando el control selector de color. La opción *Personalización avanzada* le ayuda a definir diferentes colores de fondo para cada celda (ver Figura 18.47)

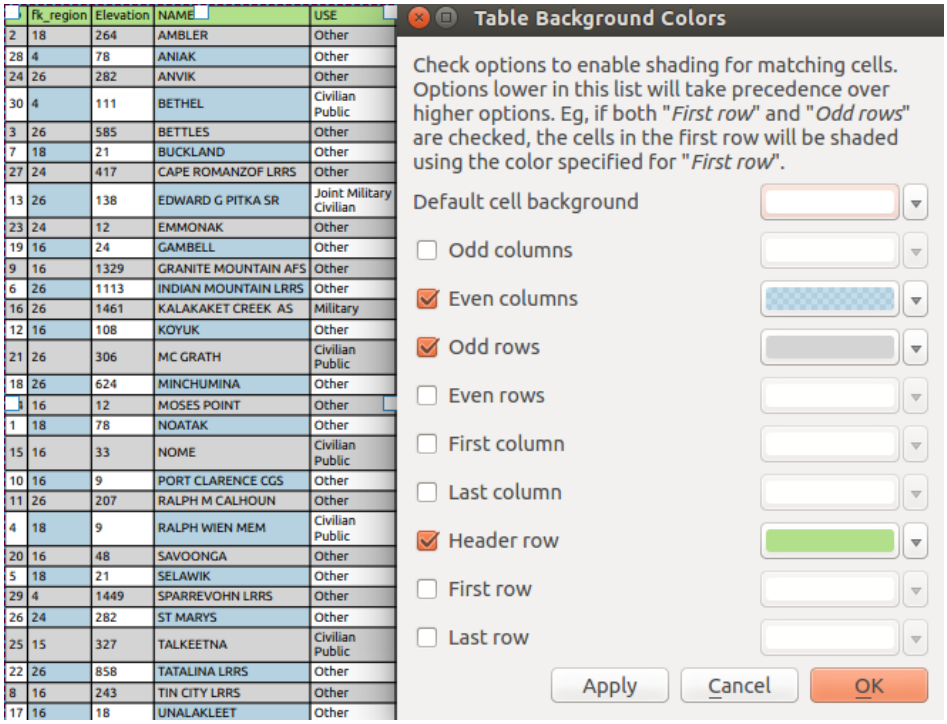


Figura 18.47: Diálogo Fondo Avanzado de la Tabla de Atributos

- *Aplicar colores de estilo condicional de capa*: el *formato de tabla condicional* presente en la capa se aplica dentro de la tabla de atributos de diseño (color de fondo, familia de fuente y propiedades como negrita, cursiva, tachado, subrayado, color, …). Las reglas de formato condicional tienen preferencia sobre otros ajustes de formato de la tabla de diseño, p.ej. anularán otros ajustes de color de fondo como los colores alternativos de fila.
- Con la opción *Envolver texto*, puede definir un carácter en el que se envolverá el contenido de la celda cada vez que se den las condiciones
- Con *Texto extragrande* se define el comportamiento cuando el ancho establecido para una columna es menor que la longitud de su contenido. Puede ser **Ajustar texto** o **Truncar texto**.

Nota

Mas propiedades del elemento tabla de atributos son descritas en la sección *Funcionalidades generales de las tablas*.

El elemento tabla fija

Información adicional sobre el mapa puede ser insertada manualmente en la tabla eligiendo *Añadir tabla fija* y siguiendo *Instrucciones de creación de elementos* para añadir un nuevo elemento tabla que luego puede manipular de la misma manera que se expone en *Interactuando con los elementos de composición*.

De forma predeterminada, aparece una tabla vacía con dos columnas y filas minimizadas en el diseño del mapa. Tiene que personalizar la tabla en el panel *Propiedades del elemento*. Aparte de *propiedades generales de los elementos*, este objeto tiene las siguiente funcionalidades:

Propiedades principales

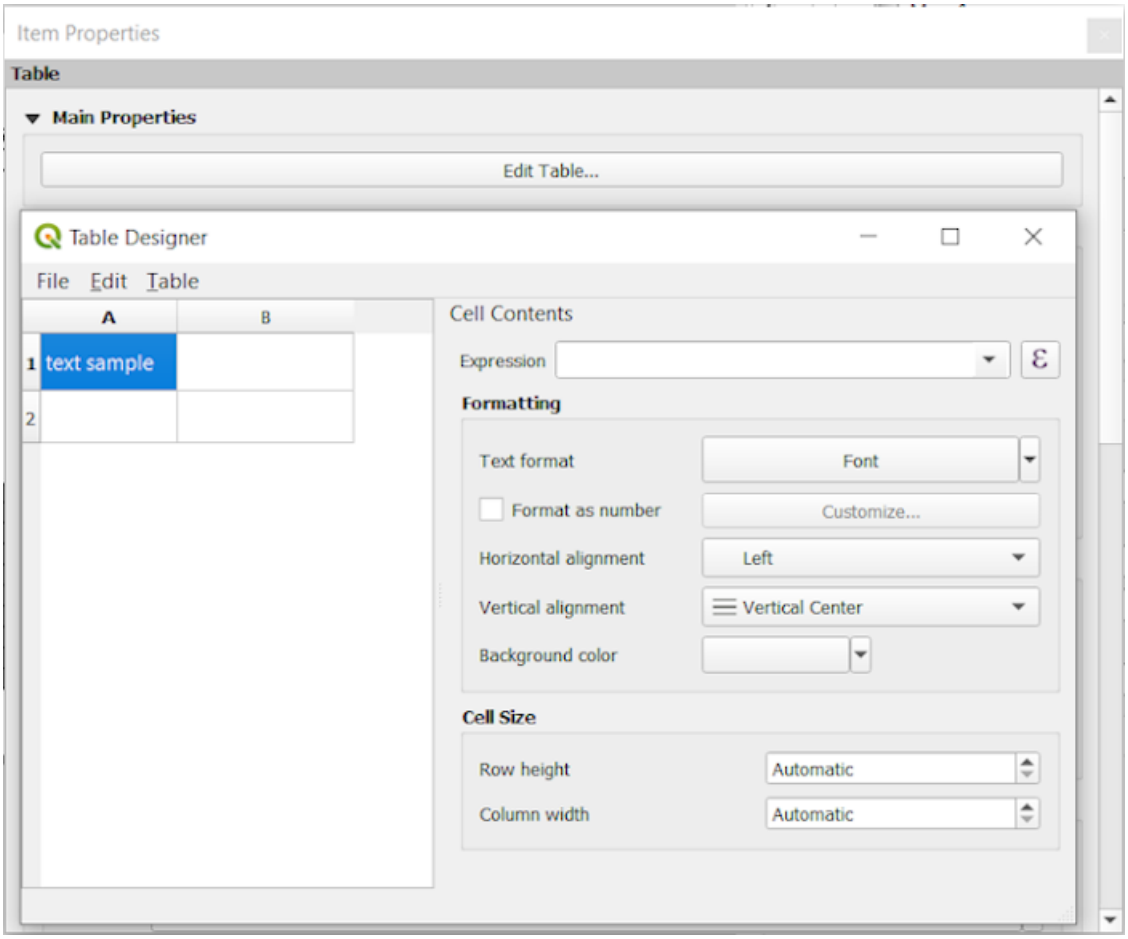


Figura 18.48: Panel de propiedades de elemento tabla fija con diseñador de tabla

In *Main properties*, press *Edit table…* to open the *Table designer* dialog where you can build your table (double-clicking the table item will also open the editor dialog):

- En la parte izquierda del cuadro de diálogo, un marco de tabla muestra el elemento de tabla activo con su contenido y configuración. Haga clic o doble clic en una celda y escriba un texto para modificar su contenido. Presione `Ctrl+Intro` para añadir un salto de línea (para un texto de varias líneas). La lógica para seleccionar varias celdas en la tabla es la misma que en la capa vectorial *tabla de atributos*. Arrastra y mueve un separador de fila o columna para cambiar el tamaño de la fila superior o la columna anterior. Al hacer doble clic en el separador, se cambiará el tamaño de la fila superior o la columna anterior para ajustarse a su contenido. Al hacer clic con el botón derecho sobre los encabezados de columna o fila, se muestra un menú contextual para insertar columnas o filas alrededor de la selección o para eliminarlas.
- A través de los menús situados en la parte superior del cuadro de diálogo, es posible:
  - *Archivo* ► *Importar contenido del portapapeles*: sobrescribe el contenido de la tabla.
  - *Archivo* ► *Cerrar editor*: cierra el cuadro de diálogo *Diseñador de tablas*.
  - Desde el menú *Editar*, puede trabajar con las funciones de selección de filas y columnas:
    - *Seleccionar todas* las celdas de la tabla
    - *Seleccionar columnas* o *Seleccionar filas*: cuando se seleccionan celdas en la tabla, puede ampliar la selección respectivamente a sus columnas o filas.
    - *Borrar celdas*: elimina el contenido de las celdas seleccionadas.

  - El menú *Tabla* es el lugar donde se diseña la estructura de la tabla. Puede:
    * *Insertar filas* ► encima o debajo de la selección
    * *Insertar columnas* ► antes o después de la selección
    * *Eliminar filas* o *Eliminar columnas* de las celdas seleccionadas
    * *Combinar celdas seleccionadas*: seleccione varias celdas en forma rectangular y podrá combinarlas y concatenar su contenido en la celda superior izquierda (cuyo estilo también se aplica a la celda combinada, a menos que haya una celda de última fila/columna en la selección).
    * *Dividir celdas seleccionadas*: seleccione una celda combinada y podrá volver a dividirla en sus celdas individuales. El texto actual se mantiene en la celda superior izquierda, y las demás celdas se rellenan con su contenido antes de combinarse. También se les vuelve a aplicar su estilo original.
    * **|casilla de verificación|** *Incluir fila de encabezado*, cuyo estilo solo se puede controlar desde el widget *Estilo de Fuentes y Texto*.
- Seleccione las celdas y, a la derecha del marco de la tabla, puede proporcionar el *Contenido de la celda*:
  - Introduzca manualmente el contenido de cada celda o utilice una **|expresión|** *Expresión* para rellenarla automáticamente.
  - En el grupo *Formato*, establezca opciones específicas que se aplican a la selección y tienen prioridad sobre el estilo global del contenido de la tabla. <layout_table_fonts_and_styling>`:
    * Eligiendo el *Formato de texto*
    * Por  *Formato como número* (varios formatos están disponibles)
    * Definiendo la *Alineación horizontal* y la *Alineación vertical*
    * Eligiendo un *Color de fondo*
  - Defina el *Tamaño de celda* con *Alto de fila* y *Ancho de columna*.

Apariencia

El grupo *Apariencia* de la tabla fija proporciona las siguientes funcionalidades:

- Haga clic en  *Mostrar filas vacías* para rellenar la tabla de atributos con celdas vacías.
- Con *Márgenes de celda* puede definir el margen alrededor del texto en cada celda de la tabla.
- Con *Mostrar cabecera* puede seleccionar de una lista una de las opciones “En el primer marco”, “En todos los marcos” o “Sin encabezado”.
- Con *Color de fondo* puede establecer el color de fondo de la tabla usando el widget *selector de color*. La opción *Personalización avanzada* le permite definir colores de fondo diferentes para cada celda.
- Con *Texto extragrande* se define el comportamiento cuando el ancho establecido para una columna es menor que la longitud de su contenido. Puede ser **Ajustar texto** o **Truncar texto**.

> **Nota**
>
> Mas propiedades del elemento tabla fija fueron descritas en la sección *Funcionalidades generales de las tablas*.

Funcionalidades generales de las tablas

Mostrar cuadrícula

El grupo *Mostrar cuadrícula* de los elementos de tabla proporciona las siguientes funcionalidades (ver Figura 18.49):

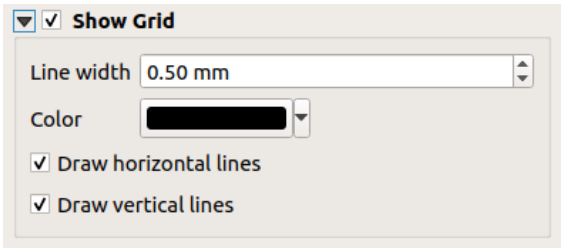


Figura 18.49: Grupo Mostrar cuadrícula de la Tabla de Atributos

- Activar *Mostrar cuadrícula* cuando desee mostrar la cuadrícula, los contornos de las celdas de la tabla. También puede seleccionar *Dibujar líneas horizontales*, *Dibujar líneas verticales* o ambos.
- Con *Ancho de línea* se puede establecer el espesor de las líneas utilizadas en la cuadrícula.
- El *Color* de la cuadrícula se puede configurar usando el widget de selección de color.

Estilo de Fuentes y Texto

El grupo *Estilo de fuentes y texto* de los elementos de tabla proporciona las siguientes funcionalidades (ver Figura 18.50):



Figura 18.50: Grupo Estilo de Fuentes y Texto de la Tabla de Atributos

- Puede definir las propiedades de *Fuente* para el *Encabezado de tabla* y *contenido de tabla*, usando el control avanzado *formato de texto* (con buffer, sombra, efectos de trazado, transparencia, fondo, coloreado, …). Note que estos cambios no afectan a las celas que tienen una fuente personalizada asignada, ya sea desde la sección de *Apariencia* o desde el diálogo *Diseñador de Tabla*. Solo las celdas con la representación predeterminada son sobreescritas.
- Para *Encabezado de tabla*, también puede establecer la *Alineación* en `Seguir alineación de columna` o anular esta configuración seleccionando `Izquierda`, `Centro` o `Derecha`. La alineación de columna se establece mediante el cuadro de diálogo *Seleccionar atributos* (véase Figura 18.44).

Marcos

El grupo *Marcos* del elemento de tabla proporciona las siguientes funcionalidades (ver Figura 18.51):

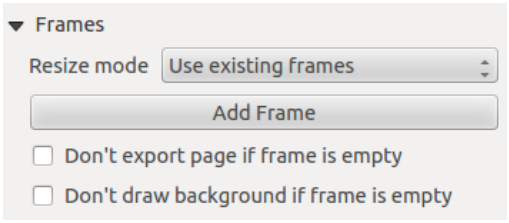


Figura 18.51: Grupo Marcos de la Tabla de Atributos

- Con *Modo de redimensionado* puede seleccionar cómo representar el contenido de la tabla de atributos:
  - `Usar marcos existentes` muestra el resultado en el primer marco y solo en los marcos agregados.
  - `Extender a la página siguiente` creará tantos marcos (y las páginas correspondientes) como sea necesario para mostrar la selección completa de la tabla de atributos. Cada cuadro se puede mover en el diseño. Si cambia el tamaño de un marco, la tabla resultante se dividirá entre los otros marcos. El último marco se recortará para adaptarse a la tabla.
  - `Repetir hasta finalizar` también creará tantos marcos como la opción *Extender a la página siguiente*, excepto que todos los marcos tendrán el mismo tamaño.
- Utilice el botón *Agregar marco* para agregar otro marco con el mismo tamaño que el marco seleccionado. El resultado de la tabla que no cabe en el primer marco continuará en el siguiente marco cuando utilice el modo de cambio de tamaño `Usar marcos existentes`.
- Activar *No exportar página si el marco está vacío* evita que la página se exporte cuando el marco de la tabla no tiene contenido. Esto significa que todos los demás elementos de diseño, mapas, barras de escala, leyendas, etc. no serán visibles en el resultado.
- Activar *No dibujar fondo si el marco está vacío* impide que el fondo que se puede extraer cuando el marco de la tabla no tiene contenido.

18.2.8 Los elementos de marcador, imagen y flecha de norte

- *El elemento Imagen*
  - *Propiedades principales*
  - *Tamaño y colocación*
  - *Rotación de imagen*
- *El elemento Flecha del Norte*
- *El elemento marcador*

Junto con los elementos del mapa o la leyenda en el diseño de impresión, es posible que desee decorar su realización con imágenes o anotaciones. QGIS proporciona diferentes herramientas para lograr esto:

- el *picture item*:decora el diseño con una imagen rasterizada o un archivo SVG (por ejemplo: logotipos, imágenes, flechas de norte, …)
- el *north arrow item*: un elemento de imagen predefinido con una imagen de flecha norte

- el elemento *marker item*: decora el diseño con el vector QGIS *symbols*. Se puede utilizar para colocar marcadores sobre un elemento del mapa o para la creación de leyendas personalizadas avanzadas.

El elemento Imagen

Puede agregar una imagen arrastrándola desde su administrador de archivos al lienzo, pegándola directamente en el diseño usando `Ctrl+V` o *Editar* ► *Pegar* y usando Add Picture, siguiendo las instrucciones de creación de elementos. Luego puedes manipularlo, como se explica en *Interactuando con los elementos de composición*.

Al usar Add Picture, el elemento de la imagen será un marco en blanco que puede personalizar usando su panel *Propiedades del elemento*. Además que los *items common properties*, este objeto tiene las siguientes funcionalidades:

Propiedades principales

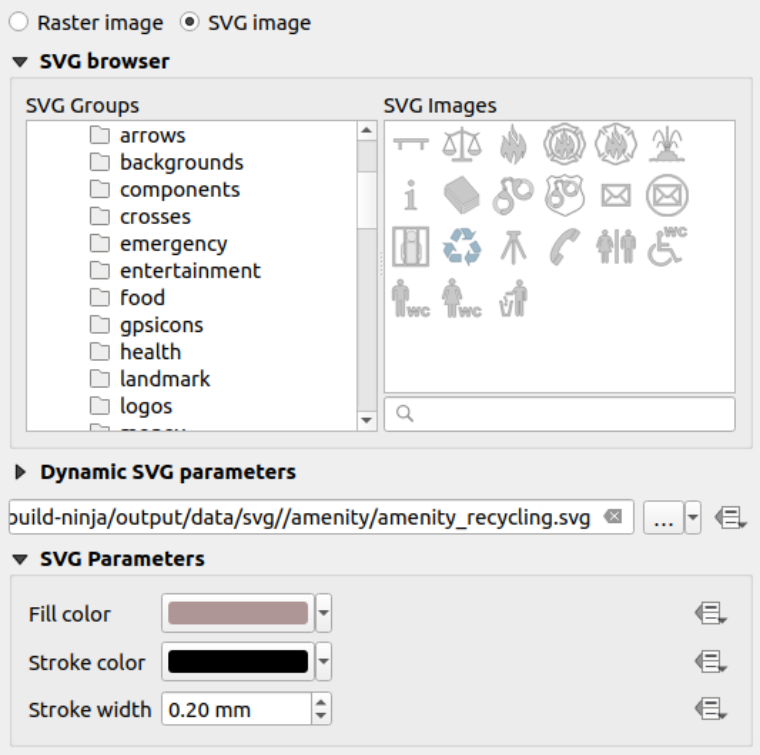


Figura 18.52: Panel Propiedades del Elemento Imagen

El elemento de imagen admite dos tipos de imágenes:

- *Imagen ráster*: se puede usar un widget de selección de archivos para obtener los datos. Use el botón … Borwse para seleccionar un archivo en su computadora o ingrese la ruta directamente en el campo de texto. Incluso puede proporcionar una URL remota que apunte a una imagen. La imagen asociada también se puede :ref:`embedded <embedded_file_selector>`en el diseño.

  Utilice el botón Suplantación definida por datos para establecer la fuente de la imagen a partir de un atributo de objeto o usando una expresión regular.

- *Imagen SVG*: utilizando por defecto las bibliotecas SVG proporcionadas en *Configuración* ► *Opciones* ► *Sistema* ► *Rutas SVG*. Sin embargo, puede utilizar cualquier otro archivo y la selección de archivos sigue las mismas reglas que para la imagen de trama. Los parámetros SVG también se pueden configurar de forma dinámica.

Los archivos `.SVG` de QGIS proporcionados (predeterminados) son personalizables, lo que significa que puede aplicar fácilmente otros *Color de relleno*, *Color de trazo* (incluida la opacidad) y *Anchura de trazo* al original, utilizando su función correspondiente en el grupo *Parámetros SVG*. Estas propiedades también pueden ser *definidas por datos*.

Si agrega un archivo `.SVG` que no habilita estas propiedades, es posible que deba agregar las siguientes etiquetas al archivo para agregar soporte, p.Ej. por transparencia:

- *fill-opacity=»param(fill-opacity)»*
- *stroke-opacity=»param(outline-opacity)»*

Más detalles en *SVG parametrizable*.

Nota

Arrastrar y soltar un archivo de imagen (ráster o SVG) en la página de diseño creará un elemento de imagen de diseño con la configuración correspondiente.

Tamaño y colocación



Figura 18.53: Tamaño de las imágenes de diseño y propiedades de ubicación

Con la opción *modo de redimensionado*, puede establecer cómo se muestra la imagen cuando se cambia el tamaño del marco:

- `Zoom`: amplía/reduce la imagen al marco mientras mantiene el ratio de aspecto de la imagen
- `Estirar`: estira la imagen para que quepa dentro del marco
- `Cortar`: use este modo solo para imágenes de trama, establece el tamaño de la imagen en el tamaño de la imagen original sin escalar, y el marco se usa para recortar la imagen. Por lo tanto, solo la parte de la imagen que está dentro del marco será visible.
- `Zoom y cambio de tamaño del marco`: amplía la imagen para que se ajuste al marco y luego cambia el tamaño del marco para que se ajuste a las dimensiones de la imagen resultante
- `Cambiar el tamaño del marco al tamaño de la imagen`: establece el tamaño del marco para que coincida con el tamaño original de la imagen (sin escala)

Dependiendo del *Modo de cambio de tamaño* seleccionado, las opciones de *Colocación* y *Rotación de imagen* pueden estar deshabilitadas. *Colocación* te permite seleccionar la posición de la imagen dentro de su marco (arriba/medio/abajo e izquierda/centro/derecha).

Rotación de imagen

Las imágenes se pueden rotar con el campo *Imagen rotada*. Activando el *Sync with map* la casilla sincroniza la rotación de la imagen con la rotación aplicada al elemento del mapa seleccionado. Esta es una característica conveniente para hacer que cualquier imagen se comporte como una flecha de norte. La *Alineación norte* puede ser:

- **Norte de cuadrícula**: la dirección de una línea de cuadrícula que es paralela al meridiano central de la cuadrícula nacional/local
- **Norte verdadero**: dirección de un meridiano de longitud.

También puede aplicar una declinación *Desplazamiento* a la rotación de la imagen.

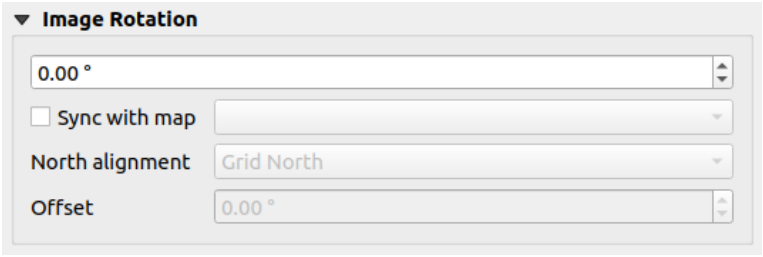


Figura 18.54: Propiedades de rotación de imagen de imágenes de diseño

El elemento Flecha del Norte

Puede agregar una flecha norte con el botón Agregar flecha norte, al lado de *instrucciones de creación de elementos* y manipularlo de la misma manera que se expone en *Interactuando con los elementos de composición*.

Dado que las flechas norte son imágenes, el elemento *Flecha norte* tiene las mismas propiedades que el *elemento de imagen*. Las principales diferencias son:

- Una flecha norte por defecto es usada al añadir el elemento, en lugar de un marco en blanco
- El elemento de la flecha norte se sincroniza con un elemento del mapa de forma predeterminada: la propiedad *Sincronizar con el mapa* es el mapa sobre el que se dibuja el elemento de la flecha norte. Si no hay ninguno, vuelve al *mapa de referencia*.

Nota

Muchas de las flechas del norte no tienen una “N” agregada en la flecha del norte. Esto se hace a propósito, ya que hay idiomas que no usan una “N” para el norte.

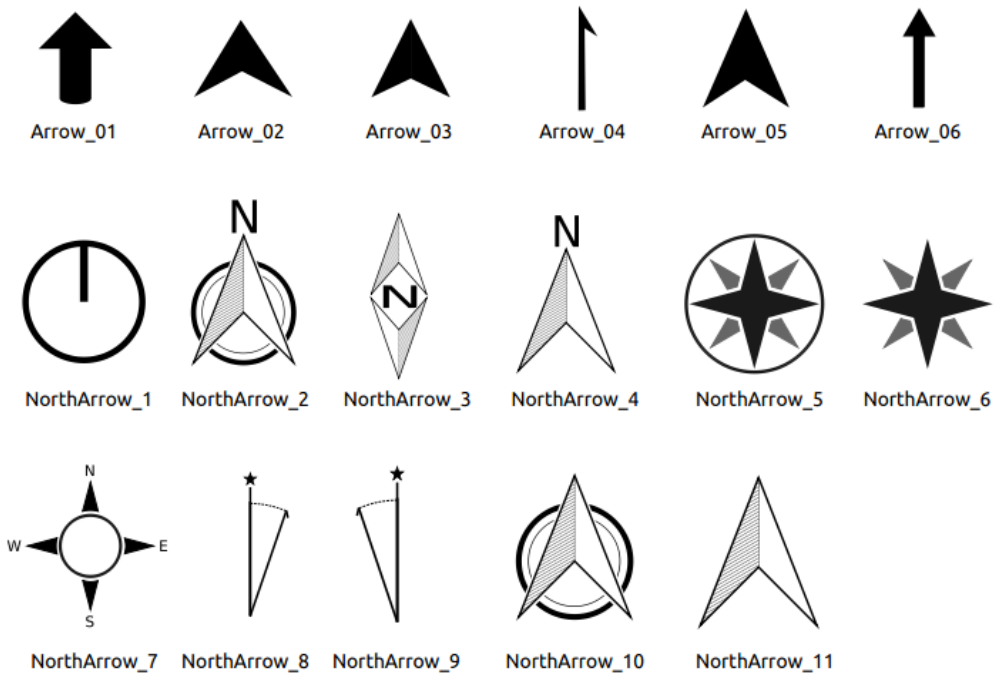


Figura 18.55: Las flechas de norte disponibles para la selección en la biblioteca SVG proporcionada.

El elemento marcador

Para agregar un elemento de marcador, seleccione el botón Add Marker y haga clic en la página. Se agrega un símbolo de marcador de punto predeterminado. Luego puedes manipularlo, como se explica en *Interactuando con los elementos de composición*. Pero tenga en cuenta que, a diferencia de la mayoría de los otros elementos, cambia el tamaño del elemento dado que su tamaño está controlado por las propiedades de los símbolos incrustados.

El elemento marcador se puede personalizar desde el panel *Propiedades del elemento*. Además de las:ref:*items common properties <item_common_properties>* , también puede:

- modificar el *Símbolo*, basándose en todo el símbolo *widget capabilities*
- sincroniza la rotación del elemento del marcador con la del mapa (ver *Rotación de imagen*), actuando como una flecha de norte. La rotación del mapa se agrega a cualquier rotación de nivel de símbolo de marcador existente (por ejemplo, si tiene que girar el marcador triangular 90 ° para que apunte hacia arriba, ¡aún funcionará bien en el modo de flecha norte!)

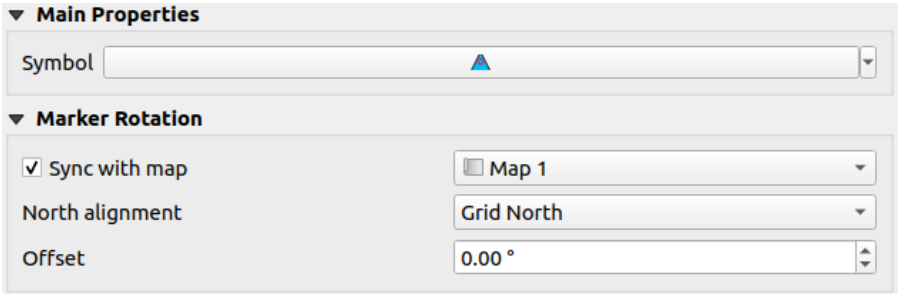


Figura 18.56: Las propiedades personalizadas del elemento de marcador

18.2.9 Elemento del perfil de elevación

El elemento Perfil de elevación se utiliza para mostrar una *vista perfil de elevación* en una composición. Utilice el botón  *Añadir perfil de elevación* y siga las *instrucciones de creación de elementos* para añadir un nuevo elemento de perfil de elevación que posteriormente podrá manipular del mismo modo que se demuestra en *Interactuando con los elementos de composición*.

Un nuevo elemento de perfil de elevación tiene una configuración predeterminada que hace que muestre un gráfico vacío. Puede personalizar sus propiedades en el panel *Propiedades del elemento*. Además de las *propiedades comunes*, esta prestación tiene las siguientes funcionalidades:

El panel Perfil de elevación *Propiedades del elemento* incorpora una barra de herramientas superior con las siguientes funcionalidades:

-  Actualizar perfil de elevación para actualizar la representación del elemento
-  Copiar desde perfil de elevación: un menú desplegable desde el que seleccionar una vista de perfil de elevación. La configuración de la vista se aplica al elemento del perfil de elevación del diseño y puede modificarse posteriormente.

Capas

En el grupo *Capas*, marque en la vista de árbol las capas que desea representar en el elemento de perfil. Recuerde configurar correctamente las propiedades *Elevación* de las capas seleccionadas.

Curva de perfil

-  *Controlado por atlas*: la *curva de perfil* se tomará de la prestación actual del atlas y la vista del perfil de elevación se actualizará a medida que recorra las prestaciones del atlas. Esta función es compatible actualmente con un atlas o informe de diseño activo que utilice una capa de cobertura de geometría lineal.
- La distancia *Tolerancia*, que puede definirse a partir de los datos, ayuda a controlar a qué distancia debe estar un objeto espacial de las capas visibles de la curva de perfil para que se muestre en la vista de alzado del trazado. Actualmente sólo se devuelven objetos espaciales punto.

Show Subsections Indicator

It is possible to highlight the position of the vertices of the curve on the Elevation Profile. Select the  *Show Subsections Indicator* to display vertical lines at the distances of the vertices. Use the *Lines* drop-down list to set the symbol, color, opacity and width of the Subsections Indicator lines.

Rangos de gráficos

Un elemento de perfil de elevación de diseño no muestra necesariamente toda la extensión de la vista de perfil de elevación en la que se basa. Se puede limitar el área a renderizar, siempre que:

- en el eje X, la *Distancia mínima* y la *Distancia máxima* desde el punto inicial de la curva de perfil.
- En el eje Y, la *Elevación mínima* y la *Elevación máxima*.

Ejes de distancia y elevación

Los grupos *Eje de distancia* y *Eje de elevación* ofrecen opciones para ajustar la cuadrícula sobre el elemento del perfil de elevación, respectivamente en los ejes X e Y:

- la visualización de la distancia *Unidad*, que permite anular la unidad del lienzo del mapa
- la graduación en el eje con un *Intervalo mayor* y un *Intervalo menor*.
- los símbolos de línea que se aplicarán a las correspondientes *Líneas de cuadrícula mayores* y *Líneas de cuadrícula menores*.
- la regularidad con la que deben etiquetarse los elementos de graduación (*Intervalo de etiqueta*), así como su *Formato de etiqueta* y *Fuente de etiqueta*.
- las *Etiquetas de distancia*: configuran si el símbolo de la unidad de distancia debe colocarse junto a *Cada valor*, *Primer valor*, *Último valor*, *Primeros y últimos valores*, o saltarse (*Ninguno*).

Área del gráfico

En *Área del gráfico*, puede configurar la representación del área en la que se muestra realmente el gráfico del perfil de elevación:

- un símbolo de relleno de *Fondo*
- un símbolo de línea de *Borde*
- los márgenes del borde del elemento del perfil de elevación

18.2.10 El elemento del marco HTML

Es posible agregar un marco que muestre el contenido de un sitio web o incluso crear y diseñar su propia página HTML y mostrarla. Puede agregar una imagen con  *Agregar HTML* siguiendo las *instrucciones de creación de elementos* y manipularlo de la misma manera que se expone en *Interactuando con los elementos de composición*. Tenga en cuenta que la escala HTML está controlada por la resolución de exportación del diseño en el momento en que se crea el marco HTML.

El elemento HTML se puede personalizar usando su panel *Propiedades del elemento*. Aparte de *propiedades comunes de los elementos*, esta prestación tiene las siguientes funcionalidades (ver Figura 18.57):

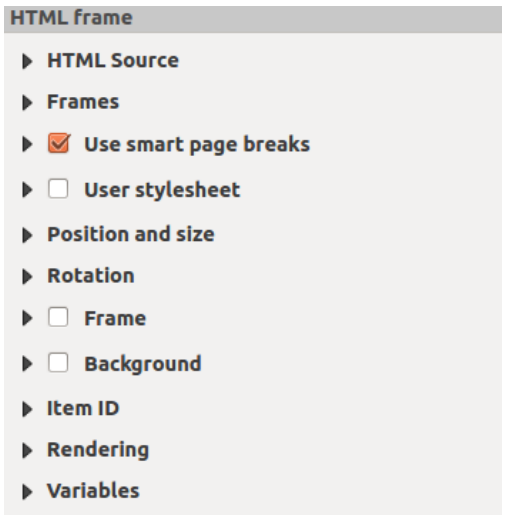


Figura 18.57: Marco HTML, el Panel de Propiedades de Elemento

Fuente HTML

El grupo *fuente HTML* del marco HTML *Propiedades del elemento* proporciona las siguientes funcionalidades (ver Figura 18.58):

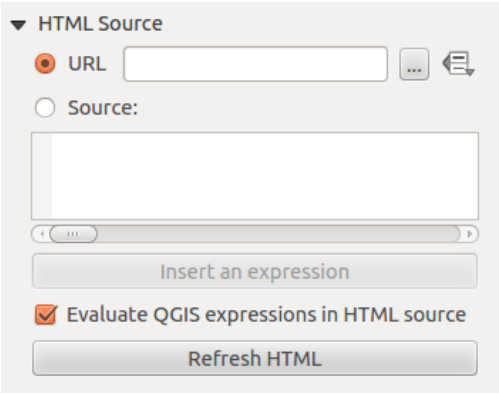


Figura 18.58: Marco HTML, las propiedades de Fuente HTML

- En *URL* puedes ingresar la URL de una página web que copiaste desde tu navegador de Internet o seleccionar un archivo HTML usando el botón … `*:sup:* Examinar`. También existe la opción de utilizar el botón Suplantación definida por datos, para proporcionar una URL del contenido de un campo de atributo de una tabla o usando una expresión regular.
- En *Fuente* se puede ingresar el texto en la caja de texto con algunas pestañas HTML o proporcionar una pagina completa de HTML.
- El botón *Insertar o editar una expresión…* se puede usar para agregar una expresión como `[% Año ($ ahora)%]` en el cuadro de texto Fuente para mostrar el año actual. Este botón sólo se activa cuando se selecciona el botón de radio *Fuente*. Después de insertar la expresión, haga click en algún lugar del cuadro de texto antes de actualizar el marco HTML; de lo contrario, perderá la expresión.
- Activar la expresión *Evaluar las expresiones QGIS en código HTML* para ver el resultado de la expresión que ha incluido, de lo contrario, verá la expresión en su lugar.
- Utilice el botón *Actualizar HTML* para actualizar el marco(s) HTML y ver el resultado de los cambios.

Marcos

El grupo *Marcos* del marco HTML *Propiedades de elemento* proporciona las siguientes funcionalidades (ver :numref: *figure_layout_html_frames*):

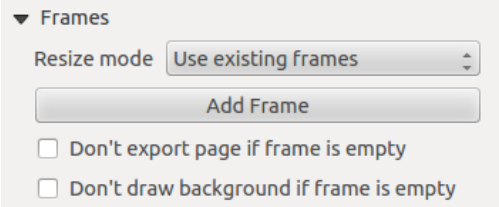


Figura 18.59: Marco HTML, las Propiedades del Marco

- Con *Modo cambio de tamaño* se puede seleccionar cómo renderizar el contenido HTML:
  - `Usar marcos existentes` muestra el resultado en el primer marco y solo en los marcos agregados.
  - `Extender a la página siguiente` creará tantos marcos (y páginas correspondientes) como sea necesario para representar la altura de la página web. Cada cuadro se puede mover en el diseño. Si

cambia el tamaño de un marco, la página web se dividirá entre los otros marcos. El último marco se recortará para que se ajuste a la página web.

  - `Repetir en todas las páginas` repetirá la parte superior izquierda de la página web en cada página en marcos del mismo tamaño.
  - `Repetir hasta finalizar` también creará tantos marcos como la opción `Extender a la página siguiente`, excepto que todos los marcos tendrán el mismo tamaño.

- Utilice el botón *Añadir marco* para agregar otro marco con el mismo tamaño que el marco seleccionado. Si la página HTML no cabe en el primer fotograma, continuará en el siguiente cuando utilice *Modo de redimensionado* o *Usar marcos existentes*.
- Activar *No exportar página si el marco está vacío* evita que la página se exporte cuando el marco no tiene contenido HTML. Esto significa que todos los demás elementos de diseño, mapas, barras de escala, leyendas, etc.no serán visibles en el resultado.
- Activar *No dibujar el fondo si el marco está vacío* impide que el marco HTML sea dibujado si el marco está vacío.

Usar saltos de página inteligentes y hoja de estilos de usuario

El cuadro de diálogo *Usar saltos de página inteligentes* y *Hoja de estilo del usuario* del marco HTML *Propiedades del elemento* proporciona las siguientes funcionalidades (ver :numref:` figure_layout_html_breaks`):

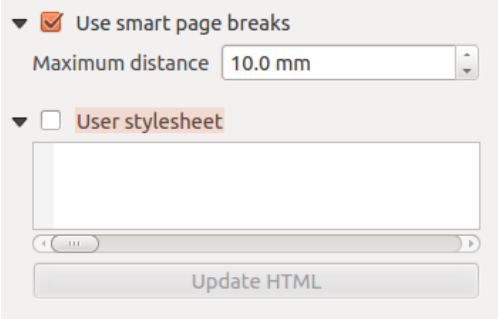


Figura 18.60: Marco HTML, uso de saltos de página inteligentes y propiedades de la hoja de estilo de usuario

- Activar *Usar saltos de página inteligentes* para impedir para evitar que el contenido del marco html de romper a mitad de camino una línea de texto por lo que continúa agradable y sencillo en el cuadro siguiente.
- Establecer la *Máxima distancia* permite en el cálculo de dónde colocar los saltos de página en el html. Esta distancia es la cantidad máxima de espacio vacío permitido en la parte inferior de un marco después de calcular la ubicación óptima de descanso. Ajustar un valor más grande resultará en una mejor elección de la ubicación de salto de página, pero más espacio perdido en la parte inferior de los marcos. Esto sólo se utiliza cuando *Usar saltos de página inteligentes* está activado.
- Active *Hoja de estilo del usuario* para aplicar estilos HTML que a menudo se proporcionan en hojas de estilo en cascada. A continuación, se proporciona un ejemplo de código de estilo para establecer el color de la etiqueta de encabezado `<h1>` en verde y establecer la fuente y el tamaño de fuente del texto incluido en las etiquetas de párrafo``<p>``.

```
h1 {color: #00ff00;
}
p {font-family: "Times New Roman", Times, serif;
   font-size: 20px;
}
```

- Utilice el botón *Actualizar HTML* para ver el resultado de la configuración de la hoja de estilo.

18.2.11 Los elementos de Forma

QGIS proporciona un par de herramientas para dibujar formas regulares o más complejas sobre la composición de impresión.

> **Nota**
>
> A diferencia de otros elementos del compositor de impresión, no puede aplicar estilo al marco ni al color de fondo de las formas que delimitan el marco (establecido en transparente de forma predeterminada).

El elemento Forma Regular

El elemento *Forma* es una herramienta que ayuda a decorar su mapa con formas regulares como triángulo, rectángulo, elipse… Puede agregar una forma regular usando la Herramienta Añadir Forma que da acceso a herramientas particulares como Añadir Rectángulo, Añadir Elipse y Añadir triángulo. Una vez que haya seleccionado la herramienta adecuada, puede dibujar el siguiente elemento siguiendo las *instrucciones de creación de elementos*. Al igual que otros elementos de diseño, una forma regular se puede manipular de la misma manera que se expone en *Interactuando con los elementos de composición*.

> **Nota**
>
> Mantener presionada la tecla `Shift` mientras dibuja la forma básica con el método de hacer clic y arrastrar le ayuda a crear un cuadrado, círculo o triángulo perfecto.

La forma predeterminada para el objeto puede ser personalizada usando su panel *Propiedades de elemento*. Aparte de las *Propiedades comunes de objetos*, esta prestación tiene las siguientes funcionalidades (ver Figura 18.61):

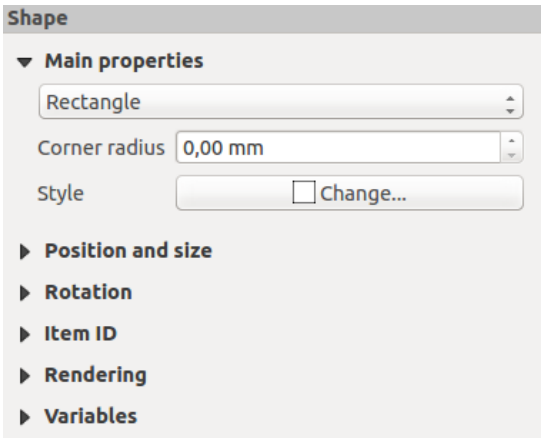


Figura 18.61: Panel Propiedades del Elemento Forma

El grupo *Propiedades principales* muestra y le permite cambiar el tipo de elemento de forma (**Elipse**, **Rectángulo** o **Triángulo**) dentro del marco dado.

Puede establecer el estilo de la forma usando el wifget avanzado *símbolo* y el selector de *color*…

Para la forma de rectángulo, puede establecer en diferentes unidades el valor de *Radio de la esquina* para redondear las esquinas.

Los elementos de forma basados en nodos

Mientras que la herramienta *Añadir Forma* proporciona una forma de crear un elemento geométrico simple y predefinido, la herramienta *Agregar elemento de nodo* le ayuda a crear un elemento geométrico personalizado y más avanzado. Para polilíneas o polígonos, puede dibujar tantas líneas o lados como desee y los vértices de los elementos se pueden manipular de forma independiente y directa mediante *Editar elemento de nodos*. El elemento en sí se puede manipular como se expone en *Interactuando con los elementos de composición*.

Para agregar una forma basada en nodos:

1. Click en el icono Añadir elemento nodo
2. Seleccione la herramienta Añadir polígono o Agregar polilínea
3. Realice clics izquierdos consecutivos para agregar nodos de su artículo. Si mantiene presionada la tecla `Shift` mientras dibuja un segmento, está restringido a seguir una orientación múltiplo de 45|degrees|.
4. Cuando hayas terminado, haz clic derecho para terminar la forma.

Puede personalizar la apariencia de la forma en el panel *Propiedades del elemento*.

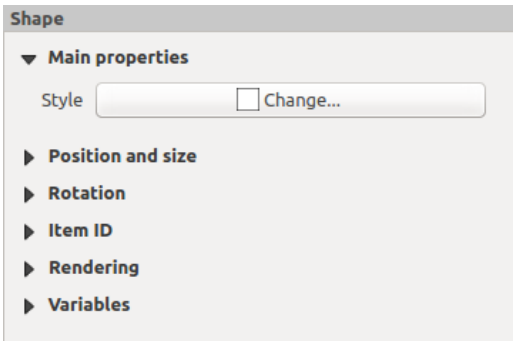


Figura 18.62: Panel de propiedades de elemento de forma de polígono de nodos

En *Propiedades principales*, puede establecer el estilo de la forma usando el widget avanzado símbolo…

Para elementos de nodo de polilínea, también puede parametrizar *Marcadores de línea*, es decir, agregar:

- marcadores de inicio y/o fin con opciones:
  - *Ninguno*: dibuja una polilínea simple.
  - *Flecha*: agrega una punta de flecha triangular regular que puede personalizar.
  - marcador *SVG*: usa un archivo `SVG` como punta de flecha del elemento.
- personalizar la punta de la flecha:
  - *Color del borde de la flecha*: establece el color de trazo de la punta de la flecha.
  - *color de relleno de flecha*: establece el color de relleno de la punta de la flecha.
  - *Anchura del borde de la flecha*: establece el ancho del trazo de la punta de la flecha.
  - *Anchura de la punta de la flecha*: establece el tamaño de la punta de la flecha.

Las imágenes SVG se rotan automáticamente con la línea. Los colores de trazo y relleno de las imágenes SVG predefinidas de QGIS se pueden cambiar usando las opciones correspondientes. SVG personalizado puede requerir algunas etiquetas siguiendo esta *instrucción*.

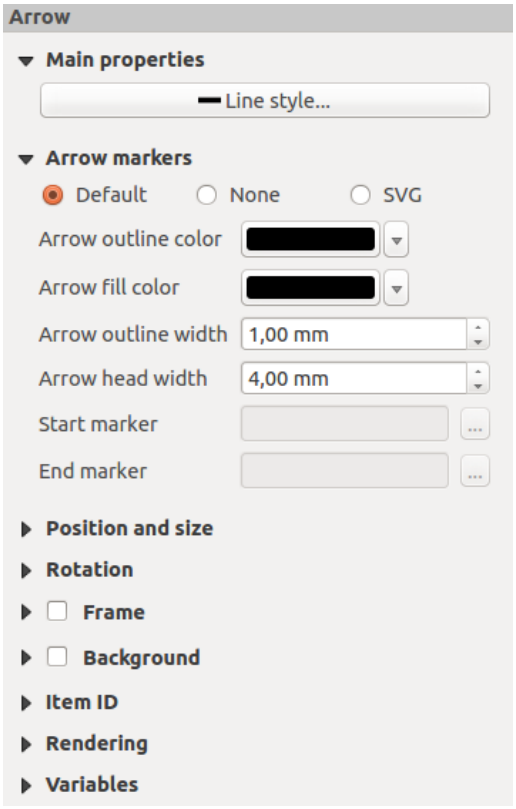


Figura 18.63: Panel de propiedades de elemento de forma de polilínea de nodos

El elemento flecha

La herramienta Agregar flecha es un atajo para crear una polilínea habilitada para flechas de forma predeterminada y, por lo tanto, tiene las mismas propiedades y comportamiento que *elemento nodo de polilínea*.

En realidad, el elemento de flecha se puede utilizar para agregar una flecha simple, por ejemplo, para mostrar la relación entre dos elementos de diseño de impresión diferentes. Sin embargo, para crear una flecha de norte, debe considerarse en primer lugar el elemento de imagen, ya que da acceso a un conjunto de flechas de norte en formato `.SVG` que puede sincronizar con un elemento de mapa para que gira automáticamente con él.

Editar la geometría de un elemento de nodo

Se proporciona una herramienta específica para editar formas basadas en nodos mediante Editar elemento de nodos. Dentro de este modo, puede seleccionar un nodo haciendo clic en él (se muestra un marcador en el nodo seleccionado). Un nodo seleccionado se puede mover arrastrándolo o usando las teclas de flecha. Además, en este modo, puede agregar nodos a una forma existente: haga doble clic en un segmento y se agregará un nodo en el lugar en el que haga clic. Finalmente, puede eliminar el nodo seleccionado actualmente presionando la tecla `Del`.

18.3 Crear salida

Figura 18.64 muestra un ejemplo de diseño de impresión que incluye todos los tipos de elementos de diseño descritos en la sección anterior.

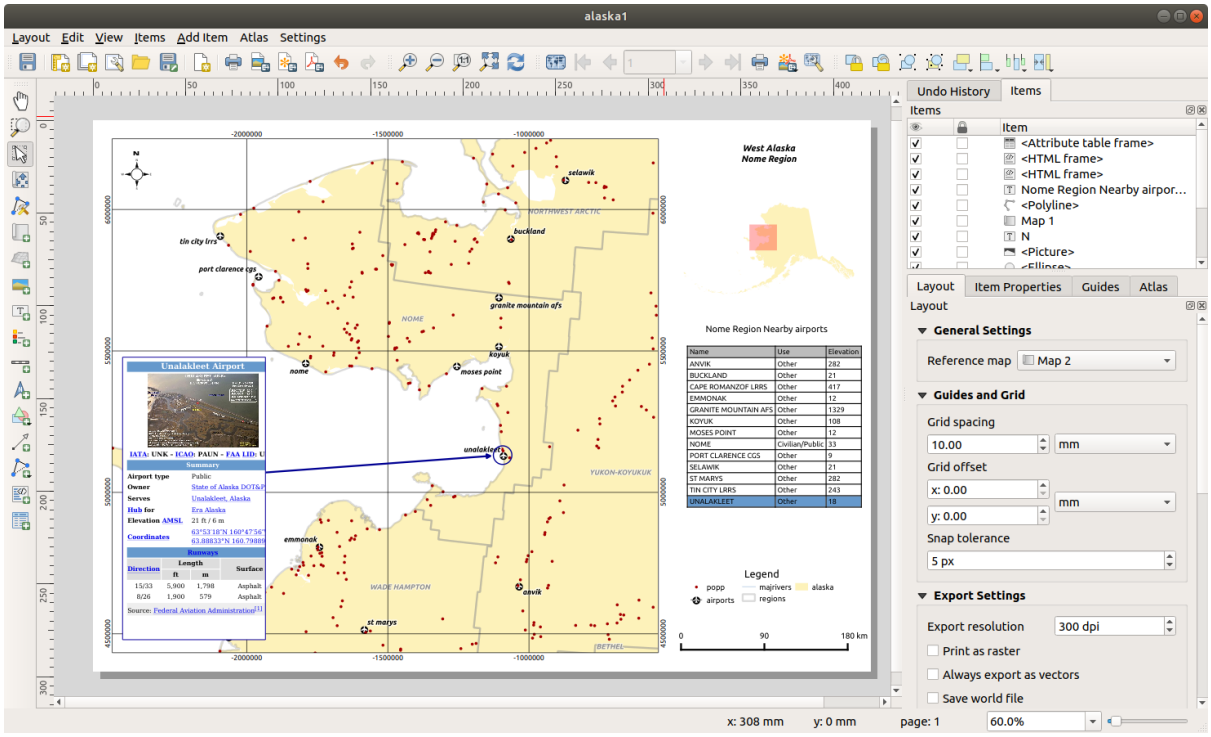


Figura 18.64: Diseño de impresión con vista de mapa, leyenda, imagen, barra de escala, coordenadas, texto y marco HTML agregado

Desde el menú *Diseño* o barra de herramientas, puede generar el diseño de impresión en diferentes formatos de archivo, y es posible modificar la resolución (calidad de impresión) y el tamaño del papel:

- El icono Imprimir le permite imprimir el diseño en una impresora conectada o en un archivo PostScript, según los controladores de impresora instalados.
- El icono Exportar como imagen exporta los formatos de imagen de diseño de impresión como :file:` PNG`, `BMP`, :file:` TIF`, `JPG`, y muchos otros. .
- El icono Exportar como SVG guarda el diseño de impresión como un `SVG` (Gráfico vectorial escalable).
- El icono Exportar como PDF guarda el diseño de impresión definido directamente como un archivo `PDF` (Formato de documento portátil).

18.3.1 Configuración de exportación

Siempre que exporta un diseño de impresión, hay una selección de configuraciones de exportación que QGIS debe verificar para producir la salida más adecuada. Estas configuraciones son:

- La *Configuración de exportación* del panel *Diseño*, como *Resolución de exportación*, *Imprimir como raster* *Exportar siempre como vectoriales* o :guilabel:` Guardar archivo del georreferenciación`
- *Excluir la página de las exportaciones* en el panel *propiedades del elemento de la página*
- *Excluir elemento de las exportaciones* en el panel *propiedades del elemento*

Además, se aplican automáticamente al diseño una serie de comprobaciones predefinidas. Actualmente, estas comprobaciones incluyen la comprobación de que las barras de escala están correctamente vinculadas a los elementos del mapa, y que los elementos de la vista general del mapa también están correctamente vinculados a un mapa. Si las comprobaciones fallan, se muestra una bonita advertencia que le informa del problema.

18.3.2 Exportar como imagen

Para exportar una composición como imagen:

1. Click en el icono Exportar como imagen
2. Seleccione el formato de la imagen, la carpeta y el nombre del archivo (por ejemplo `myill.png`) para usar. Si el diseño contiene más de una página, cada página se exportará a un archivo con el nombre de archivo dado con el número de página adjunto (por ejemplo `myill_2.png`).
3. En el siguiente diálogo (*Opciones de exportación de imágenes*) :
   - Puede suplantar la *Resolución de exportación* del diseño de impresión y las dimensiones de la página exportada (como se establece en el panel *Diseño*).
   - For JPEG and JPG image formats, you can set the *JPEG Quality* of the output image.
   - La representación de imágenes también se puede mejorar con la opción *Activar antialiasing*.
   - Si desea exportar su diseño como una **imagen georreferenciada** (por ejemplo, para compartir con otros proyectos), marque la opción *Generar archivo de georreferenciación* y un *Archivo georreferenciado ESRI * con el mismo nombre que la imagen exportada, pero con una extensión diferente (`.tfw` para TIFF,:file:.*pnw* para PNG ,:file:*jgw* para JPEG, …) se creará al exportar. Esta opción también se puede marcar por defecto en el *panel de diseño*.

     Nota

     Para la salida de varias páginas, solo la página que contiene el *mapa de referencia* obtendrá un archivo mundial (asumiendo que la opción *Generar archivo de georreferenciación* está marcada).

   - Marcando la opción *Recortar al contenido*, la imagen generada por el diseño incluirá el área mínima que encierra todos los elementos (mapa, leyenda, barra de escala, formas, etiqueta, imagen …) de cada página de la composición:
     - Si la composición incluye una sola página, se cambia el tamaño de la salida para incluir TODO en la composición. Luego, la página se puede reducir o extender a todos los elementos según su posición (en, arriba, abajo, izquierda o derecha de la página).
     - En el caso de un diseño de varias páginas, se cambiará el tamaño de cada página para incluir elementos en su área (lados izquierdo y derecho para todas las páginas, más arriba para la primera página e inferior para la última página). Cada página redimensionada se exporta a un archivo separado.

     El cuadro de diálogo *Recortar al contenido* también le permite agregar márgenes alrededor de los límites recortados.

- Marcando *Abrir archivo después de exportar* el archivo exportado se abrirá automáticamente en el visor de imágenes Predeterminado.

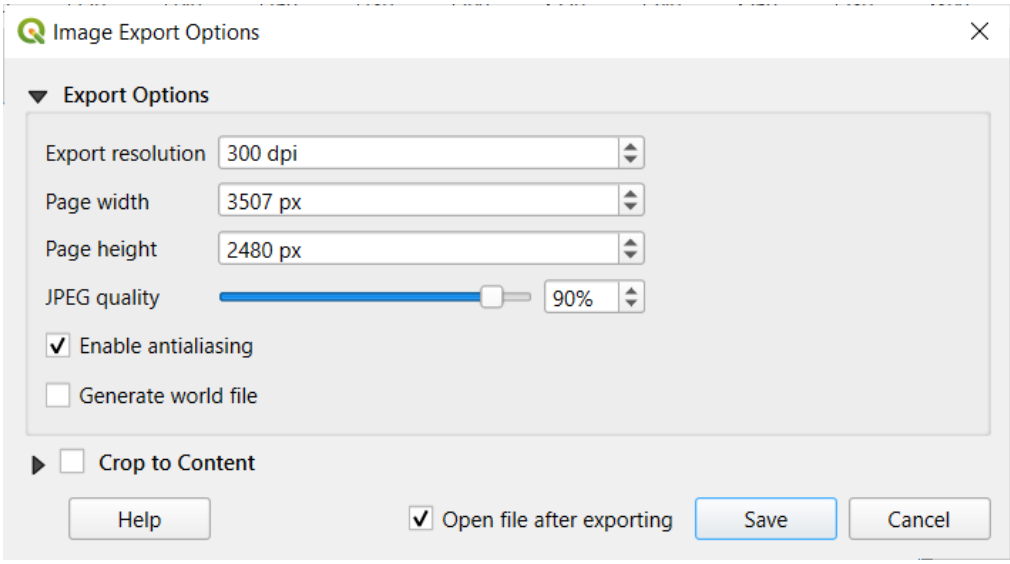


Figura 18.65: Opciones de exportación de imágenes

Truco

Utilice formatos de imagen que admitan transparencia cuando los elementos se extienden más allá de la extensión del papel

Los elementos de diseño pueden colocarse fuera de la extensión del papel. Al exportar con la opción *Recortar al contenido*, la imagen resultante puede extenderse más allá de la extensión del papel. Dado que el fondo fuera de la extensión del papel será transparente, para los formatos de imagen que no admiten la transparencia (por ejemplo, `BMP` y `JPG`), el fondo transparente se representará en negro completo, «corrompiendo» la imagen. Utilice formatos compatibles con transparencias (por ejemplo, `TIFF` y `PNG`) en tales casos.

Nota

Cuando sea compatible con el formato (por ejemplo `PNG`) y la biblioteca Qt subyacente, la imagen exportada puede incluir :ref:` metadatos del proyecto <project_metadata>`(autor, título, fecha, descripción …)

18.3.3 Exportar como SVG

Para exportar un diseño como SVG:

1. Haga clic en el botón del icono Exportar como SVG
2. Complete la ruta y el nombre del archivo (se usa como nombre base para todos los archivos en el caso de una composición de varias páginas, como para la exportación de imágenes)
3. En las siguientes *Opciones de exportación de SVG*, puede anular el diseño predeterminado *ajustes de exportación* o configurar unos nuevos:
   - *Exportar capas de mapas como grupos SVG*: los elementos exportados se agrupan dentro de capas cuyo nombre coincide con los nombres de capa de QGIS, lo que facilita la comprensión del contenido del documento.

- *Exportar siempre como vectores*: algunas opciones de representación requieren que los elementos se rastericen para una mejor representación. Marque esta opción para mantener los objetos como vectores con el riesgo de que la apariencia del archivo de salida no coincida con la vista previa del diseño de impresión (para más detalles, consulte *Configuración de exportación*).
- *Exportar metadatos RDF* del documento, como el título, autor, fecha, descripción…
- *Simplifique las geometrías para reducir el tamaño del archivo de salida*: esto evita exportar TODOS los vértices de la geometría, lo que puede resultar en un tamaño de archivo de exportación ridículamente complejo y grande que podría fallar al cargar en otras aplicaciones. Las geometrías se simplificarán al exportar el diseño para eliminar cualquier vértice redundante que no sea discerniblemente diferente en la resolución de exportación (por ejemplo, si la resolución de exportación es `300 ppp`, vértices a distancias inferiores a `1/600 pulgadas` serán eliminados).
- Establecer la *Exportación de texto*: controla si las etiquetas de texto se exportan siempre o preferentemente como *objetos texto o contorno.*.
- Aplicar *Recortar al contenido opción*
- *Desactivar las exportaciones de capas ráster teseladas*: Al exportar archivos, QGIS utiliza una representación en mosaico de capa ráster incorporada que ahorra memoria. A veces, esto puede causar «vetas» visibles en los rásteres de los archivos generados. Marcar esta opción solucionaría eso, a costa de un mayor uso de memoria durante las exportaciones.
- Marcando *Abrir archivo tras exportar* el archivo exportado se abrirá automáticamente en el visor SVG predeterminado.

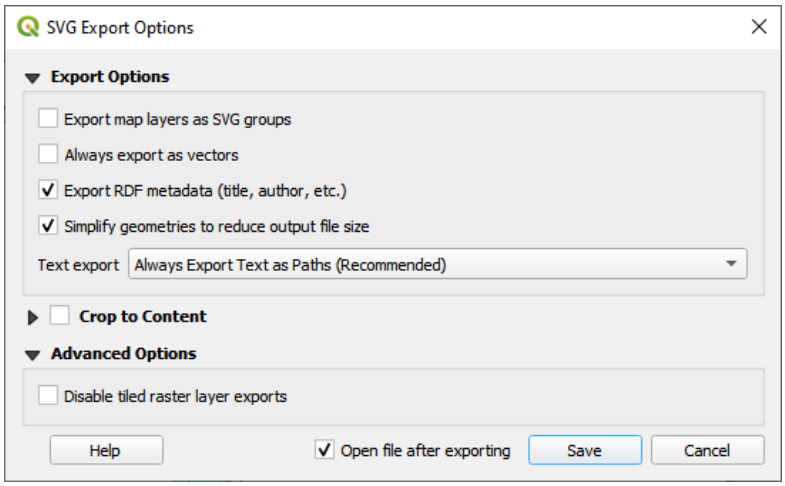


Figura 18.66: Opciones de Exportación de SVG

Nota

Actualmente, la salida SVG es muy básica. Este no es un problema en QGIS, pero un problema con la biblioteca Qt subyacente. Esperamos que esto se solucione en futuras versiones.

18.3.4 Exportar como PDF

Para exportar un diseño como PDF:

1. Click en el icono  Exportar como PDF
2. Complete la ruta y el nombre del archivo: a diferencia de la exportación de imágenes y SVG, todas las páginas del diseño se exportan a un solo archivo PDF.
3. En el siguiente cuadro de diálogo *Opciones de exportación de PDF*, puede anular el diseño predeterminado :ref:`configuración de exportación <layout_export_settings>`o configurar unos nuevos:

   - ☐ *Exportar siempre como vectores*: algunas opciones de representación requieren que los elementos se rastericen para una mejor representación. Marque esta opción para mantener los objetos como vectores con el riesgo de que la apariencia del archivo de salida no coincida con la vista previa del diseño de impresión (para más detalles, consulte *Configuración de exportación*).

     > **Nota**
     >
     > **Configuración de la opacidad de las capas vectoriales**
     >
     > Cuando se establece una opacidad para toda la capa (u opacidad para cualquier simbología vectorial) inferior al 100%, se fuerza una renderización rasterizada de esos objetos durante la exportación a PDF. Dependiendo del tamaño de la capa, esto puede aumentar considerablemente el tamaño del archivo PDF.

   - ☑ *Anexar información de georeferencia*: disponible sólo si el *reference map*, del que se toma la información, está en la primera página.
   - ☑ *Exportar metadatos RDF* del documento, como el título, autor, fecha, descripción…
   - Establecer la *Exportación de texto*: controla si las etiquetas de texto se exportan siempre o preferentemente como *objetos texto o contorno.*.
   - Controle la *compresión de imagen* PDF usando:
     - *Pérdida (JPEG)*, el cuál es el modo de compresión predeterminado
     - o *Sin pérdida*, que crea archivos más grandes en la mayoría de los casos, pero es mucho más adecuado para impresiones o para postproducción en aplicaciones externas (requiere Qt 5.13 o posterior).
   - ☐ *Crear PDF geoespacial*: Genera un archivo PDF georreferenciado.
   - ☐ *Desactivar las exportaciones de capas ráster teseladas*: Al exportar archivos, QGIS utiliza una representación basada en mosaicos que ahorra memoria. A veces, esto puede causar «vetas» visibles en los rásteres de los archivos generados. Marcar esta opción solucionaría eso, a costa de un mayor uso de memoria durante las exportaciones.
   - ☑ *Simplificar geometrías para reducir el tamaño del archivo de salida*: Las geometrías se simplificarán al exportar el diseño al eliminar los vértices que no sean perceptiblemente diferentes en la resolución de exportación (por ejemplo, si la resolución de exportación es `300 ppp`, vértices que están a distancia menor de `1/600 de pulgadas` se eliminarán). Esto puede reducir el tamaño y la complejidad del archivo de exportación (los archivos muy grandes pueden fallar al cargarse en otras aplicaciones).
   - Marcando ☑ *Abrir archivo después de exportar* el archivo exportado se abrirá automáticamente en el visor de PDF Predeterminado.

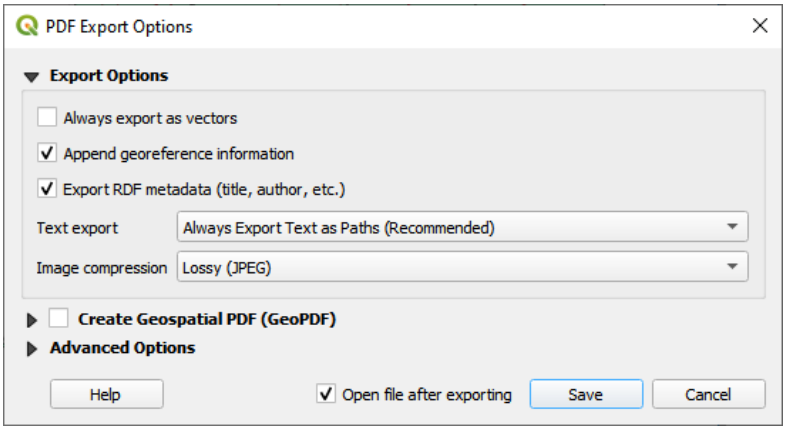


Figura 18.67: Opciones de exportación a PDF

Nota

Se admite la exportación a PDF geoespacial y se dispone de varias opciones específicas para PDF geoespacial:

- *Formato* (Formato PDF geoespacial - existen algunas variaciones),
- *Incluir múltiples temas de mapa* (especificar los temas de mapa a incluir),
- :guilabel:‘Incluir información del objeto vectorial` (elija las capas y agrúpelas en grupos de PDF lógicos).

Nota

La exportación de un diseño de impresión a formatos que admiten georreferenciación (por ejemplo, `PDF` y `TIFF`) crea una salida georreferenciada de forma predeterminada.

Nota

When exporting maps or layouts to vector formats such as PDF or SVG, line pattern fills are rendered using a line-by-line vector-based approach instead of rasterized tiles. This change results in smaller and higher-quality output files, with individual pattern lines remaining fully editable in external vector graphics software.

18.3.5 Generar un Atlas

Las funciones de Atlas le permiten crear libros de mapas de forma automatizada. Atlas usa los objetos de una tabla o capa vectorial (*capa de cobertura*) para crear una salida para cada objeto (**objeto atlas**) en la tabla/capa. El uso más común es acercar un elemento del mapa a la función del atlas actual. Otros casos de uso incluyen:

- un elemento de mapa que muestra, para otra capa, solo entidades que comparten el mismo atributo que la entidad atlas o que están dentro de su geometría.
- una etiqueta o elemento HTML cuyo texto se reemplaza a medida que los objetos se repiten
- un elemento de la tabla que muestra los atributos de los objetos asociados :ref:”padre o hijo<vector_relations>” del objeto actual del atlas …

Para cada objeto, la salida se procesa para todas las páginas y elementos de acuerdo con su configuración de exportación.

> **Truco**
>
> **Utilizar variables para mayor flexibilidad**
>
> QGIS proporciona un gran panel de funciones y *variables*, incluidas las relacionadas con el atlas, que puede utilizar para manipular los elementos de diseño, pero también la simbología de las capas, según el estado del atlas. La combinación de estas funciones le brinda mucha flexibilidad y le ayuda a producir fácilmente mapas avanzados.

Para habilitar la generación de un atlas y acceder a los parámetros del atlas, consulte el panel *Atlas*. Este panel contiene lo siguiente (ver Figura 18.68):

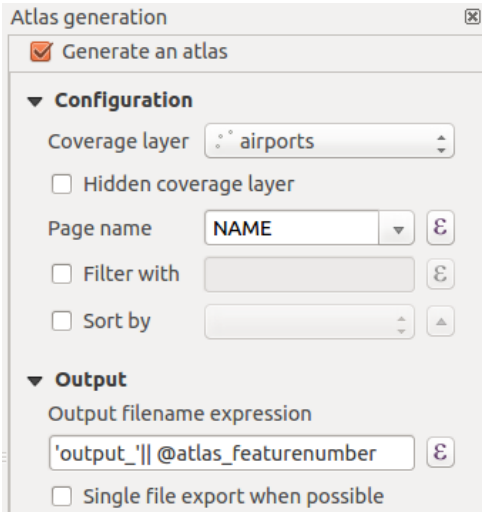


Figura 18.68: Panel Atlas

- *Generar atlas* habilita o deshabilita la generación de atlas.
- *Configuración*
  - Una *capa de cobertura* cuadro combinado que le permite elegir la tabla o capa vectorial que contiene los objetos sobre los que iterar.
  - Un opcional *Ocultar capa de cobertura* que, si está marcada, ocultará la capa de cobertura (pero no las otras capas) durante la generación.
  - Un cuadro combinado opcional *Nombre de la página* para especificar el nombre de la página(s) de objetos. Puede seleccionar un campo de la capa de cobertura o establecer una *expresión*. Si esta opción está vacía, QGIS utilizará una ID interna, de acuerdo con el filtro y / o el orden de clasificación aplicado a la capa.
  - Un Área de texto opcional *Filtrar con* que le permite especificar una expresión para filtrar entidades de la capa de cobertura. Si la expresión no está vacía, solo se procesarán los objetos que se evalúen como `Verdadero`.
  - Una opcional *Ordenar por* que le permite ordenar las entidades de la capa de cobertura (y la salida), usando un campo de la capa de cobertura o una expresión. El orden de clasificación (ascendente o descendente) se establece mediante el botón de dos estados *Dirección de clasificación* que muestra una flecha hacia arriba o hacia abajo.
- *Salida* - aquí es donde se puede configurar la salida del atlas:
  - Un cuadro de texto :guilabel:‘fichero expresión de salida` que se usa para generar un nombre de archivo para cada objeto del atlas. Se basa en expresiones. es significativo solo para renderizar en varios archivos.
  - A *Exportar fichero simple cuando sea posible* que le permite forzar la generación de un solo archivo si esto es posible con el formato de salida elegido (`PDF`, por ejemplo). Si este campo está marcado, el

valor del campo *fichero de expresión de salida* no tiene sentido.

- Una lista desplegable *Formato de exportación de imagen* para seleccionar el formato de salida cuando se utiliza el botón  Exportar atlas como imágenes… .

Controlar mapa por Atlas

El uso más común de atlas es con el elemento del mapa, acercándose a la función del atlas actual, a medida que la iteración pasa por la capa de cobertura. Este comportamiento se establece en las propiedades del grupo *Controlado por atlas* del elemento del mapa. Consulte control_atlas para conocer las diferentes configuraciones que puede aplicar en el elemento del mapa.

Personalizar etiquetas por expresión

Para adaptar las etiquetas al objeto sobre el que itera el atlas, puede incluir expresiones. Asegúrese de colocar la parte de la expresión (incluidas las funciones, campos o variables) entre `[%` y `%]` (consulte *El elemento etiqueta* para más detalles).

Por ejemplo, para una capa de ciudad con campos `CITY_NAME` y `ZIPCODE`, podrías insertar esto:

```
The area of [% concat( upper(CITY_NAME), ',', ZIPCODE, ' is ',
format_number($area/1000000, 2) ) %] km2
```

o, otra combinación:

```
The area of [% upper(CITY_NAME)%],[%ZIPCODE%] is
[%format_number($area/1000000,2) %] km2
```

La información `[% concat( upper(CITY_NAME), ',', ZIPCODE, ' is ', format_number($area/1000000, 2) ) %]` es una expresión que se usa dentro de la etiqueta. Ambas expresiones darían como resultado el siguiente tipo de etiqueta en el atlas generado:

```
The area of PARIS,75001 is 1.94 km2
```

Explorar los botones de suplantación definida por datos con atlas

Hay varios lugares donde puede utilizar el botón  Suplantación definida por datos para suplantar la configuración seleccionada. Esto es particularmente útil con la generación de atlas. Consulte *Configuración de anulación definida por datos* para obtener más detalles sobre este widget.

Para los siguientes ejemplos, la capa `Regions` del conjunto de datos de muestra de QGIS se usa y se selecciona como *capa de cobertura* para la generación del atlas. Suponemos que es un diseño de página única que contiene un elemento de mapa y un elemento de etiqueta.

Cuando la altura (norte-sur) de la extensión de una región es mayor que su ancho (este-oeste), debe utilizar la orientación *Vertical* en lugar de *Horizontal* para optimizar el uso del papel. Con un botón  Suplantación definida por datos puede establecer dinámicamente la orientación del papel.

Haga clic derecho en la página y seleccione *Propiedades de la página* para abrir el panel. Queremos establecer la orientación dinámicamente, usando una expresión dependiendo de la geometría de la región, así que presione el botón de campo  *Orientación*, seleccione *Editar … `para abrir el diálogo :guilabel:* Generador de cadenas de expresión` e ingrese la siguiente expresión:

```
CASE WHEN bounds_width(@atlas_geometry) > bounds_height(@atlas_geometry)
THEN 'Landscape' ELSE 'Portrait' END
```

Ahora bien, si previsualiza el atlas, el documento se orienta automáticamente, pero la ubicación de los elementos puede no ser la ideal. Para cada región, también debe cambiar la ubicación de los elementos de diseño. Para el elemento del mapa, puede utilizar el botón de la propiedad *Anchura* para establecerla dinámica usando la siguiente expresión:

```
@layout_pagewidth - 20
```

Del mismo modo, utilice el botón |dataDefine | de la propiedad *Altura* para proporcionar la siguiente expresión para restringir el tamaño del elemento del mapa:

```
@layout_pageheight - 20
```

Para asegurarse de que el elemento del mapa esté centrado en la página, establezca su *Punto de referencia* en el botón de opción superior izquierdo e ingrese `10` para sus posiciones *X* e *Y*.

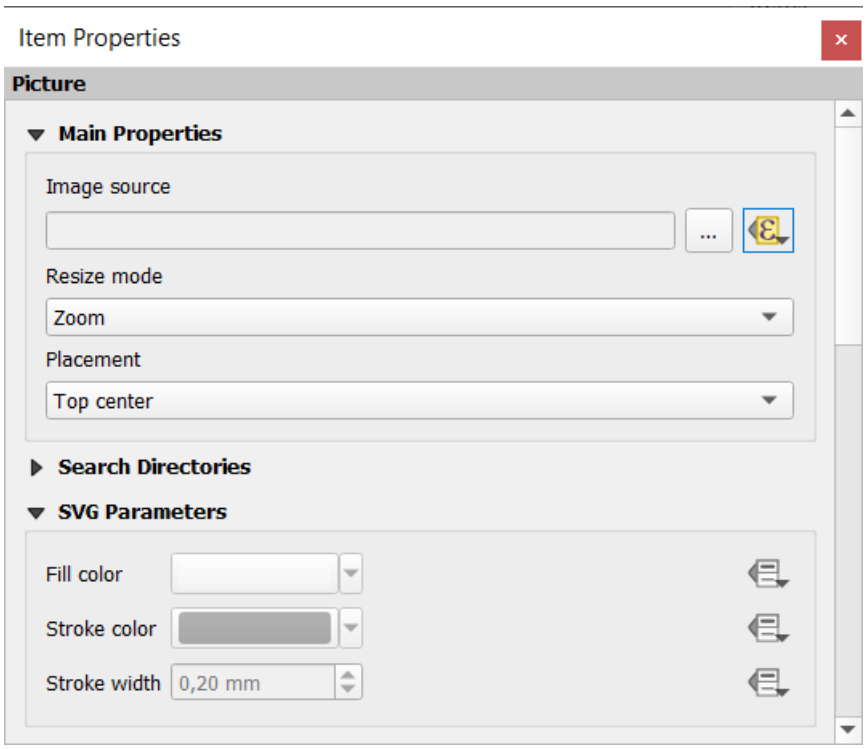
Agreguemos un título sobre el mapa en el centro de la página. Seleccione el elemento de la etiqueta y establezca la alineación horizontal en *Centro*. A continuación, mueva la etiqueta a la posición correcta, elija el botón central para *Punto de referencia*, y proporcione la siguiente expresión para el campo :guilabel:` X`:

```
@layout_pagewidth / 2
```

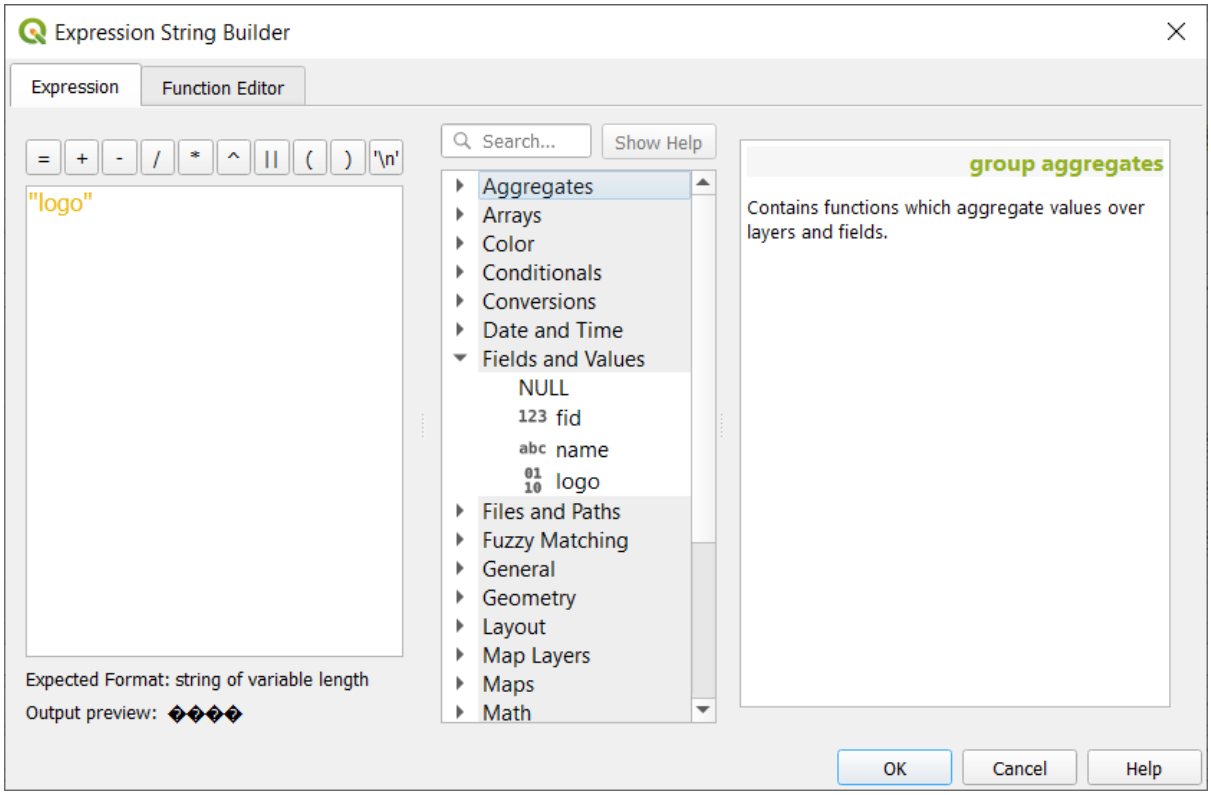
Para todos los demás elementos de diseño, puede establecer la posición de manera similar para que estén colocados correctamente tanto para retrato como para paisaje. También puede hacer más ajustes, como personalizar el título con atributos de objetos (ver ejemplo *Personalizar etiquetas por expresión*), cambiar imágenes, cambiar el tamaño del número de columnas de leyenda según la orientación de la página, …

La información proporcionada aquí es una actualización del excelente blog (en inglés y portugués) sobre las opciones de anulación definida por datos Multiple_format_map_series_using_QGIS_2.6.

Otro ejemplo de uso de botones de anulación definidos por datos es el uso de una imagen dinámica. Para los siguientes ejemplos usamos una capa de geopaquete que contiene un campo BLOB llamado `logo` con el tipo de campo binario (ver *Creando una nueva capa de GeoPackage*). Para cada objeto se define una imagen diferente para que el atlas pueda iterar como se describe en *Obtenga una vista previa y genere un atlas*. Todo lo que necesita hacer es agregar una imagen en el diseño de impresión e ir a sus *Propiedades del elemento* en el contexto del atlas. Allí puede encontrar un botón de suplantación definida por datos en la sección *Fuente de la imagen* de *Propiedades principales*.



En la siguiente ventana, elija *Editar* para que se abra el *Generador de cadenas de expresión*. Desde la sección *Campos y valores* puede encontrar el campo BLOB que se definió en la capa de geopaquete. Haga doble click en el nombre del campo `logo` y haga click en *Aceptar*.



El atlas itera sobre las entradas en el campo BLOB siempre que elija la capa de geopaquete como *Capa de cobertura* (puede encontrar más instrucciones en *Obtenga una vista previa y genere un atlas*).

Estos son solo dos ejemplos de cómo puede usar algunas configuraciones avanzadas con atlas.

Obtenga una vista previa y genere un atlas



Figura 18.69: Barra de herramienta Vista Previa del Atlas

Una vez que se han configurado los ajustes del atlas y los elementos de diseño (mapa, tabla, imagen …) vinculados a él, puede crear una vista previa de todas las páginas eligiendo *Atlas -> Vista preliminar de Atlas* o haciendo clic en el botón de icono Vista previa de Atlas. Luego, puede usar las flechas para navegar por todos los objetos:

- Primer Objeto Espacial
- Objeto Espacial Anterior
- Objeto Espacial Siguiente
- Ultimo Objeto Espacial

También puede utilizar el cuadro combinado para seleccionar y obtener una vista previa de una función específica. El cuadro combinado muestra los nombres de los objetos del atlas de acuerdo con la expresión establecida en la opción de atlas *Nombre de página*.

En cuanto a las composiciones simples, un atlas se puede generar de diferentes maneras (consulte *Crear salida* para obtener más información; solo use las herramientas del menú o barra de herramientas *Atlas* en lugar del menú *Diseño*.

Esto significa que puede imprimir directamente sus composiciones con *Atlas -> Imprimir Atlas*. También puede crear un PDF usando *Atlas -> Exportar Atlas como PDF…*: Se le pedirá un directorio para guardar todos los archivos PDF generados, excepto si se ha seleccionado el *Exportar fichero simple cuando sea posible*. En ese caso, se le pedirá que proporcione un nombre de archivo.

Con *Atlas -> Exportar Atlas como imágenes…* o *Atlas -> Exportar Atlas como SVG …*, también se le pedirá que seleccione una carpeta. Cada página de cada composición de objetos de atlas se exporta al formato de archivo de imagen establecido en el panel *Atlas* o a SVG.

Nota

Con la salida de varias páginas, un atlas se comporta como un diseño en el que solo la página que contiene *Configuración General* obtendrá un archivo georreferenciado (para cada objeto saliente).

Truco

Imprimir un objeto de Atlas específico

Si desea imprimir o exportar la composición de un solo objeto del atlas, simplemente inicie la vista preliminar, seleccione el objeto deseado en la lista desplegable y haga clic en *Diseño -> Imprimir* (o :menuselection:‘Exportar… `a cualquier formato de archivo compatible).

Utilizar relaciones definidas por el proyecto para la creación de atlas

Para los usuarios con conocimientos de HTML y Javascript, es posible operar en objetos GeoJSON y usar relaciones definidas por el proyecto del proyecto QGIS. La diferencia entre este enfoque y el uso de expresiones insertadas directamente en el HTML es que le brinda una función GeoJSON completa y no estructurada con la que trabajar. Esto significa que puede usar bibliotecas y funciones de Javascript existentes que operan en representaciones de objetos de GeoJSON.

El siguiente código incluye todos los objetos hijo relacionados de la relación definida. Usar la función de JavaScript `setFeature`, le permite crear HTML flexible que representa relaciones en el formato que desee (listas, tablas, etc.). En el ejemplo de código, creamos una lista dinámica de viñetas de los objetos secundarios relacionados.

```
// Declare the two HTML div elements we will use for the parent feature id
// and information about the children
<div id="parent"></div>
<div id="my_children"></div>

<script type="text/javascript">
   function setFeature(feature)
   {
     // Show the parent feature's identifier (using its "ID" field)
     document.getElementById('parent').innerHTML = feature.properties.ID;
     //clear the existing relation contents
     document.getElementById('my_children').innerHTML = '';
     feature.properties.my_relation.forEach(function(child_feature) {
     // for each related child feature, create a list element
     // with the feature's name (using its "NAME" field)
       var node = document.createElement("li");
       node.appendChild(document.createTextNode(child_feature.NAME));
       document.getElementById('my_children').appendChild(node);
     });
   }
</script>
```

Durante la creación del atlas, habrá una iteración sobre la capa de cobertura que contiene las entidades principales. En cada página, verá una lista con viñetas de las entidades secundarias relacionadas siguiendo el identificador del padre.

18.4 Creando un informe

Esta sección le ayudará a configurar un informe en QGIS.

18.4.1 ¿Que es?

Por definición, un informe GIS es un documento que contiene información organizada de forma narrativa, que contiene mapas, texto, gráficos, tablas, etc. Un informe puede prepararse ad hoc, periódico, recurrente, regular o según se requiera. Los informes pueden referirse a períodos, eventos, sucesos, temas o ubicaciones específicas.

En QGIS, un *Informe* es una extensión de un *Diseño*.

Los informes permiten a los usuarios generar sus proyectos GIS de una manera simple, rápida y estructurada.

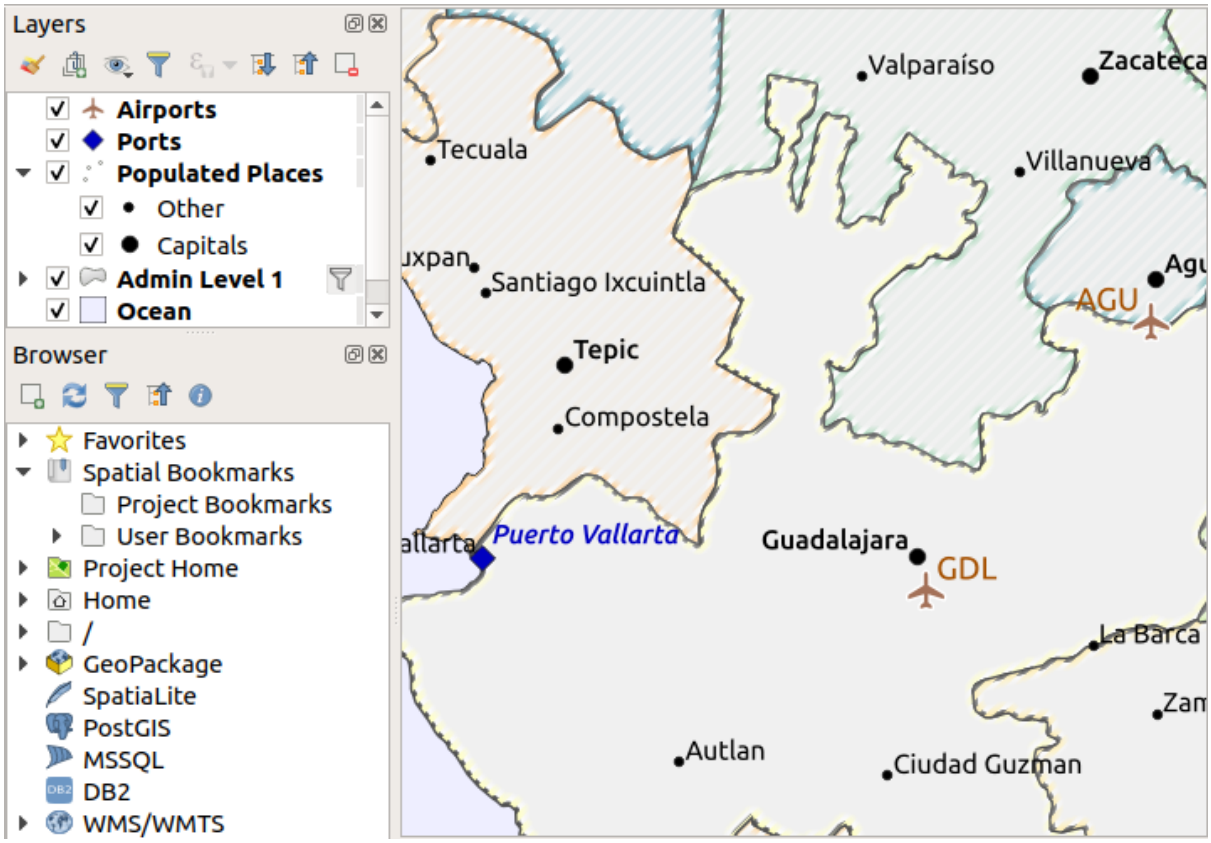
Se puede crear un informe con *Proyecto -> Nuevo informe* o dentro de *Proyecto -> Administrador de composiciones*.

> **Nota**
>
> Los mapas en los informes de QGIS se comportan de la misma manera que los mapas en diseños de impresión y atlas. Nos concentraremos en los detalles específicos de los informes QGIS. Para obtener detalles sobre el manejo de mapas, consulte las secciones sobre *diseños de impresión* y *atlas*.

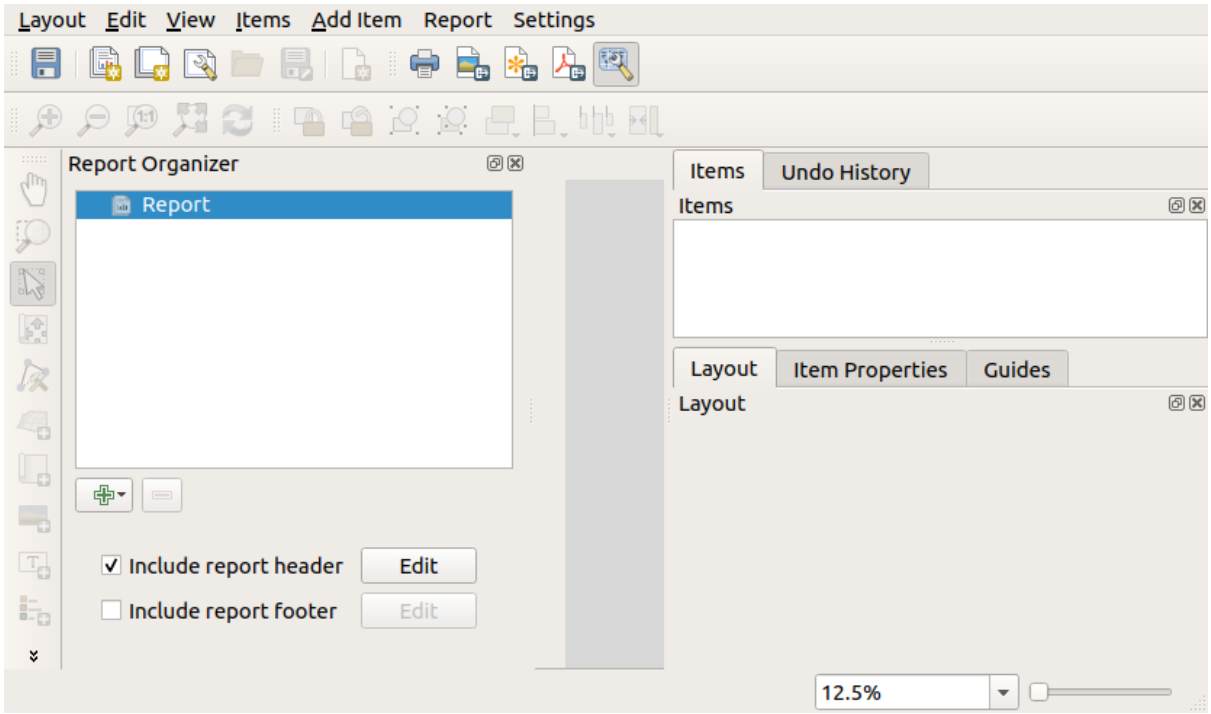
18.4.2 Empezando

En el cuadro de diálogo *Administrador de composiciones*, se puede crear un informe a través de *Nuevo desde plantilla* seleccionando la opción desplegable *Informe vacío* y presionando el botón :guilabel:`Crear… `.

Para este ejemplo, usamos algunos límites administrativos, lugares poblados, puertos y aeropuertos del Conjunto de datos de la Tierra Natural (1:10M).

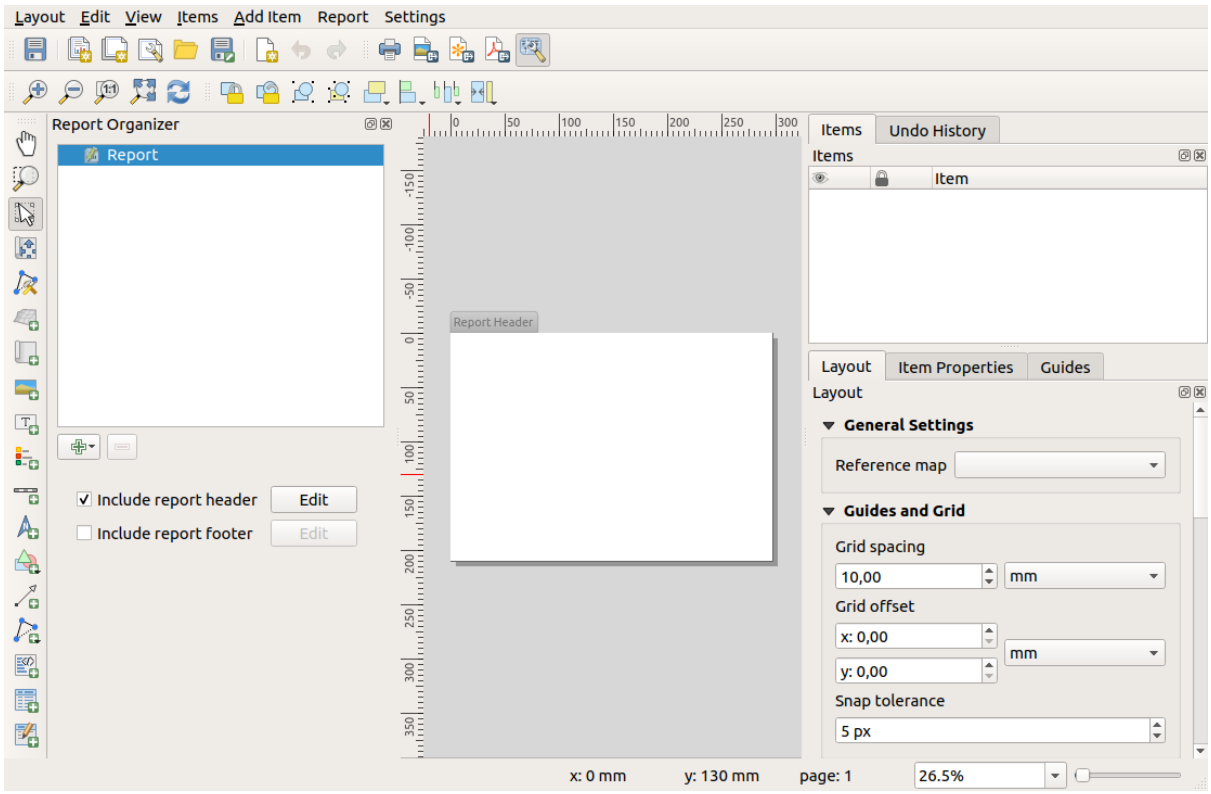


Usando el comando *Proyecto -> Nuevo informe*, creamos un informe en blanco. Inicialmente, no hay mucho que ver: el diálogo que se muestra se parece mucho al diseñador de diseño de impresión, excepto por el panel *Organizador de informes* a la izquierda:



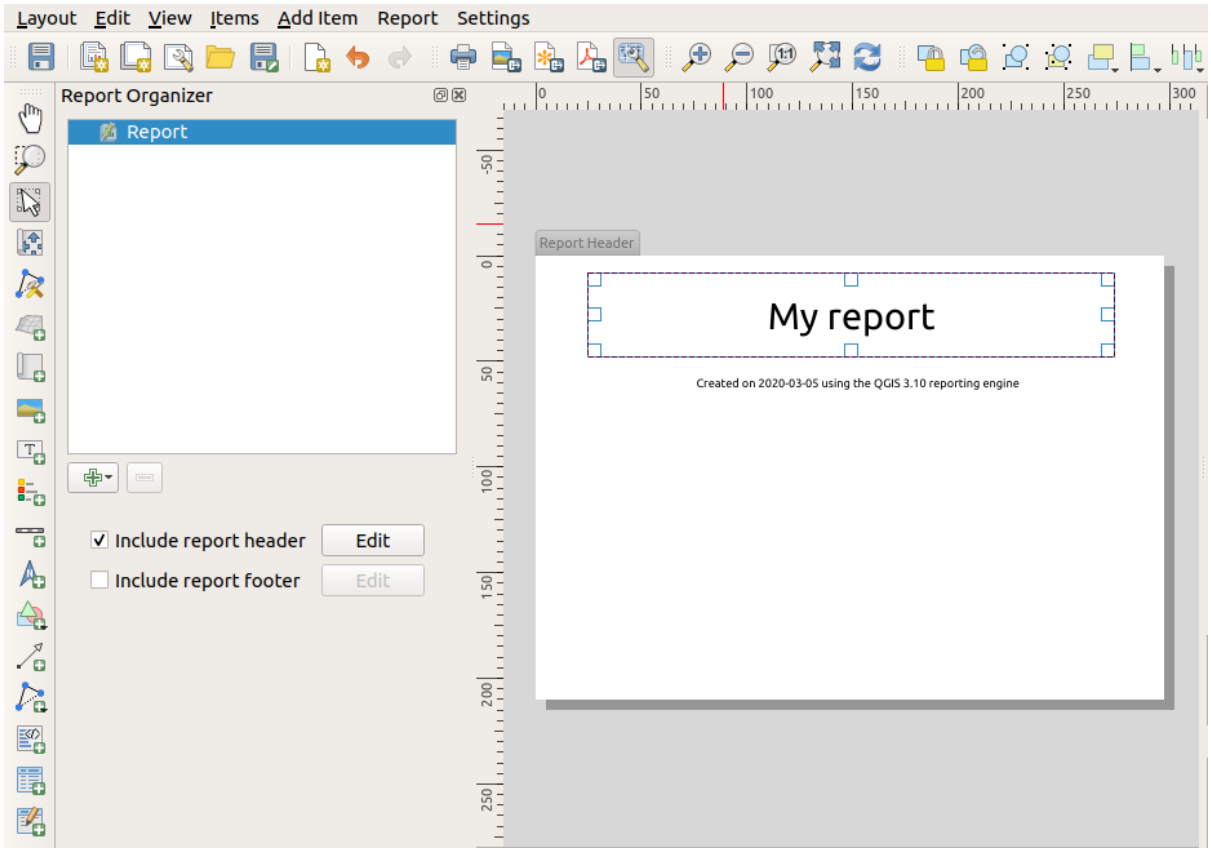
18.4.3 Espacio de trabajo de Diseñador de Informes

Los informes de QGIS pueden constar de varias secciones anidadas. En nuestro nuevo informe en blanco, inicialmente solo tenemos la sección del informe principal. Las únicas opciones para esta sección de informe son *Incluir encabezado de informe* y *Incluir pie de de informe*. Si habilitamos estas opciones, se incluirá un encabezado como primera página(s) (las partes individuales de los informes pueden ser de varias páginas si se desea) en el informe, y un pie de página constituirá la última página(s). Habilite el encabezado (*Incluir encabezado de informe*), y presione el botón *Editar* al lado:

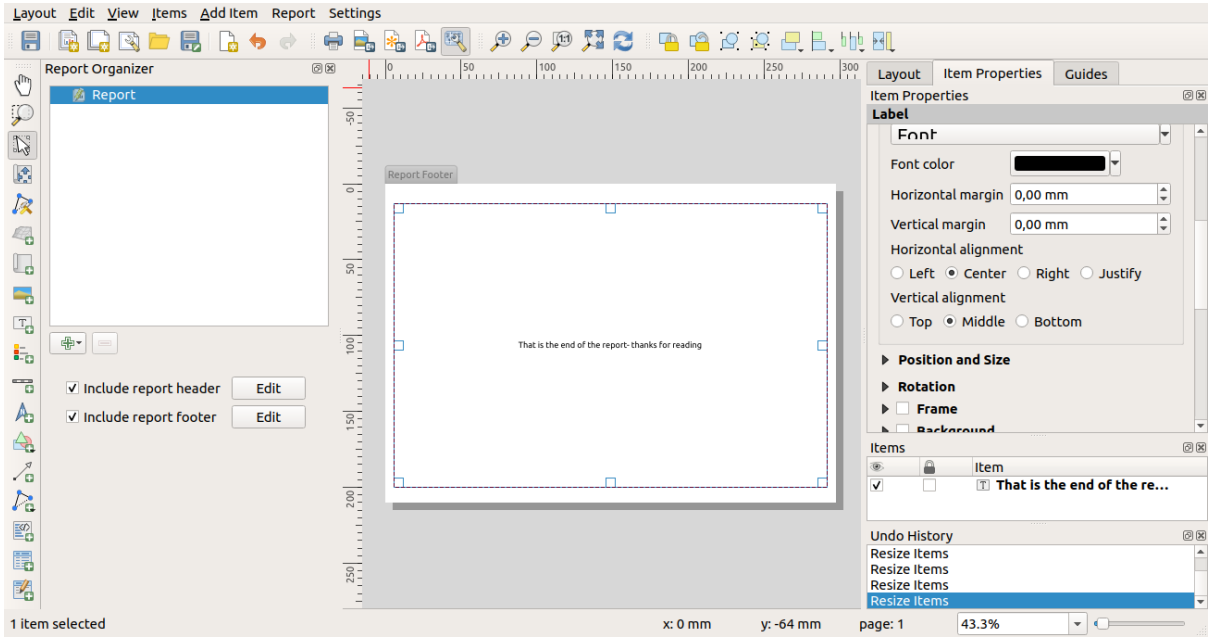


Como resultado, suceden algunas cosas. En primer lugar, se muestra un lápiz de edición junto a *Informe* en el *Organizador de informes*, lo que indica que la sección del informe se está editando actualmente en el diseñador. También vemos una nueva página con un pequeño título *Encabezado de Informe*. La página tiene orientación *horizontal* por defecto, pero esta (y otras propiedades de la página) se puede cambiar haciendo clic derecho en la página y eligiendo *Propiedades de la página*. Esto abrirá la pestaña *Propiedades del elemento* para la página, y se puede especificar para la página el *Tamaño*, *Ancho*, *Alto*, y más.

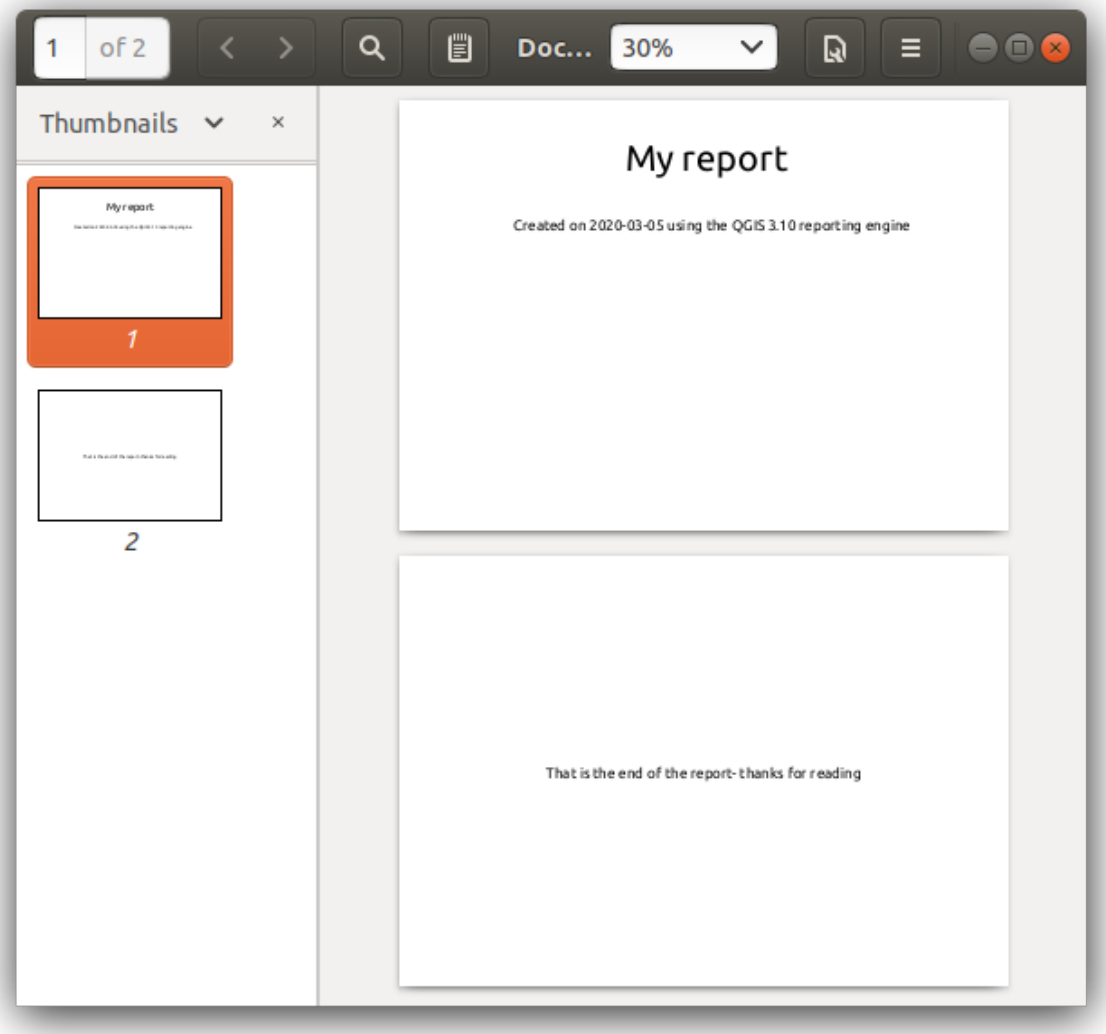
En los informes de QGIS, cada componente del informe se compone de diseños individuales. Se pueden crear y modificar utilizando las mismas herramientas que para los diseños de impresión estándar, por lo que puede usar cualquier combinación deseada de etiquetas, imágenes, mapas, tablas, etc. Agreguemos algunos elementos al encabezado de nuestro informe para demostrar:



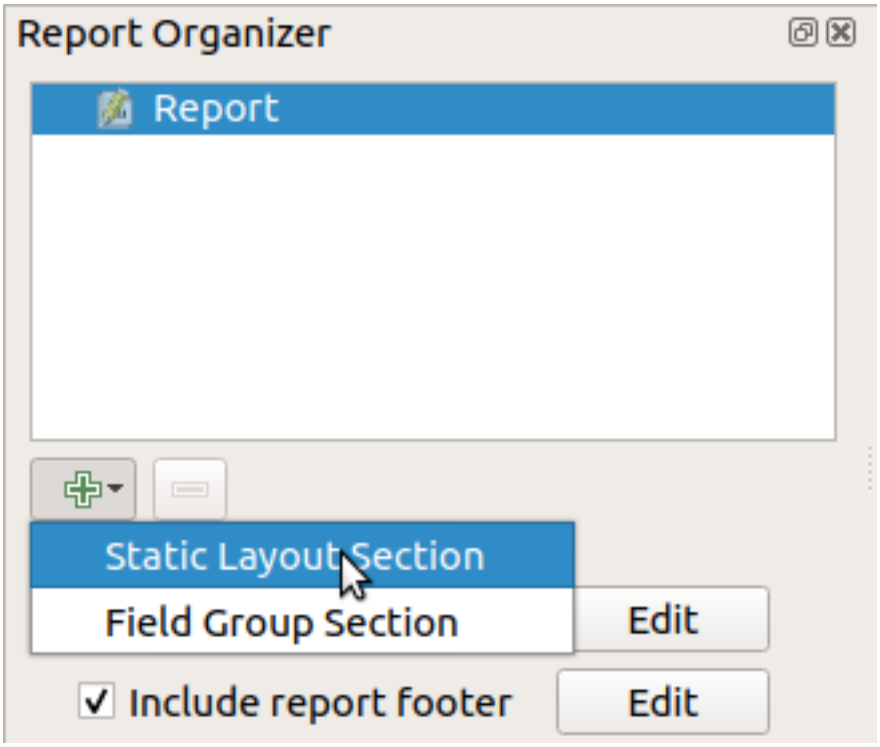
También crearemos un pie de página simple para el informe marcando la opción *Incluir pie del informe* y presionamos *Editar*.



Antes de continuar, exportemos este informe y veamos qué obtenemos. La exportación se realiza desde el menú *Informe*; en este caso, seleccionamos :menuselection:` Exportar Informe como PDF… `para convertir el informe completo en un archivo PDF. Aquí está el resultado no muy impresionante: un PDF de dos páginas que consta de nuestro encabezado y pie de página:



Hagamos las cosas más interesantes. Pulsando el botón Añadir sección en el :guilabel:` Organizador de informes`, se nos ofrece una opción de nuevas secciones a agregar a nuestro informe.

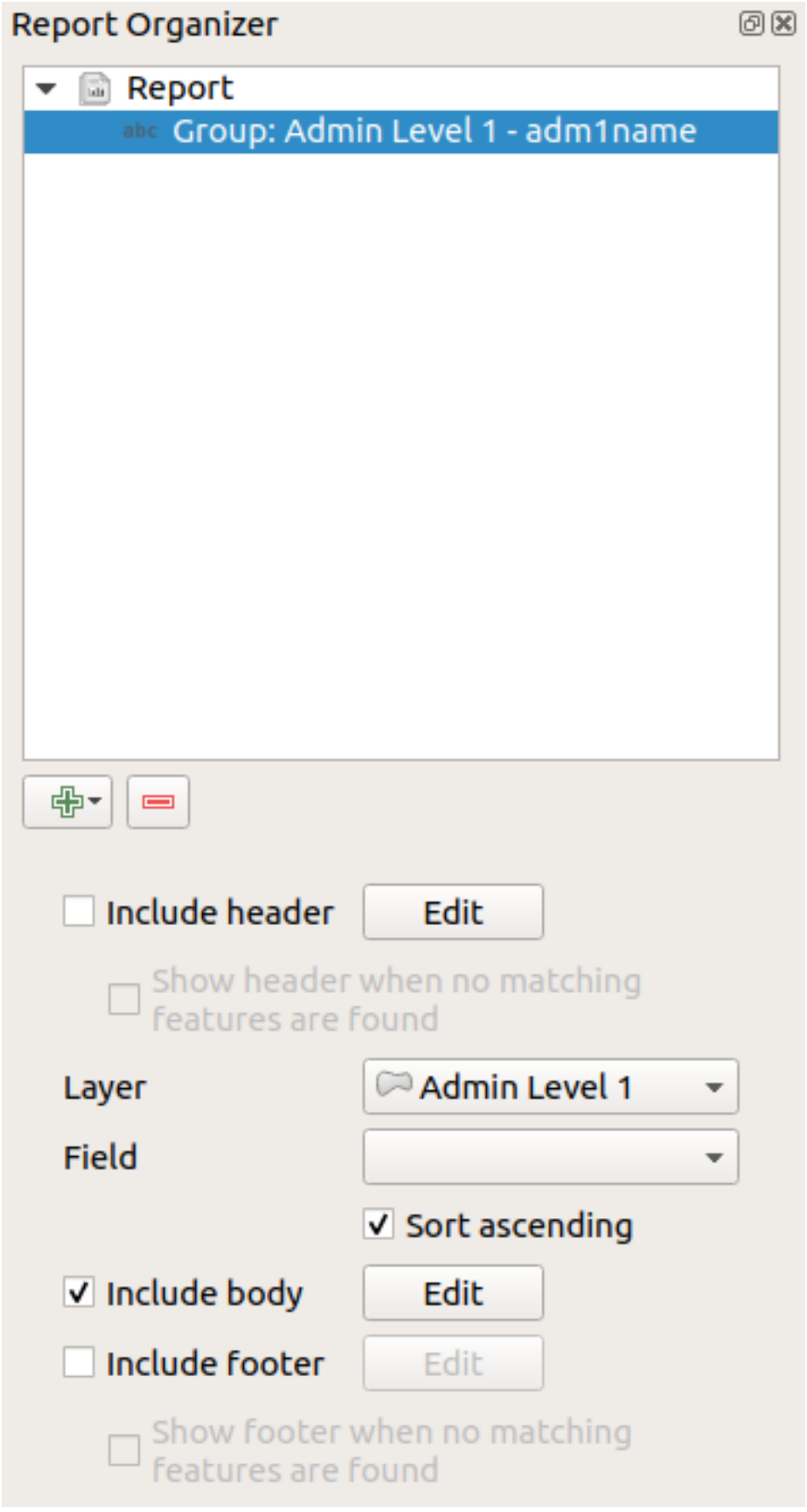


Hay dos opciones: *Sección de diseño estático* y :guilabel:` Sección de grupo de campo`.

El *Añadir Sección de composición estática* es un único diseño de cuerpo estático. Esto se puede utilizar para incrustar diseños estáticos en mitad de un informe.

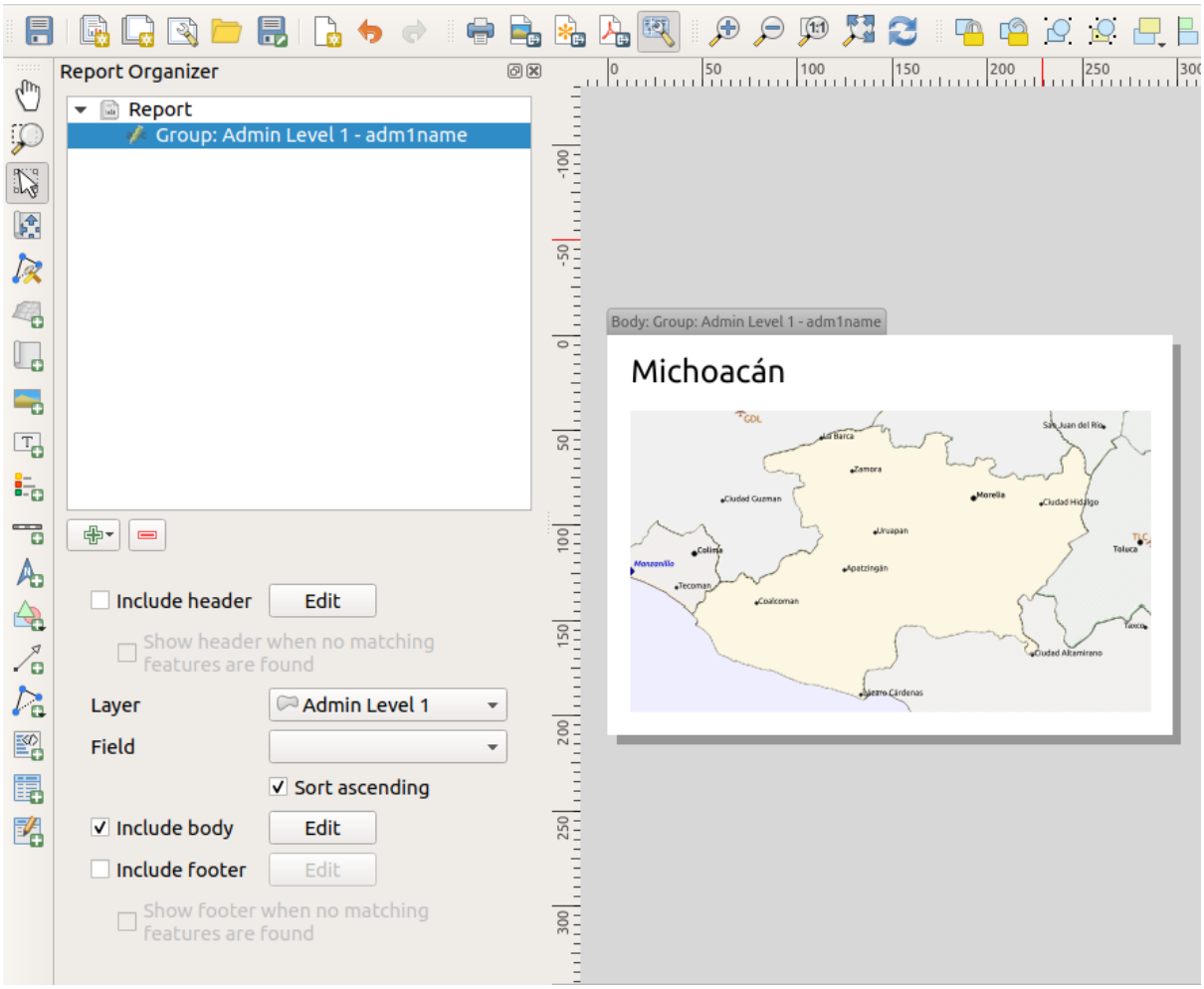
La *Sección de grupo de campos* repite su diseño de cuerpo para cada ojeto de una capa. Los objetos están ordenados por la característica de agrupación seleccionada (con una opción para clasificación ascendente/descendente). Si una sección de grupo de campos tiene secciones secundarias (por ejemplo, otra sección de grupo de campos con un campo diferente), solo se repiten los objetos con valores únicos para la característica de grupo. Esto permite informes anidados.

Por ahora, agregaremos *sección de grupo de campo* a nuestro informe. En su nivel más básico, puede pensar en una *sección de grupo de campo* como el equivalente a *imprimir atlas*: selecciona una capa para iterar y el informe insertará una sección para cada objeto encontrado. La selección de la nueva *sección de grupo campo* revela una serie de nuevas configuraciones relacionadas:



En este caso, configuramos nuestro Grupo de campo para iterar sobre todos los estados de la capa *Admin Level 1*,

usando los valores del campo :guilabel:` adm1name`. Las mismas opciones para incluir encabezado y pie de página están presentes, junto con una nueva opción para incluir *cuerpo* para esta sección. Haremos eso y editaremos el cuerpo:

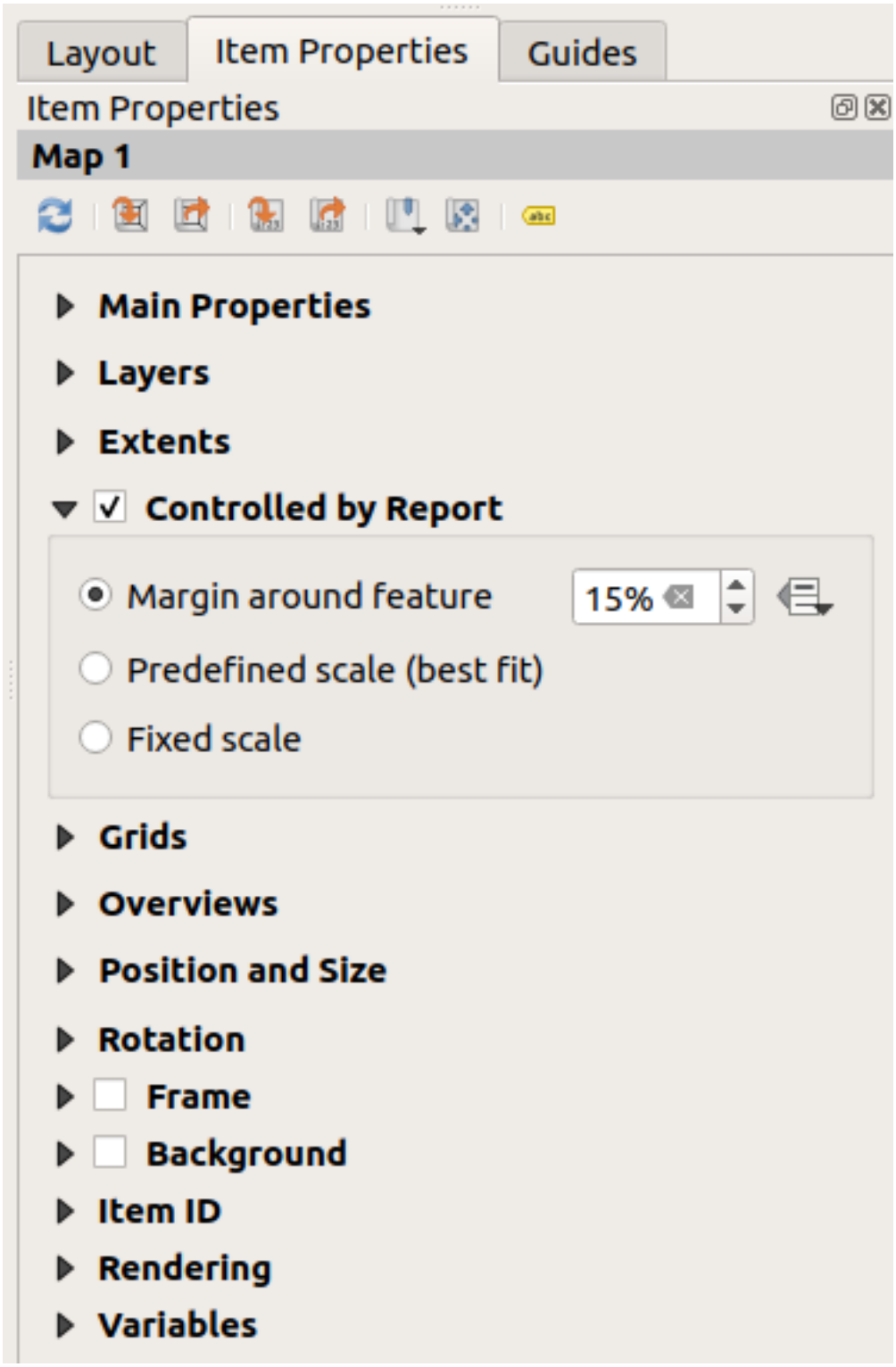


Nuestro cuerpo ahora consta de un mapa y una etiqueta que muestra el nombre del estado. Para incluir el nombre del estado, seleccionamos *Agregar elemento -> Agregar Etiqueta* y los datos definidos por texto en *Propiedades Principales* con la ayuda de *Insertar o Editar una Expresión* .. ..

El resultado fue la siguiente expresión (*name* es el nombre del atributo en la capa *Admin Level 1* que contiene el nombre del estado):

```
[% "name" %]
```

El mapa está configurado para seguir la función de informe actual (habilitado al marcar *Controlado por Informe* - al igual que un elemento de mapa en un atlas seguirá la función de atlas actual cuando :ref:`Controlado por Atlas<controlled_atlas>`está marcado):



Si continuamos y exportamos nuestro informe ahora, obtendríamos algo como esto:

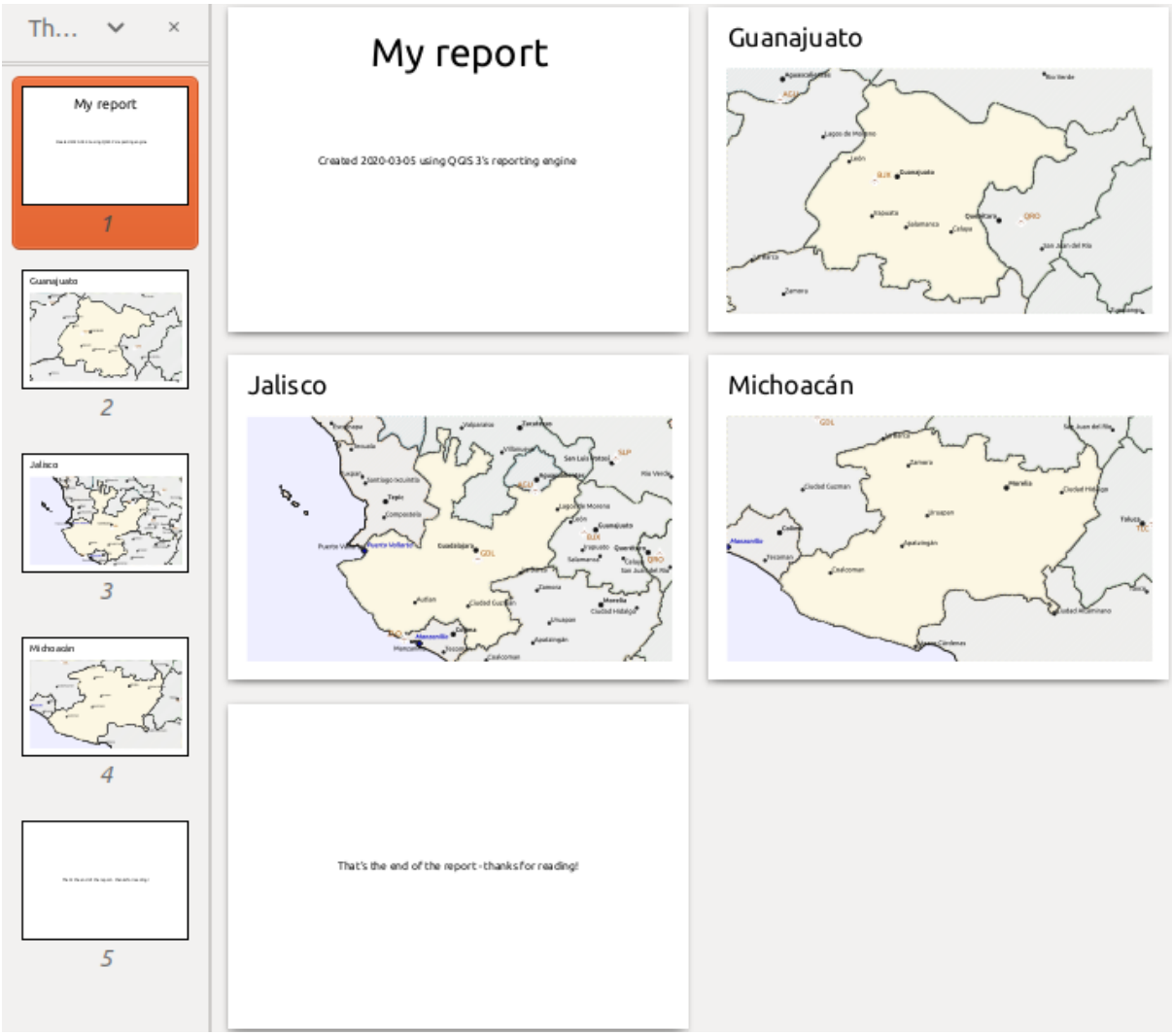
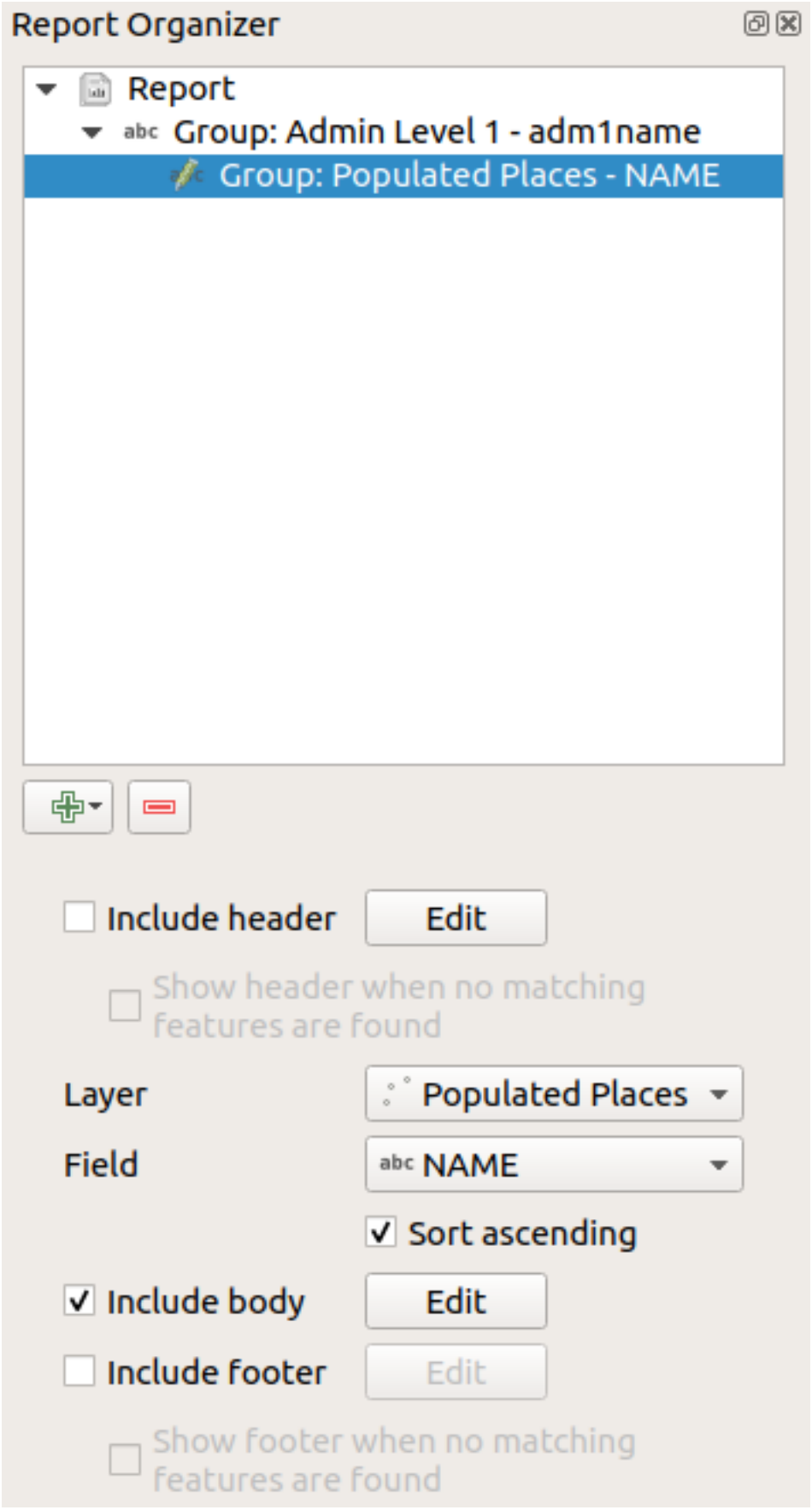


Figura 18.70: El encabezado del informe, una página para cada estado y el pie de página del informe.

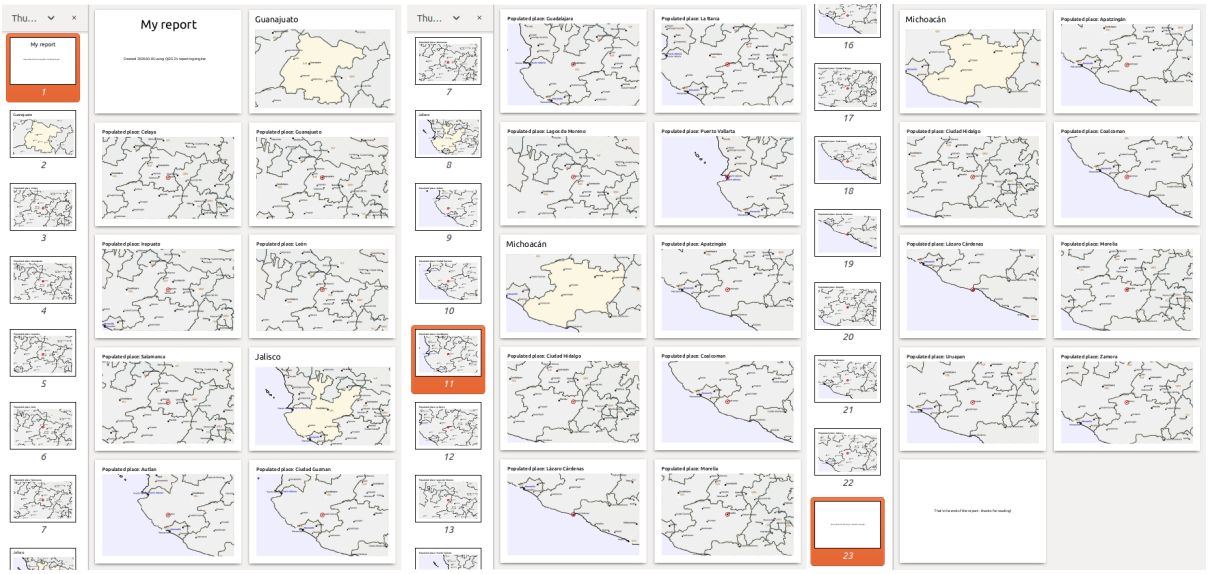
Entonces más o menos un atlas, pero con una página de encabezado y pie de página.

Hagamos las cosas más interesantes agregando una subsección a nuestro grupo de estado. Hacemos esto seleccionando primero el grupo de campo *Admin Level 1* en el organizador, luego presionando el botón Añadir campo y agregando una nueva :guilabel:` Sección de grupo de campo`:



Al iterar sobre los objetos de una *Sección de grupo de campo*, las objetos se filtrarán para que coincidan con el campo

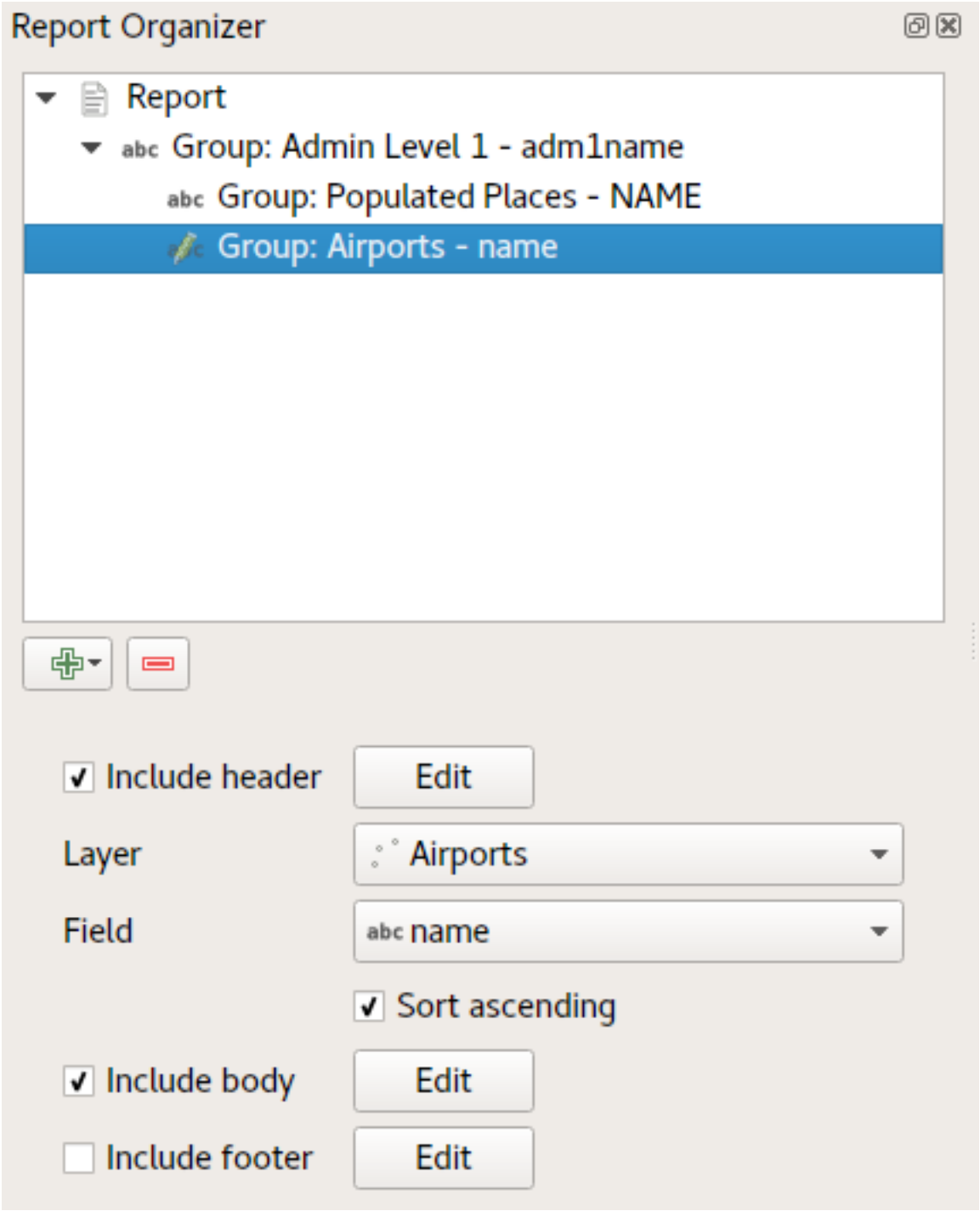
de definición de su grupo principal (`adm1name` en este caso). Aquí, la subsección que agregamos iterará sobre una capa *Populated Places*, incluida una sección del cuerpo para cada lugar encontrado. La magia aquí es que la capa *Populated Places* tiene un atributo con el mismo nombre que el campo de definición en la capa principal, *adm1name*, etiquetando cada lugar con el estado en el que está contenido (si Tiene suerte de que sus datos ya estarán estructurados de esta manera; de lo contrario, ejecute el algoritmo de procesamiento *Unir atributos por ubicación* y cree su propio campo). Cuando exportamos este informe, QGIS tomará el primer estado de la capa *Admin Level 1*, y luego iterará sobre todos los *Populated Places* con un valor *adm1name* coincidente. Esto es lo que obtenemos:



Aquí creamos un cuerpo básico para el grupo de Lugares poblados, que incluye un mapa del lugar y una tabla de algunos atributos del lugar. Así que nuestro informe ahora es un encabezado de informe, una página para el primer estado, seguida de una página para cada lugar poblado dentro de ese estado, luego el resto de los estados con sus lugares poblados y finalmente el pie de página del informe. Si tuviéramos que agregar un encabezado para el grupo Populated Places, se incluiría justo antes de enumerar los lugares poblados para cada estado, como se muestra en la siguiente ilustración.

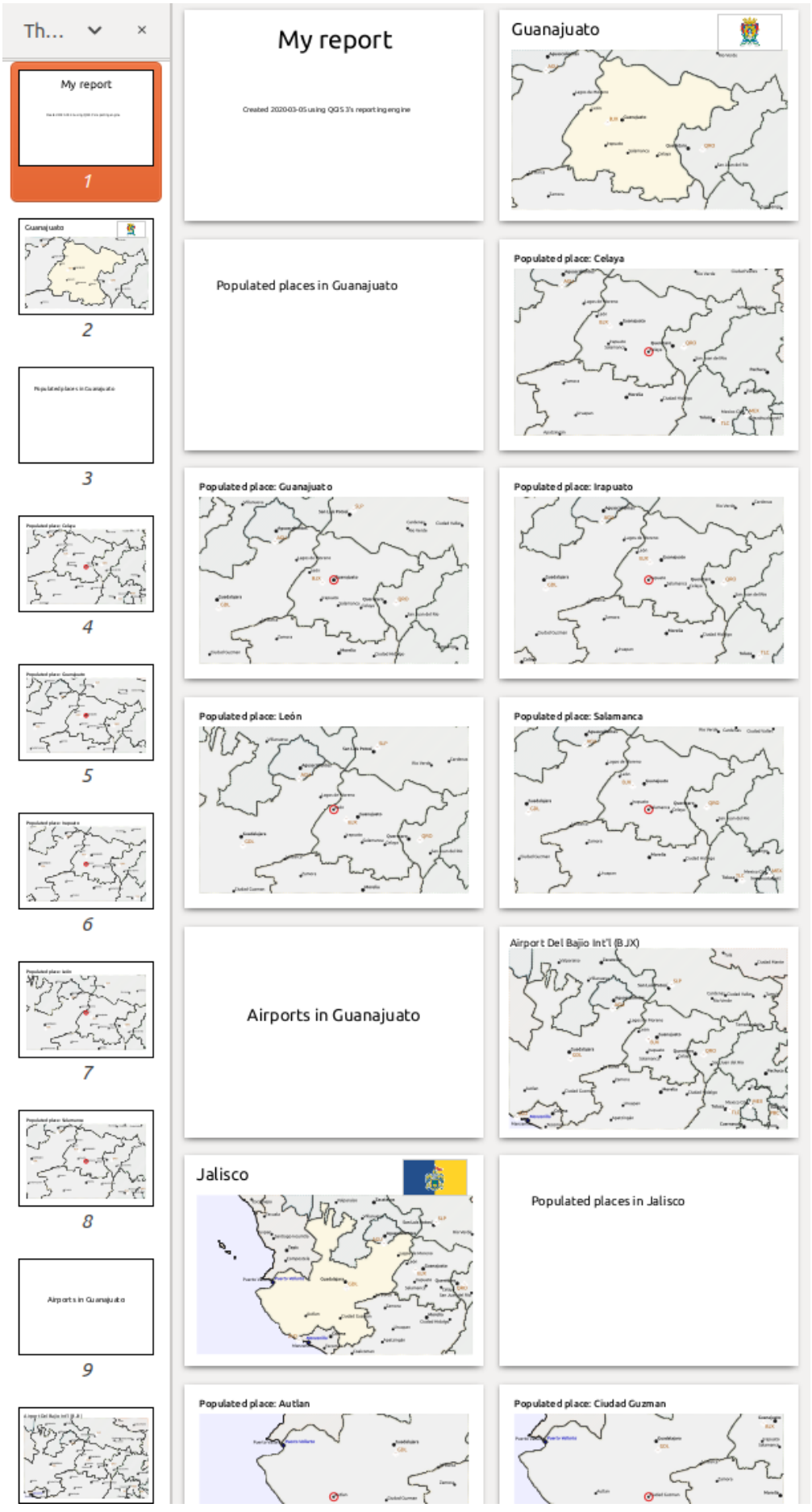
De manera similar, se inserta un pie de página para el grupo Lugares poblados después de que se incluye el lugar final de cada estado.

Además de las subsecciones anidadas, las subsecciones de un informe también se pueden incluir consecutivamente. Si agregamos una segunda subsección al *Grupo de nivel de administrador 1* para :guilabel:` Airports`, entonces (si la capa *Airports* tiene un atributo *adm1name* que puede vincularlo al grupo principal) nuestro informe primero enumerará TODOS los lugares poblados para cada estado, seguidos de todos los aeropuertos dentro de ese estado, antes de pasar al siguiente estado.



El punto clave aquí es que nuestro *Grupo de aeropuertos* es una subsección del :guilabel:` Grupo de Admin Level 1` - no el *Grupo de Populated Places*.

En este caso, nuestro informe estaría estructurado de esta manera (tenga en cuenta que también se han incluido indicadores de estado; el procedimiento para agregar imágenes específicas de objetos de esta manera se describe a continuación):


Th...
My report
1
2
3
4
5
6
7
8
9
My report
Created 2020-03-05 using QGIS 3's reporting engine
Guanajuato
Populated places in Guanajuato
Populated place: Celaya
Populated place: Guanajuato
Populated place: Irapuato
Populated place: León
Populated place: Salamanca
Airports in Guanajuato
Airport Del Bajio Int'l (BJX)
Jalisco
Populated places in Jalisco
Populated place: Autlan
Populated place: Ciudad Guzman

Incluir imágenes en un informe

Las imágenes pueden ser bastante útiles en los informes y QGIS permite imágenes tanto en las partes estáticas como dinámicas de un informe. Las imágenes se agregan de la misma manera que para los diseños de impresión estándar, y para las partes del informe estático (e imágenes estáticas en partes dinámicas) no hay más.

Pero si desea ilustraciones que se adapten a los objetos del informe, su capa debe tener un atributo que se pueda usar para definir la imagen que se incluirá.

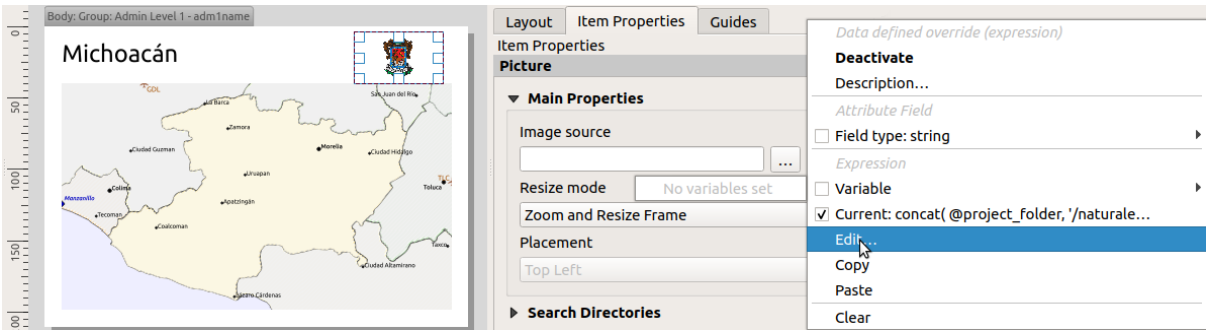
QGIS depende de los nombres de archivo absolutos para las imágenes en los informes.

Para imágenes dinámicas, primero agregue una imagen a la parte del cuerpo del grupo, como de costumbre. En :guilabel:‘propiedades de elemento` de la imagen, establece :guilabel:‘Origen de la imagen` usando el botón Suplantación de datos definidos, y seleccione un atributo que contenga la ruta absoluta de las imágenes o :guilabel:‘Editar…` (para ingresar una expresión que genere la ruta absoluta de la imagen).

A continuación se muestra una expresión de ejemplo que usa la concatenación de cadenas para especificar la ruta absoluta a las imágenes, usando el directorio donde se encuentra el archivo del proyecto `@project_path`) y un atributo (`adm1name`) desde el cual se genera el nombre del archivo (en este caso transformando la cadena en el atributo `adm1name` a mayúsculas y agregando “_flag.png”):

```
concat(@project_folder, '/naturalearth/pictures/' ,
       upper("adm1name"), '_flag.png')
```

Esto significa que las imágenes se encuentran en el subdirectorio `naturalearth/pictures` del directorio de archivos del proyecto.



Resaltar la función del informe actual en un mapa

En el informe anterior, los objetos del informe se enfatizan en los mapas mediante resaltado (estado) y círculos (lugares poblados). Para enfatizar los objetos del informe en los mapas (además de colocarlos en el centro de los mapas), debe definir el estilo de los datos mediante una comparación entre su `@id` y el `@atlas_featureid`, como para los atlas .

Por ejemplo, si desea utilizar una línea/borde más grueso para el objeto del informe que los otros objetos, puede definir los datos del ancho de línea:

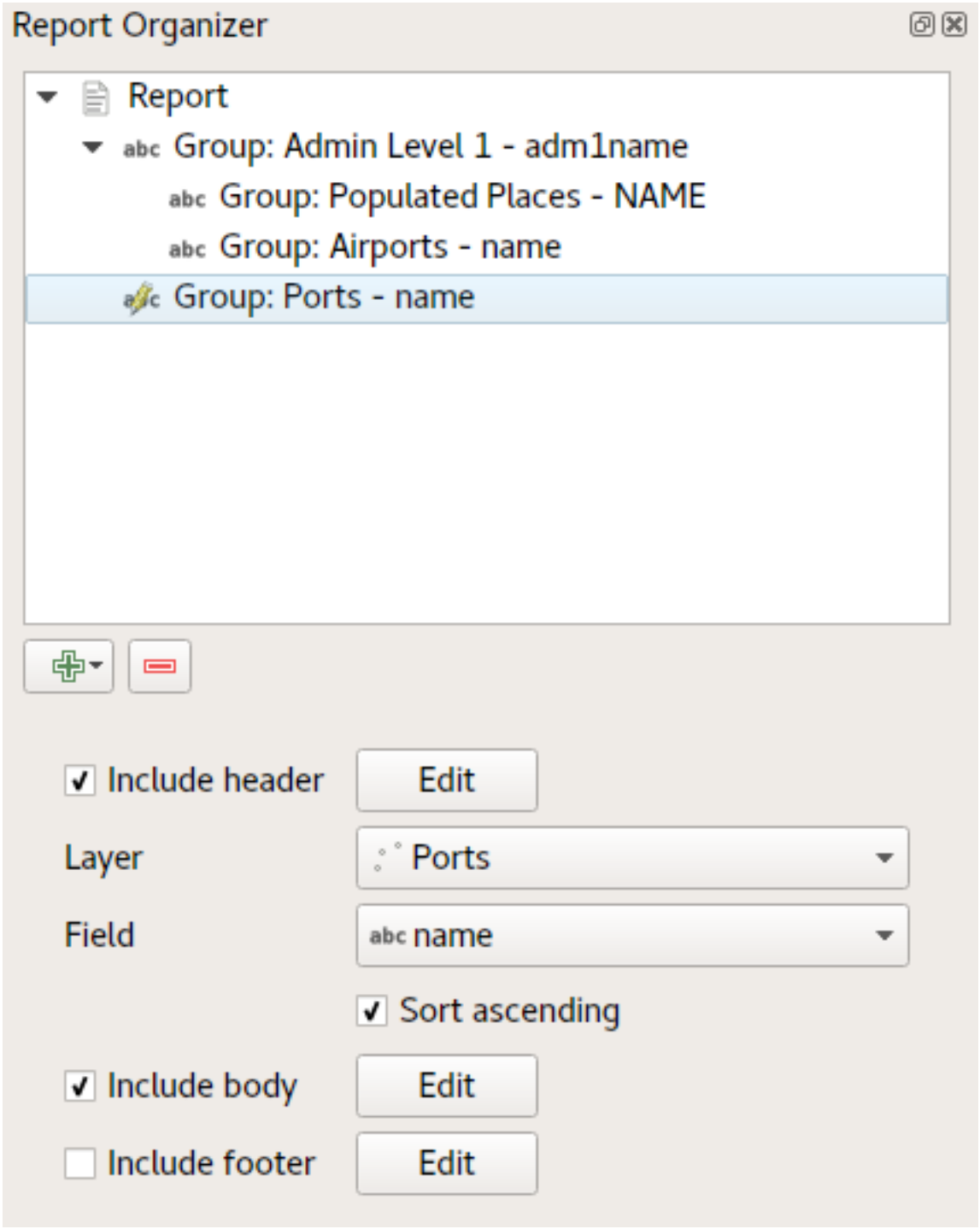
```
if($id=@atlas_featureid, 2.0, 0.1)
```

La objeto de informe obtendrá un contorno de polígono de 2 unidades de ancho, mientras que todas los demás objetos tendrán una línea de 0,1 unidades de ancho. También es posible definir los datos del color (magenta oscuro no transparente para el objeto de informe y gris claro semitransparente para los otros objetos):

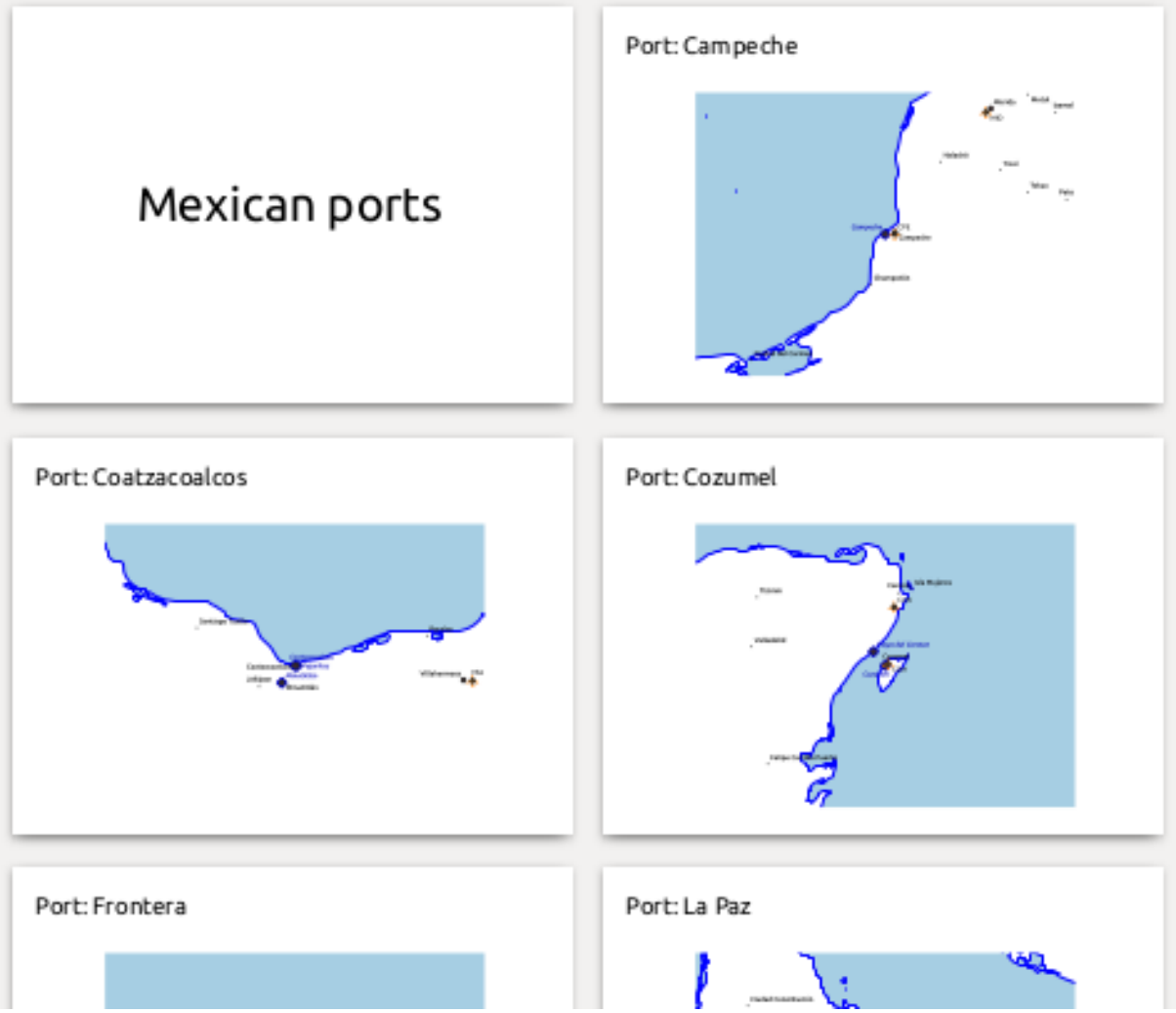
```
if($id=@atlas_featureid, '#FF880088', '#88CCCCCC')
```

Mas grupos Nivel 1

La combinación de secciones anidadas y consecutivas, junto con los encabezados y pies de sección, permite una gran flexibilidad. Por ejemplo, en el siguiente informe añadimos otro grupo de campos como hijo del informe principal para la capa *Puertos*. Ahora, tras listar los estados junto con sus lugares poblados y aeropuertos, obtendremos una lista resumen de todos los puertos de la región:



Esto dá como resultado que la última parte de nuestro informe se exporte como:



18.4.4 Configuración de exportación

Cuando exporta un informe (*Informe -> Exportar informe como imágenes … / SVG … / PDF …*), se le pedirá un nombre de archivo y luego tendrá la oportunidad de ajuste la configuración de exportación para obtener el resultado más apropiado.

Como puede ver, los informes en QGIS son extremadamente poderosos y flexibles.

Nota

La información actual fue adaptada de un blog de North Road, 'Exploring Reports in QGIS 3.0 - the Ultimate Guide! <https://north-road.com/2018/01/23/exploring-reports-in-qgis-3-0-the-ultimate-guide>`_

CAPÍTULO 19

Trabajando con protocolos OGC / ISO

El Open Geospatial Consortium (OGC) es una organización internacional con miembros de más de 300 organizaciones comerciales, gubernamentales, sin fines de lucro y de investigación de todo el mundo. Sus miembros desarrollan e implementan estándares para contenido geoespacial y servicios, procesamiento de datos SIG y el intercambio.

Describir un modelo de datos básico para características geográficas, OGC desarrolla un creciente número de especificaciones para cubrir necesidades específicas de ubicación interoperable y tecnología espacial, incluyendo GIS. Se puede encontrar más información en https://www.ogc.org/.

Importar especificaciones OGC implementadas por QGIS:

- **WMS** — Web Map Service (*Cliente WMS/WMTS*)
- **WMTS** — Web Map Tile Service (*Cliente WMS/WMTS*)
- **WFS** — Web Feature Service (*Cliente WFS y WFS-T*)
- **WFS-T** — Web Feature Service - Transactional (*Cliente WFS y WFS-T*)
- **WCS** — Web Coverage Service (*WCT Cliente*)
- **WPS** — Web Processing Service
- **CSW** — Catalog Service for the Web
- **SFS** — Simple Features for SQL (*PostgreSQL Database*)
- **GML** — Geography Markup Language
- **API de SensorThings** — API de SensorThings (*SensorThings*)

OGC services are increasingly being used to exchange geospatial data between different GIS implementations and data stores. QGIS can deal with the above specifications as a client, being **SFS** (through support of the PostgreSQL data provider, see section *PostgreSQL Database*).

También puede compartir sus mapas y datos a través de los protocolos WMS, WMTS, WFS, WFS-T y WCS usando un servidor web con QGIS Server, UMN MapServer o GeoServer instalado.

19.1 Cliente WMS/WMTS

19.1.1 Información general de la implementación WMS

Actualmente QGIS puede actual como un cliente WMS que entiende servidores WMS 1.1, 1.1.1 y 1.3. En particular, se ha probado contra los servidores de acceso público como DEMIS.

Un servidor WMS actúa sobre las peticiones por parte del cliente (por ejemplo, QGIS) para un mapa ráster con una extensión dada, el conjunto de capas, el estilo de simbolización y la transparencia. El servidor WMS después consulta sus fuentes de datos locales, rásteriza el mapa y lo envía de vuelta al cliente en formato ráster. Para QGIS, este formato sería típicamente JPEG o PNG.

WMS es genéricamente un servicio REST (Representational State Transfer) en lugar de un servicio Web en toda regla. Como tal, puede tomar las URLs generadas por QGIS y utilizarlos en el navegador web para recuperar las mismas imágenes que QGIS utiliza internamente. Esto puede ser útil para la solución de problemas, ya que hay varias marcas de servidores WMS en el mercado y ellos tienen su propia interpretación de la estándar WMS.

Las capas WMS se pueden añadir sencillamente, siempre que conozca la URL para acceder al servidor WMS, si tiene una conexión útil a ese servidor, y el servidor entiende HTTP como mecanismo de transporte de datos.

Además, QGIS almacenará en caché sus respuestas WMS (es decir, imágenes) durante 24 horas mientras no se active la solicitud GetCapabilities. La solicitud GetCapabilities se activa cada vez que se utiliza el botón *Conectar* del cuadro de diálogo *WMS/WMTS* para recuperar las capacidades del servidor WMS. Se trata de una función automática destinada a optimizar el tiempo de carga del proyecto. Si se guarda un proyecto con una capa WMS, los mosaicos WMS correspondientes se cargarán desde la caché la próxima vez que se abra el proyecto, siempre que no tengan más de 24 horas de antigüedad.

19.1.2 Información general de la implementación WMTS

QGIS también puede actuar como un cliente WMTS. WMTS es un estándar OGC para distribuir tiles de conjunto de datos geoespaciales. Este es una manera más rápida y más eficiente de distribuir datos que WMS porque con WMTS, el conjunto de tile es pregenerado y el cliente sólo solicita la transmisión de los tiles, no su producción. Una petición WMS implica típicamente ambos la generación y la transmisión de los datos. Un ejemplo bien conocido de un estándar no OGC para la visualización de datos geoespaciales de tiles es Google Maps.

Para mostrar los datos en una variedad de escalas cercanas a lo que el usuario podría querer, los conjuntos de teselas WMTS se producen en varios niveles de escala diferentes y están disponibles para el cliente SIG para pedirlos.

Este diagrama ejemplifica el concepto de conjunto de teselas:

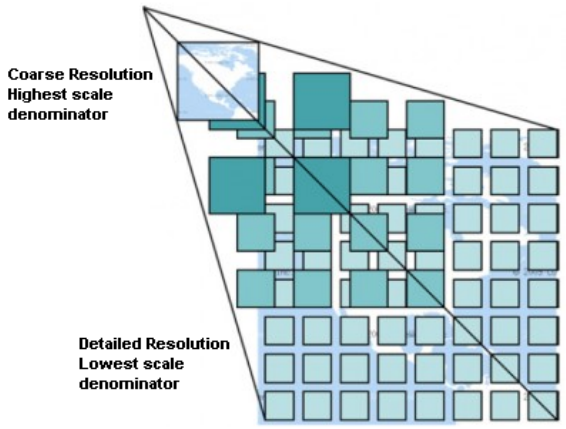


Figura 19.1: Concepto de conjunto de teselas WMTS

Los dos tipos de interfaz WMTS que QGIS reconoce son a través de Key-Value-Pairs (KVP) y RESTful. Estas dos interfaces son diferentes y es necesario especificarlos a QGIS diferente.

1. Con el fin de acceder a un servicio **WMTS KVP**, un usuario QGIS debe abrir la interfaz WMS/WMTS y añadir la siguiente cadena a la URL del servicio tile WMTS:

   ```
   "?SERVICE=WMTS&REQUEST=GetCapabilities"
   ```

   Un ejemplo de este tipo de dirección es:

   ```
   https://opencache.statkart.no/gatekeeper/gk/gk.open_wmts?service=WMTS&
   ↪request=GetCapabilities
   ```

   Para probar la capa topo2 en este WMTS funciona muy bien. Añadir esta cadena indica que un servicio web WMTS se va a utilizar en lugar de un servicio WMS.

2. EL servicio **RESTful WMTS** toma una forma diferente, una URL sencilla. EL formato recomendado por OGC es:

   ```
   {WMTSBaseURL}/1.0.0/WMTSCapabilities.xml
   ```

   Este formato ayuda a reconocer que se trata de una dirección RESTful. Se accede a un WMTS RESTful en QGIS simplemente añadiendo su dirección en la configuración del WMS en el campo URL del formulario. Un ejemplo de este tipo de dirección para el caso de un mapa base austriaco es:

   ```
   https://maps.wien.gv.at/basemap/1.0.0/WMTSCapabilities.xml
   ```

Nota

Todavía puede encontrar algunos servicios antiguos llamados WMS-C. Estos servicios son bastante similares a WMTS (es decir, tienen el mismo propósito pero funcionan de manera un poco diferente). Puede administrarlos de la misma manera que lo hace con los servicios WMTS. Simplemente agregue `?Tiled=true` al final de la URL. Ver https://wiki.osgeo.org/wiki/Tile_Map_Service_Specification para mas información sobre esta especificación.

Cuando se lee WMTS, a menudo se puede pensar en WMS-C también.

19.1.3 Seleccionar servidor WMS/WMTS

La primera vez que se utiliza la función WMS/WMTS en QGIS, no hay ningún servidor definido.

Luego, debe crear conexiones con el servidor al que se dirige:

1. Vaya a la pestaña  *WMS/WMTS* del Cuadro de diálogo *Administrador de Fuentes de Datos*, ya sea por:
   - click en el botón  Abrir Administrador de Fuentes de Datos (o presionando `Ctrl+L`) y activando la pestaña
   - click en el botón  Añadir capa WMS en la barra de herramientas *Administrador de Capas*
   - o seleccionando el menú *Capa* ► *Añadir Capa* ►  *Añadir capa WMS/WMTS…*
2. Presiona *Nuevo* en la pestaña *Capas*. El cuadro de diálogo *Crear una Nueva Conexión WMS/WMTS…* aparece.

   Truco

   Click derecho en la entrada  *WMS/WMTS* dentro del *panel navegador* y selecciona *Nueva Conexión…* también abre el cuadro de diálogo *Crear una nueva conexión WMS/WMTS…*.

3. Luego introduzca los parámetros para conectar con su servidor WMS deseado. como se lista abajo:

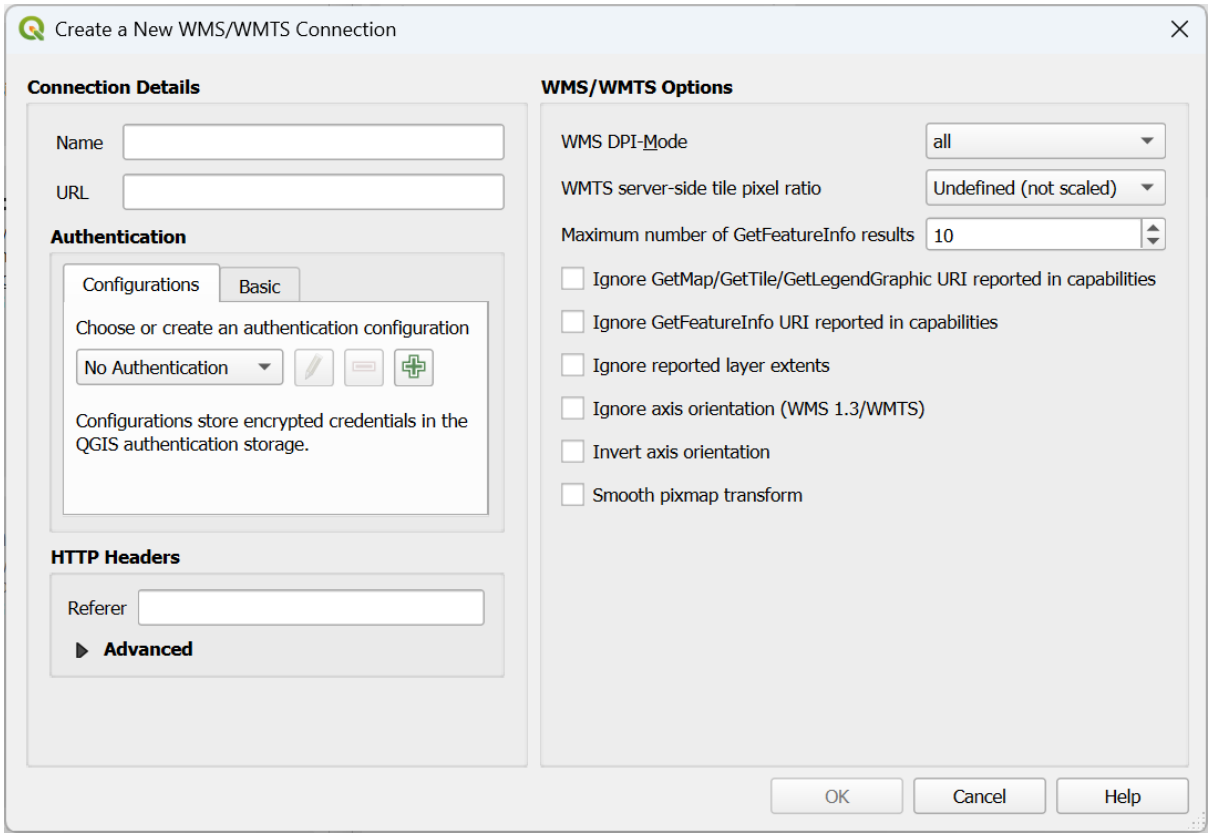


Figura 19.2: Crear una conexión a un servidor WMS

- *Nombre*: Un nombre para la conexión. Este nombre se utilizará en el cuadro desplegable Conexiones del servidor para que pueda distinguirlo de otros servidores WMS.
- *URL*: URL del servidor que proporciona los datos. Debe ser un nombre de host que se pueda resolver, el mismo formato que usaría para abrir una conexión telnet o hacer ping a un host, es decir, solo la URL base. Por ejemplo, no debería tener fragmentos como `request=GetCapabilities` o `version=1.0.0` en su URL.
- *Autenticación* (opcional): usanr una *configuración almacenada* o una autenticación básica con *Nombre de Usuario* y *Contraseña*.

> **Advertencia**
>
> Al ingresar **nombre de usuario** y **contraseña** en la pestaña *Autenticación* mantendrá las credenciales desprotegidas en la configuración de la conexión. Esas **credenciales serán visibles** si, por ejemplo, compartiste el archivo del proyecto con alguien. Por lo tanto, es aconsejable guardar sus credenciales en una *Configuración de autenticación* en su lugar (pestaña *Configuraciones*). Ver :ref:‘authentication_index` para más detalles.

- *Referente* HTTP
- *Modo PPP de WMS*: Las opciones disponibles son **all**, **off**, **QGIS**, **UMN** y **GeoServer**.
- *Relación de píxeles de tesela del lado del servidor WMTS*: Al renderizar capas WMTS, permite aumentar o reducir la escala de los mosaicos en función de los PPP de la pantalla del dispositivo. Las opciones disponibles son *Indefinido (no escalado)*, *Estándar (96 PPP)* y *Alto (192 PPP)*.
- *Número máximo de resultados de GetFeatureInfo*: especifica un valor predeterminado para el número máximo de resultados devueltos por capa por una solicitud GetFeatureInfo utilizando esta conexión (vea el parámetro FEATURE_COUNT). El valor predeterminado es `10`. Establezca el valor en `0` para utilizar

el valor predeterminado del servidor (normalmente `1`): no se añadirá el parámetro FEATURE_COUNT a la solicitud.

- ☐ *Ignorar el URI GetMap/GetTile/GetLegendGraphic indicado en las capacidades*: si está marcada, utiliza el URI indicado en el campo *URL* anterior.
- ☐ *Ignorar GetFeatureInfo URI reportado en capabilities*: si está marcado, usa el URI dado en el campo anterior *URL*.
- ☐ *Ignorar las extensiones de capa comunicadas*: debido a que la extensión informada por las capas ráster puede ser más pequeña que el área real que se puede representar (especialmente para servidores WMS con simbología que ocupa más espacio que la extensión de datos), marque esta opción para evitar recortar capas ráster a sus extensiones informadas, lo que resulta en símbolos truncados en los bordes de estas capas.
- ☐ *Ignorar orientación del eje (WMS 1.3/WMTS)*
- ☐ *Invertir orientación del eje*
- ☐ *Suavizar transformación de pixel de mapa*

4. Presiona *Aceptar*

Una vez creada la nueva conexión al servidor WMS/WMTS, ésta se conservará para futuras sesiones de QGIS. Tenga en cuenta que también es posible *Cargar* los parámetros de conexión desde un archivo `.XML` o *Guardarlos* en un archivo `.XML`.

Si necesita configurar un servidor proxy para poder recibir servicios WMS desde Internet, puede agregar su servidor proxy en las opciones. Elija *Configuración* -> *Opciones* y haga click en la pestaña *Red*. Allí, puede agregar su configuración de proxy y habilitarla configurando ☑ *Usar proxy para acceso web*. Asegúrese de seleccionar el tipo de proxy correcto del menú desplegable *Tipo de proxy* [...▾].

19.1.4 Cargando capas WMS/WMTS

Una vez que haya completado con éxito sus parámetros, puede usar el botón *Conectar* para recuperar las capacidades del servidor seleccionado. Esto incluye la codificación de la imagen, las capas, los estilos de capa y las proyecciones. Dado que se trata de una operación de red, la velocidad de la respuesta depende de la calidad de su conexión de red al servidor WMS. Mientras se descargan datos del servidor WMS, el progreso de la descarga se visualiza en la esquina inferior izquierda del cuadro de diálogo principal de QGIS.

Su pantalla ahora debería parecerse un poco a Figura 19.3, que muestra la respuesta proporcionada por un servidor WMS.

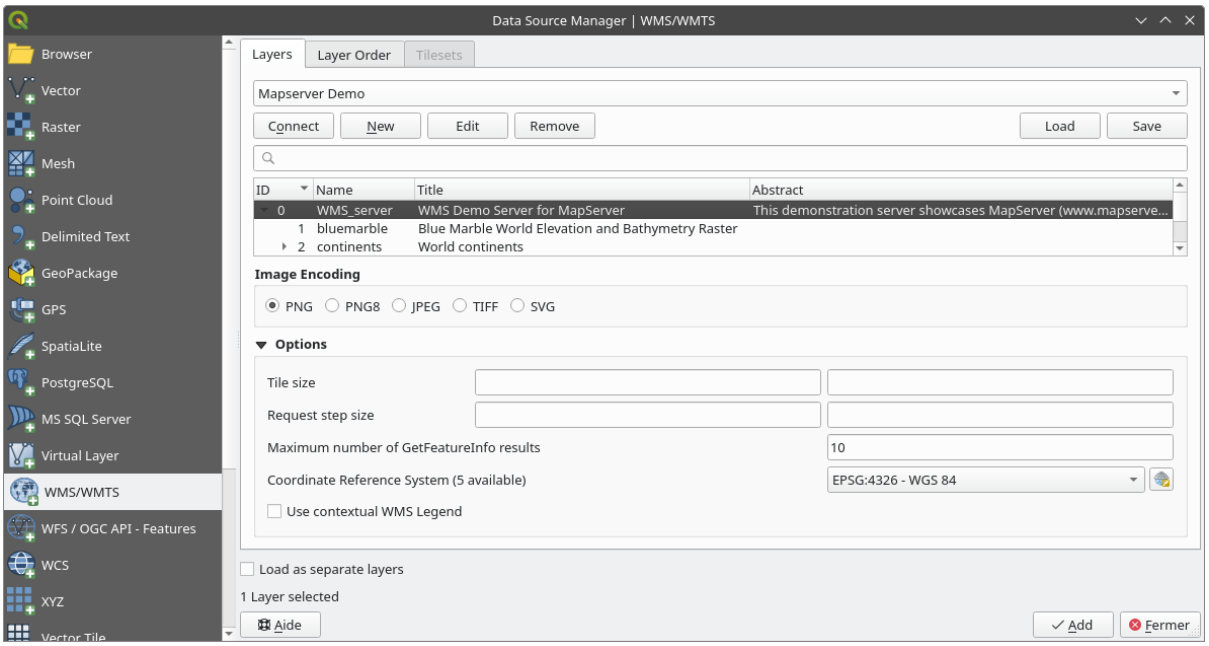


Figura 19.3: Cuadro de diálogo para agregar un servidor WMS, con filtro en capas disponibles

La parte superior de la pestaña *Capas* del diálogo muestra una estructura de árbol que puede incluir grupos de capas que incrustan capas con sus estilos de imagen asociados servidos por el servidor. Cada artículo se puede identificar por:

- una *ID*
- un *Nombre*
- un *Título*
- y una *Abstracción*.

La lista puede ser filtrada usando el control en la esquina superior derecha.

Codificación de la Imagen

La sección *Codificación de la imagen* lista los formatos que reconoce por ambos el cliente y el servidor. Elija uno dependiendo de sus requerimientos de precisión de imagen.

Truco

Codificación de la Imagen

Normalmente, encontrará que un servidor WMS le ofrece la opción de codificación de la imagen en JPEG o PNG. JPEG es un formato de compresión con pérdida, mientras que PNG reproduce fielmente los datos crudos raster.

Utilizar JPEG si se espera que los datos WMS sean de naturaleza fotográfica y/o no le importa cierta perdida de calidad de la imagen. Esta disyuntiva típicamente reduce en cinco veces la necesidad de transferencia de datos en comparación con PNG.

Utilice PNG si desea representaciones precisas de los datos originales y no le importa el incremento de los requisitos de transferencia de datos.

Opciones

El área de Opciones del cuadro de diálogo proporciona medios para configurar las solicitudes de WMS. Puede definir:

- *Tamaño de tesela* si quiere establecer el tamaño de tesela (p.ej., 256x256) para dividir la solicitud WMS en varias solicitudes.
- *Petición de tamaño de paso*: si desea reducir el efecto de las etiquetas cortadas en los bordes de los mosaicos, aumentar el tamaño del paso crea solicitudes más grandes, menos mosaicos y menos bordes. El valor predeterminado es 2000.
- La etiqueta *Número máximo de resultados de GetFeatureInfo*: especifica el número máximo de resultados devueltos por una solicitud GetFeatureInfo para las capas que se están cargando. Permite anular el *valor predeterminado de la conexión.* para capas específicas.
- Cada capa WMS puede presentarse en múltiples SRC, dependiendo de la capacidad del servidor WMS. Si selecciona un WMS de la lista, aparecerá un campo con la proyección por defecto proporcionada por el servidor web. Pulse el widget Seleccionar SRC para sustituir la proyección por defecto del WMS por otra SRC soportada por el servidor WMS.

  Aparecerá un cuadro de diálogo similar al que se muestra en Figura 6.4. La principal diferencia con la versión WMS del cuadro de diálogo es que solo se mostrarán los SRC admitidos por el servidor WMS.
- Finalmente puede activar *Usar leyenda contextual WMS* si el servidor WMS admite esta función. Entonces, solo se mostrará la leyenda relevante para la extensión de la vista de mapa actual y, por lo tanto, no se incluirán elementos de leyenda para elementos que no puede ver en el mapa actual.

You can select several layers at once, but only one image style per layer. When several layers are selected, they will be combined at the WMS server and transmitted to QGIS in one go, as a single layer. The default name is a slash (/) separated list of their original title. You can however opt to *Load as separate layers*, in which case each layer is added to the map canvas using its default name from the queried service.

Orden de la capa

La pestaña *Orden de Capa* enumera las capas seleccionadas disponibles en el servidor WMS conectado actualmente.

Las capas WMS renderizadas por un servidor se superponen en el orden indicado en la pestaña *Capas*, de arriba a abajo de la lista. Si desea cambiar el orden de superposición, puede usar los botones *Arriba* y *Abajo* de la pestaña *Orden de capa*.

Transparencia

La configuración de *Transparencia global* de *Propiedades de capa* está codificada para estar siempre activa, donde esté disponible.

19.1.5 Conjunto de teselas

Cuando se utilizan los servicios WMTS (WMS en caché), se puede navegar a través de la pestaña *Tilesets* proporcionada por el servidor. En esta tabla se muestra información adicional como el tamaño de los mosaicos, los formatos y los CRS compatibles.

```
# example of WMTS service

https://opencache.statkart.no/gatekeeper/gk/gk.open_wmts?service=WMTS&
↪request=GetCapabilities
```

Seleccionando una capa para cargar, también es posible aplicar un *Método de interpretación*, convirtiéndola en una capa ráster de tipo flotante de banda única, lista para ser estilizada utilizando los *renderizadores ráster* habituales en QGIS.

En combinación con esta función, puede utilizar el deslizador de escala de baldosas seleccionando *Ver* ► *Paneles* (o *Configuración* ► *Paneles*), y seleccionando *Panel de escala de teselas*. Esto le da las escalas disponibles del servidor de azulejos con un bonito deslizador acoplado.

19.1.6 Utilizar la herramienta de Identificar objetos espaciales

Una vez que haya añadido un servidor WMS, y si alguna capa de un servidor WMS es consultable, puede entonces utilizar la herramienta  Identificar objetos espaciales para seleccionar un píxel del lienzo del mapa. Una consulta se hace al servidor WMS por cada selección realizada. El resultado de la consulta es regresada en texto plano. El formato de este texto es dependiente del servidor WMS particular utilizado.

Selección de Formato

Si múltiples formatos de salida son reconocidos por el servidor, una lista desplegable con formatos admitidos se añade automáticamente al diálogo de resultados identificados y el formato seleccionada puede ser almacenado en el proyecto para la capa.

Usar formato GML

La herramienta  Identificar reconoce la respuesta del servidor WMS (GetFeatureInfo) en formato GML (se llama Objeto espacial en la GUI de QGIS en este contexto). Si el formato «Objeto espacial» es admitido por el servidor y seleccionado, los resultados de la herramienta de identificador son objetos vectoriales, como de una capa vectorial regular. Cuando un objeto espacial es seleccionado en el árbol, este resalta en el mapa y se puede copiar a la papelera y pegar a otra capa vectorial. Vea el ejemplo de configuración de UMN Mapserver abajo que admite GetFeatureInfo en formato GML.

```
# in layer METADATA add which fields should be included and define geometry␣
↪(example):

"gml_include_items"   "all"
"ows_geometries"      "mygeom"
"ows_mygeom_type"     "polygon"

# Then there are two possibilities/formats available, see a) and b):

# a) basic (output is generated by Mapserver and does not contain XSD)
# in WEB METADATA define formats (example):
"wms_getfeatureinfo_formatlist" "application/vnd.ogc.gml,text/html"

# b) using OGR (output is generated by OGR, it is sent as multipart and contains␣
↪XSD)
# in MAP define OUTPUTFORMAT (example):
OUTPUTFORMAT
    NAME "OGRGML"
    MIMETYPE "ogr/gml"
    DRIVER "OGR/GML"
    FORMATOPTION "FORM=multipart"
END

# in WEB METADATA define formats (example):
"wms_getfeatureinfo_formatlist" "OGRGML,text/html"
```

19.1.7 Ver propiedades

Una vez que haya añadido un servidor WMS, puede ver sus propiedades haciendo clic con el botón derecho sobre él en la leyenda y seleccionando *Propiedades*. Las propiedades de las capas WMS/WMTS son muy parecidas a las propiedades de las capas ráster, por lo que encontrará una descripción detallada en *Dialogo de Propiedades Ráster*. Sin embargo, existen algunas diferencias que se explican a continuación.

Propiedades de la información

Pestaña de Metadatos

La pestaña *Metadatos* muestra una gran cantidad de información acerca del servidor WMS, generalmente obtenida de la declaración de capacidades de ese servidor. Muchas definiciones pueden ser extraídas mediante la lectura del estándar WMS (vea OPEN-GEOSPATIAL-CONSORTIUM en *Referencias bibliográficas y web*), pero aquí hay algunas definiciones útiles:

- **Propiedades del servidor**
  - **Versión WMS —** La versión WMS implementada por el servidor.
  - **Formatos de Imagen —** La lista de MIME-types que el servidor puede responder con la elaboración del mapa. QGIS reconoce cualquier formato las bibliotecas Qt subyacentes con que fueron construidas, que es típicamente por lo menos `image/png` y `image/jpeg`.
  - **Formato de Identificación —** La lista de tipos MIME, el servidor puede responder, cuando utilice la herramienta de Identificación. Actualmente, QGIS reconoce el tipo `texto plano`.
- **Propiedades de la capa**
  - **Seleccionar —** Sea o no esta capa seleccionada cuando su servidor fue añadido a este proyecto.
  - **Visible —** Si la capa seleccionada es o no visible en la leyenda (aun no utilizada en esta versión de QGIS).
  - **Poder Identificar —** Sea o no esta capa regresará algunos resultados cuando la herramienta de identificar se utilice en él.
  - **Puede ser transparente —** Si esta capa puede ser representada o no con transparencia. Esta versión de QGIS siempre usará transparencia si este es `Si` y la codificación de la imagen admite la transparencia.
  - ** Puede Acercar zum ** — Si en esta capa se puede o no hacer zum en el servidor. Esta versión de QGIS asume que todas las capas WMS tienen este conjunto de `Si`. Capas deficientes pueden ser presentadas de manera extraña.
  - **Conteo en Cascada —** Los servidores WMS pueden actuar como proxy para otros servidores WMS para obtener datos ráster de una capa. Esta entrada muestra el número de veces que se remitió la solicitud de esta capa para ver a los servidores WMS para obtener un resultado.
  - **Ancho fijo, altura fija —** Si esta capa o no tiene fijas las dimensiones en píxeles de origen. Esta versión de QGIS asume que todas las capas WMS tienen este conjunto a nada. Capas deficientes pueden ser presentadas de forma extraña.
  - **Recuadro delimitador WGS 84 —** El recuadro delimitador de la capa, en coordenadas WGS 84. Algunos servidores WMS no establecen esto correctamente (por ejemplo, utilizan coordenadas UTM en su lugar). Si éste es el caso, entonces la vista inicial de la capa puede aparecer muy “lejana” en QGIS. El webmaster de WMS debería ser informado de este error, que probablemente conocen como los elementos XML de WMS `LatLonBoundingBox`, `EX_GeographicBoundingBox` o el *BoundingBox*` de CRS:84.
  - **Disponible en SRC —** Las proyecciones que esta capa puede representar por el servidor WMS. Éstos se enumeran en el formato nativo de WMS.
  - **Disponible en estilo —** Los estilos de imagen que esta capa puede representar por el servidor WMS.

Propiedades temporales

Las *propiedades temporales* de raster (en concreto *Control Temporal Dinámico*) pueden establecerse para las capas WMS y WMTS. Por defecto, cuando se añade al proyecto una capa WMS o WMTS con dimensión temporal, se indica en el panel *Capas* con el icono  Capa Temporal junto a ella. Sus propiedades *Temporal* se establecen por defecto en el modo temporal *Automático*, lo que significa que la capa seguirá por defecto el rango temporal actual del controlador temporal.

A continuación, puede optar por mostrar un valor temporal estático específico para la capa desmarcando *Dynamic Temporal Control* y eligiendo una opción en *Static WMS-T Temporal Range*:

- *Servidor predeterminado*
- *Fecha predefinida* con un servidor que exponga datos para rangos temporales no contiguos o *Rango predefinido* con un servidor que exponga un rango de fechas disponibles. En este último caso son necesarias las etiquetas *Fecha de inicio* y *Fecha de final*. Su formato previsto puede deducirse de la opción de hora de referencia (véase más adelante). dependiendo de si el proveedor dispone de datos para un periodo contiguo o no
- *Seguir el rango temporal del proyecto* definido en el diálogo de propiedades del proyecto

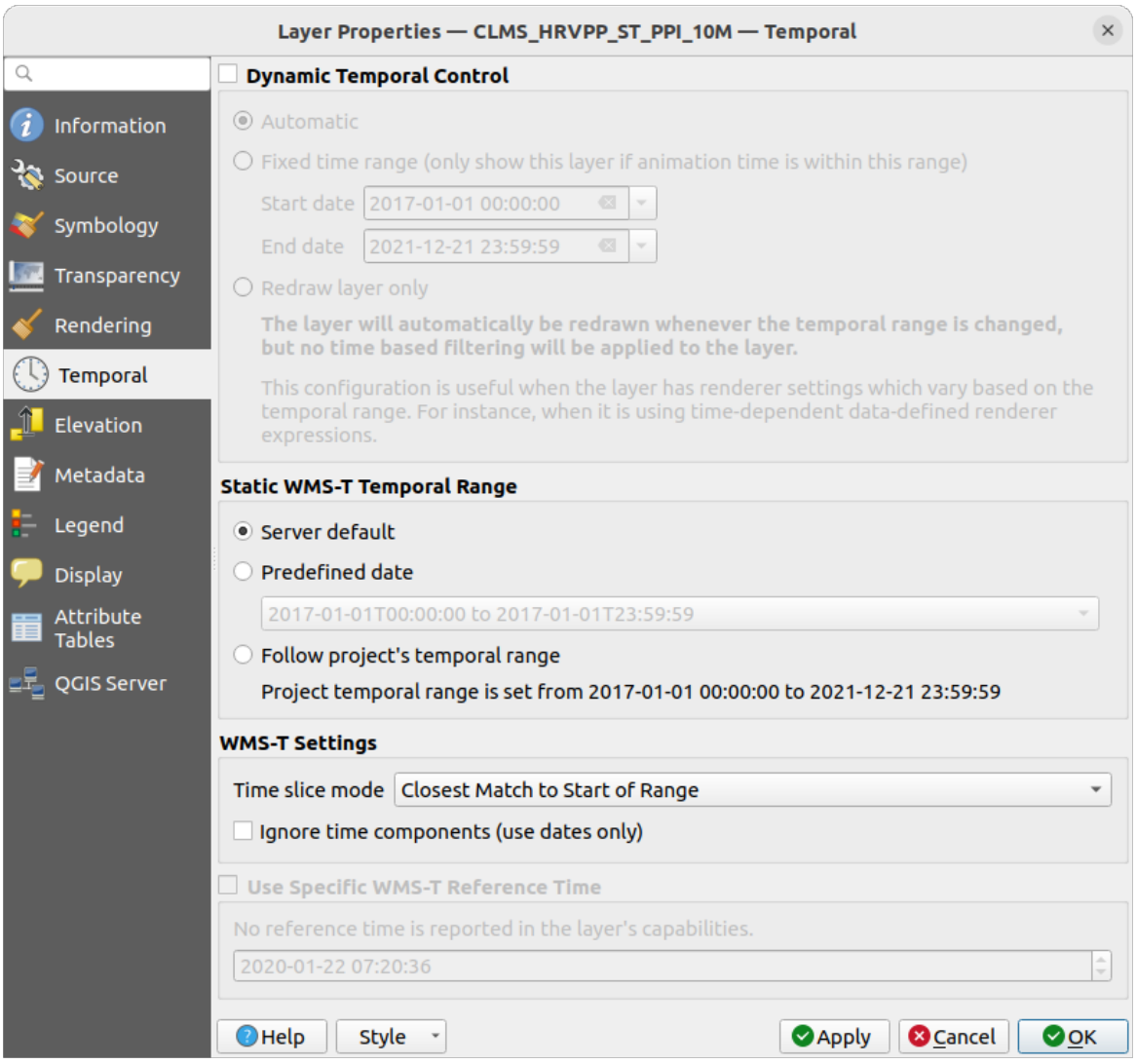


Figura 19.4: Propiedades temporales de una capa WMTS

Independientemente del control de datos temporales que se utilice, existen algunos *Configuración WMS-T* para ayudar a mostrar los datos temporales correctos:

- *Modo de corte de tiempo* que puede ser:
  - *Utilizar todo el rango temporal*.
  - *Emparejar al principio del rango*
  - *Emparejar al final del rango*
  - *La coincidencia más cercana al inicio del intervalo*
  - *Closest match to end of range*
- *Ignorar componentes temporales (usar sólo fechas)*: Si se marca, se descartará el componente de tiempo de las consultas temporales y sólo se utilizará el componente de fecha en las peticiones al servidor.

También puede  *Usar hora de referencia WMS-T específica* elegida a partir de las horas indicadas en las capacidades de la capa. Conveniente para servidores que exponen un conjunto no contiguo de instancias de fecha y hora (en lugar de un intervalo de fechas).

Propiedades del servidor QGIS

Además de las *propiedades* de la capa raster, al publicar una capa WMS/WMTS con QGIS Server se mostrarán las siguientes opciones:

- *Capa de impresión WMS*: Permite establecer la capa WMS alternativa que debe utilizarse para la impresión (en la respuesta GetProjectSettings). Conveniente para las capas WMTS que generalmente no son adecuadas para la impresión.
-  *Publicar uri de origen de datos WMS/WMTS*: Permite a un cliente web obtener los datos WMS/WMTS directamente.
-  *Anunciar como capa de fondo*.

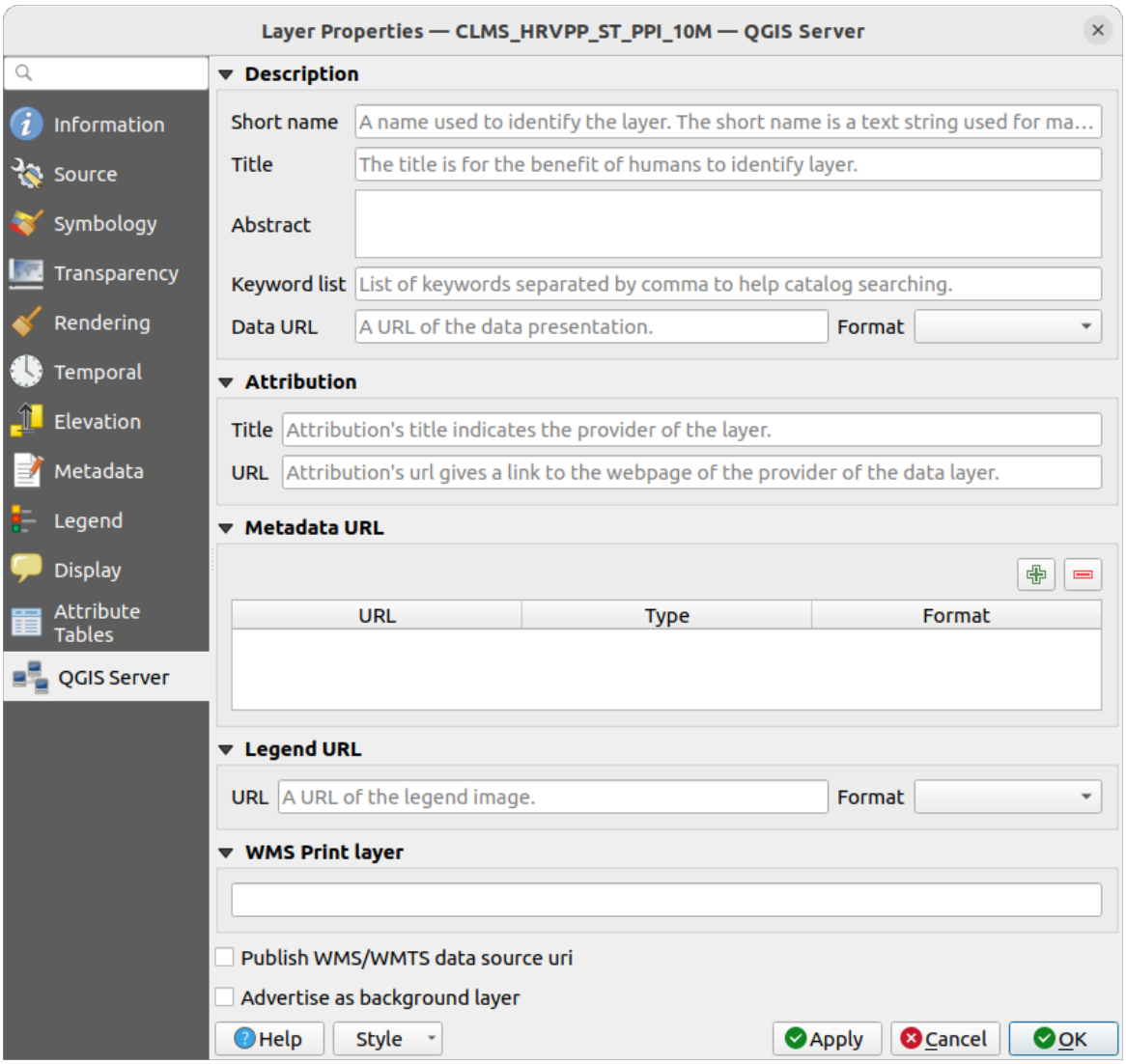


Figura 19.5: Propiedades del servidor QGIS de una capa WMS/WMTS

19.1.8 Mostrar el gráfico de la leyenda de WMS en la tabla de contenido y diseño

El proveedor de datos QGIS WMS puede mostrar un gráfico de leyenda en la lista de capas de la tabla de contenido y en el diseño de impresión. La leyenda de WMS se mostrará solo si el servidor WMS tiene la capacidad GetLegendGraphic y la capa tiene la URL getCapability especificada, por lo que también debe seleccionar un estilo para la capa.

Si hay definida una legendGraphic, ésta se mostrará debajo de la capa. Es pequeña y hay que hacer clic sobre ella para abrirla en tamaño real (debido a una limitación de la arquitectura de QgsLegendInterface). Al hacer clic en la leyenda de la capa se abrirá un cuadro con la leyenda a la máxima resolución.

En el diseño de impresión, la leyenda se integrará en su dimensión original (descargada). La resolución del gráfico de la leyenda se puede establecer en las propiedades del elemento en *Leyenda -> WMS LegendGraphic* para que coincida con sus requisitos de impresión.

La leyenda mostrará información contextual basada en su escala actual. La leyenda WMS se muestra sólo si el servidor WMS tiene capacidad GetLegendGraphic y la capa tiene definida una url getCapability, para lo que se debe seleccionar un estilo.

19.2 WCT Cliente

Un Web Coverage Service (WCS) proporciona acceso a los datos ráster en formas que son útiles para la representación del lado cliente, como datos de entrada en los modelos científicos, y para otros clientes. El WCS se puede comparar con la WFS y el WMS. Como WMS y WFS instancias de servicios, un WCS permite a los clientes elegir partes de las explotaciones de información de un servidor basado en restricciones espaciales y otros criterios de consulta.

QGIS tiene un proveedor WCS nativo y reconoce ambas versiones 1.0 y 1,1 (que son significativamente diferentes), pero actualmente se prefiere 1.0, porque 1.1 tiene muchos problemas (por ejemplo, cada servidor lo implementa de diferente forma con varias particularidades).

El proveedor de WCS nativo se encarga de todas las solicitudes de red y utiliza las configuraciones de red estándar de QGIS (especialmente de proxy). También es posible seleccionar el modo de caché (“siempre caché”, “preferentemente caché”, “preferentemente red”, “siempre red”). El proveedor también es compatible con la selección de tiempo de la posición, si el servidor ofrece el dominio temporal.

Loading a WCS Layer

To be able to load a WCS Layer, first create a connection to the WCS server:

1. Abra el cuadro de diálogo *Administrador de fuente de datos* presionando el botón
   Abrir Administrador de fuentes de datos
2. Enable the *WCS* tab
3. Click on *New…* to open the *Create a New WCS Connection* dialog

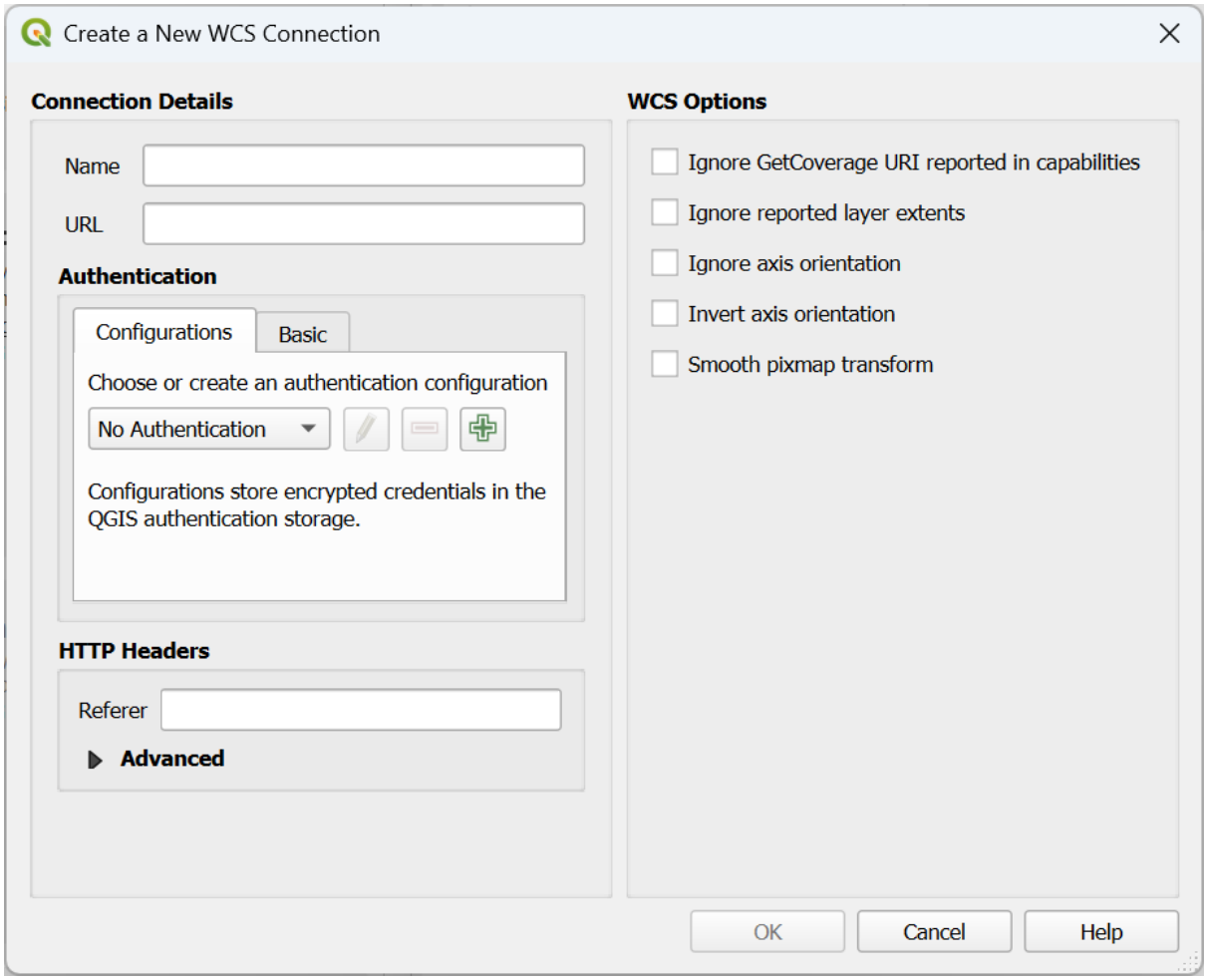


Figura 19.6: Creating a connection to a WCS server

- *Name*: A name for the connection. This name will be used in the Server Connections drop-down box so that you can distinguish it from other WCS servers.
- *URL*: URL del servidor que proporciona los datos. Debe ser un nombre de host que se pueda resolver, el mismo formato que usaría para abrir una conexión telnet o hacer ping a un host, es decir, solo la URL base. Por ejemplo, no debería tener fragmentos como `request=GetCapabilities` o `version=1.0.0` en su URL.
- *Autenticación* (opcional): usanr una *configuración almacenada* o una autenticación básica con *Nombre de Usuario* y *Contraseña*.

> **Advertencia**
>
> Al ingresar **nombre de usuario** y **contraseña** en la pestaña *Autenticación* mantendrá las credenciales desprotegidas en la configuración de la conexión. Esas **credenciales serán visibles** si, por ejemplo, compartiste el archivo del proyecto con alguien. Por lo tanto, es aconsejable guardar sus credenciales en una *Configuración de autenticación* en su lugar (pestaña *Configuraciones*). Ver :ref:‘authentication_index` para más detalles.

- *Referente* HTTP
- ☐ *Ignore GetCoverage URI reported in capabilities*: if checked, use given URI from the *URL* field above.
- ☐ *Ignore reported layer extents*: because the extent reported by raster layers may be smaller than the actual area which can be rendered (notably for WCS servers with symbology which takes more space

than the data extent), check this option to avoid cropping raster layers to their reported extents, resulting in truncated symbols on the borders of these layers.

- ☐ *Ignore axis orientation*
- ☐ *Invertir orientación del eje*
- ☐ *Suavizar transformación de pixel de mapa*

4. Presiona *Aceptar* para crear la conexión.

Tenga en cuenta que también se reconoce cualquier configuración de proxy que haya establecido en sus preferencias. Tenga en cuenta también que es posible *Cargar* los parámetros de conexión desde un archivo `.XML` o *Guardarlos* en un archivo `.XML`.

Now we are ready to load WCS layers from the above connection.

19.3 Cliente WFS y WFS-T

En QGIS, una capa WFS se comporta como cualquier otra capa vectorial. Puede identificar y seleccionar características, y ver la tabla de atributos. QGIS admite WFS 1.0.0, 1.1.0, 2.0 y OGC API - Características (OAPIF), incluida la edición (a través de WFS-T). QGIS también admite la descarga en segundo plano y el renderizado progresivo, el almacenamiento en caché en disco de las características descargadas y la autodetección de versiones.

Las capas de los servidores que implementan OGC API - Objetos espaciales- Parte 4: Crear, Reemplazar, Actualizar y Borrar pueden ponerse en modo edición para permitir crear, modificar y borrar objetos espaciales. Tenga en cuenta que cada objeto espacial creado/modificado/eliminado requiere una solicitud de red dedicada, por lo que el rendimiento podría verse afectado en caso de modificación simultánea de cientos o más objetos espaciales a la vez.

En general, agregar una capa WFS es muy similar al procedimiento que se usa con WMS. No hay servidores predeterminados definidos, por lo que debe agregar los suyos. Puede encontrar servidores WFS utilizando el *Complemento MetaSearch* o su motor de búsqueda web favorito. Hay una serie de listas con URL públicas, algunas de las cuales se mantienen y otras no.

Cargar una capa WFS

Como ejemplo, utilizamos el servidor WFS de Gateway Geomatics y mostramos una capa.

```
https://demo.gatewaygeomatics.com/cgi-bin/wfs_gateway?REQUEST=GetCapabilities&
↪VERSION=1.0.0&SERVICE=WFS
```

Para poder cargar una capa WFS, primero cree una conexión al servidor WFS:

1. Abra el cuadro de diálogo *Administrador de fuente de datos* presionando el botón 
Abrir Administrador de fuentes de datos
2. Activar la pestaña  *WFS / OGC API - Objetos espaciales*
3. Click en *Nuevo…* para abrir el diálogo *Crear una nueva conexión WFS*
4. In the *Connection Details* group, enter `Gateway Geomatics` as name
5. Introduce la URL (ver arriba)
6. Optionally, enter the *Authentication* settings

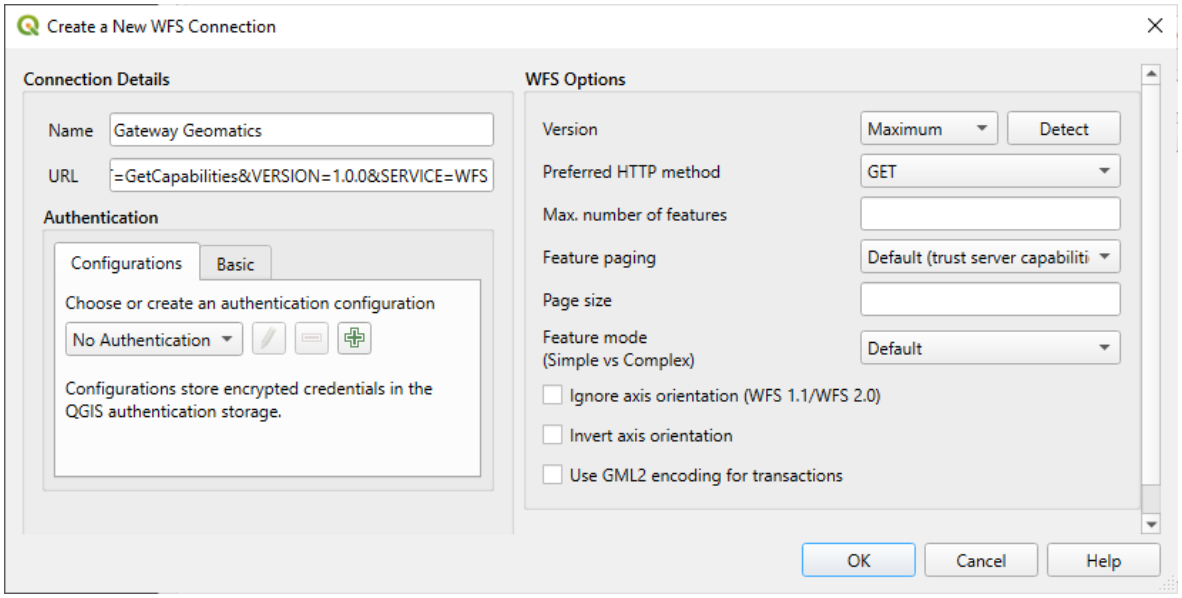


Figura 19.7: Creando una conexión a un servidor WFS

> **Nota**
>
> In case of an OGC API - Features (OAPIF), the URL to provide should be the landing page, i.e., the main page from which it is possible to navigate to all the available service endpoints.

7. In the *WFS Options* group, you can:
   - Indique la versión WFS del servidor. Si no lo conoce, presione el botón *Detectar* para recuperarlo automáticamente.
   - Select the *Preferred HTTP method* to use for requests. The default is *GET*, but you can also select *POST*.
   - Define the *Maximum number of features* retrieved in a single GetFetFeature request. If empty, no limit is set.
   - *Feature mode (simple vs complex)*: helps determine how to handle servers serving data with simple or complex GML schemas. Available options are:
     - *Default*: if the server supports editing capabilities, this is the same as using *Simple Features*. If not, it fallbacks to using *Complex Features* mode.
     - *Simple Features*: returns data as simple features only
     - *Complex Features*: uses the *OGR GMLAS <https://gdal.org/en/stable/drivers/vector/gmlas.html>*, potentially returning complex features, depending on the data schema, where nested data structures are exposed by QGIS as JSON.
   - Depending on the WFS version, indicate whether to:
     - Enable *Feature paging* and specify the maximum number of features to retrieve with *Page size*. If no limit is defined, then the server default is applied.
     - Forzar a *Ignorar orientación del eje (WFS 1.1/WFS 2.0)*
     - *Invertir orientación del eje*.
     - *Utilizar codificación GML2 para transacciones*.

> **Advertencia**
>
> Al ingresar **nombre de usuario** y **contraseña** en la pestaña *Autenticación* mantendrá las credenciales desprotegidas en la configuración de la conexión. Esas **credenciales serán visibles** si, por ejemplo, compartiste el archivo del proyecto con alguien. Por lo tanto, es aconsejable guardar sus credenciales en una *Configuración de autenticación* en su lugar (pestaña *Configuraciones*). Ver :ref:‘authentication_index` para más detalles.

8. Presiona *Aceptar* para crear la conexión.

Tenga en cuenta que también se reconoce cualquier configuración de proxy que haya establecido en sus preferencias. Tenga en cuenta también que es posible *Cargar* los parámetros de conexión desde un archivo `.XML` o *Guardarlos* en un archivo `.XML`.

Ahora estamos listos para cargar capas WFS desde la conexión anterior.

1. Seleccione “Gateway Geomatics” en la lista desplegable *Conexiones de servidores*.
2. Click *Conectar*
3. Selecciona la capa *Parks* en la lista
4. También puede elegir si desea:
   - *Solicitar solo objetos que se superpongan a la extensión de la vista*
   - *Cambiar…* el SRC de la capa por cualquier otro soportado por el servicio.
   - o crear una consulta para especificar objetos espaciales concretos a recuperar del servicio: haga doble clic en la fila de la capa para abrir el cuadro de diálogo *Compositor de consultas SQL*. Ese cuadro de diálogo proporciona widgets para escribir una consulta SQL avanzada basada en las tablas y columnas disponibles del servicio, con ordenación y filtrado y un montón de funciones SQL, predicados espaciales y operadores.

     La consulta que construya aparecerá después de la validación en la columna *SQL* dentro de la tabla *WFS / OGC API - Objetos espaciales*, y la capa filtrada mostrará el icono junto a ella en el panel *Capas*. De este modo, es posible ajustar la consulta en cualquier momento.
5. Click *Add* to add the layer to the map, using its default name from the queried service.

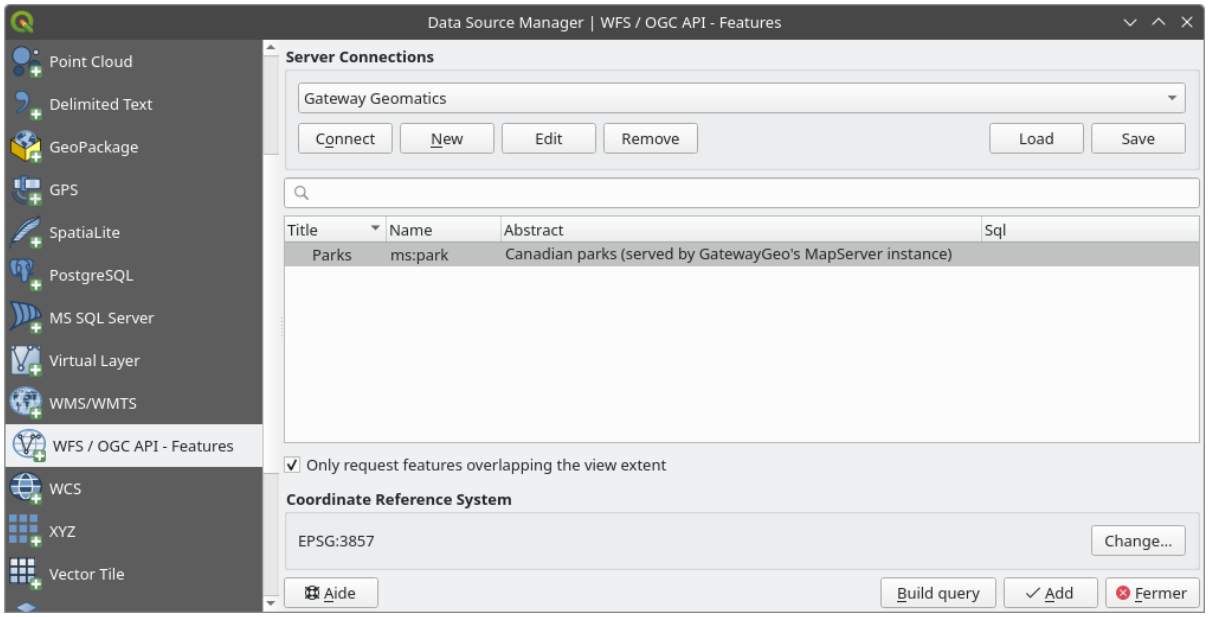


Figura 19.8: Añadir una capa WFS

Notará que el progreso de la descarga se visualiza en la parte inferior izquierda de la ventana principal de QGIS. Una vez que la capa está cargada, puede identificar y seleccionar un par de entidades y ver la tabla de atributos.

19.4 SensorThings

QGIS admite conexiones con OGC SensorThings API, un estándar que proporciona un marco abierto y unificado para interconectar dispositivos de detección, datos y aplicaciones de IoT a través de la web. Se trata de un estándar abierto que aborda la interoperabilidad sintáctica y semántica del Internet de las cosas. Se basa en el modelo de datos Observations and Measurements, un modelo estandarizado para observaciones y para características involucradas en el muestreo al realizar observaciones.

19.4.1 Configuración de la conexión

Para añadir datos de SensorThings a QGIS, utilice la pestaña  *SensorThings* en el cuadro de diálogo *Administrador de fuentes de datos*.

Para establecer una nueva conexión, pulse *Nuevo* (o *Nueva conexión SensorThings* en el panel Navegador) e introduzca *Nombre* y *URL*. Opciones avanzadas, como *autenticación* y una etiqueta *Referer*, también se puede configurar.

Pulse *Aceptar* para establecer la conexión. A continuación, podrá:

- *Editar* la configuración de conexión de SensorThings
- *Eliminar* la conexión SensorThings

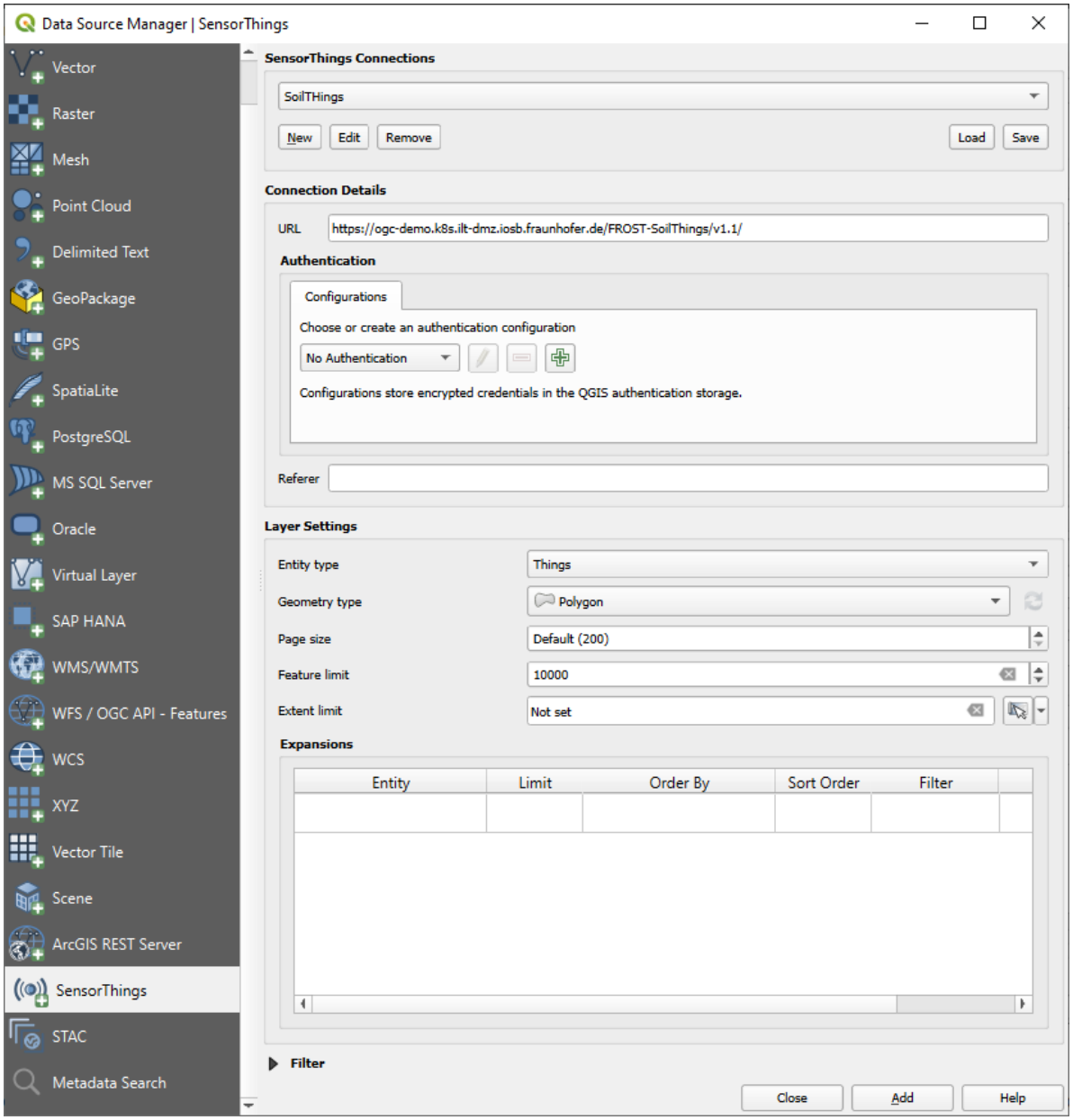


Figura 19.9: Cuadro de diálogo Conexión de SensorThings

Las configuraciones se pueden guardar en un archivo `.XML` (*Guardar*) a través de la entrada *SensorThings* en el cuadro de diálogo *Administrador de fuentes de datos* o su menú contextual en el panel *Navegador* (*Guardar conexiones*). Del mismo modo, las configuraciones se pueden añadir desde un archivo (*Cargar*).

19.4.2 Cargando datos de SensorThings

Las relaciones entre las capas (las denominadas entidades) almacenadas en un conjunto de datos de SensorThings se expresan en el siguiente diagrama.

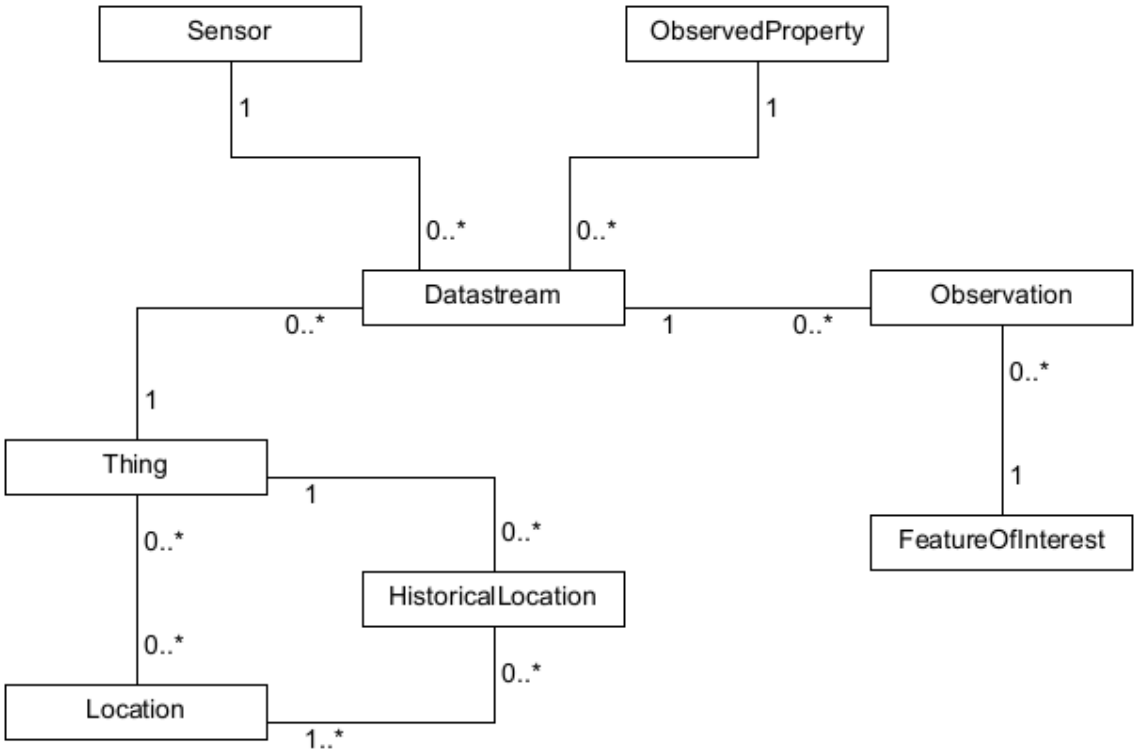


Figura 19.10: Modelo de datos Observaciones y mediciones (Fuente: «Tutorial sobre redes de sensores de SIST <https://sist.pages.in2p3.fr/anf21-sos52north/hands-on/09_sta-example-request/>»_)

Se puede cargar cualquier tipo de entidad en QGIS, pero no todas son datos espaciales. Para cargar una entidad, hay *Configuración de capa* que se puede configurar:

- *Tipo de entidad*: la entidad que se cargará desde el modelo de datos como capa en QGIS.
- *Tipo de geometría*: el tipo de geometría de la entidad seleccionada que se va a cargar. Pulse  Comprobar tipos disponibles para limitar la lista a los tipos de geometría realmente compatibles.
- *Tamaño de página*
- *Límite de objeto espacial* establece el número máximo de objetos espaciales que se pueden solicitar al servicio.
- *Límite de extensión* establece un límite máximo de extensión para la capa, de modo que solo se solicitan las características que se encuentran dentro de la extensión.
- *Expansiones*: El modelo de datos de SensorThings proporciona un mecanismo de expansión de los resultados a entidades relacionadas, similar a cómo se unen las tablas en una base de datos relacional. Con este enfoque, puede expandir la capa seleccionada para incluir datos de otros elementos. Esto aplanará la relación, creando tantas características principales como secundarias, y se añadirán propiedades adicionales como columnas en la tabla de atributos.

Utilice *Filtro* para crear una consulta que filtre los datos, utilizando la sintaxis de filtro de SensorThings.

> **Nota**
>
> Las opciones de configuración y filtrado anteriores también están disponibles para su actualización en el cuadro de diálogo de propiedades de la capa, pestaña *Fuente*, una vez cargadas en QGIS.

Pulse *Añadir* para cargar el tipo de entidad seleccionado como capa en QGIS.

19.4.3 Trabajar con una capa vectorial de SensorThings

Una capa SensorThings se carga en QGIS como una capa vectorial. Como tal, muestra las mismas pestañas en las *propiedades de la capa.* y permite las mismas interacciones de funciones utilizando las herramientas de selección o identificación. Sin embargo, hay algunas particularidades que debes tener en cuenta al trabajar con datos de SensorThings.

Debido al modelo de datos, la propiedad «result» de una observación de SensorThings es un campo de cadena. Si desea utilizar su representación numérica, por ejemplo, en un estilo graduado, utilice una expresión para convertir el valor a real y try() en caso de que esto falle (por ejemplo, `try( to_real("Observation_result"), Null)`).

Si desea crear un gráfico con las observaciones de una o varias ubicaciones, puede instalar el complemento de QGIS *Data Plotly*.

1. Ahora seleccione las observaciones en una ubicación puntual en la vista del mapa.
2. Abre el panel plotly y activa la casilla de verificación *Usar solo objetos espaciales seleccionados*.
3. Seleccione en la columna x una propiedad de fecha y hora y en la columna y la etiqueta *Observation_result*. Esto trazará las observaciones en esa ubicación a lo largo del tiempo.
4. Verifique para filtrar por una sola propiedad observada.
5. Observe que el gráfico cambia tan pronto como selecciona otras ubicaciones en el mapa.

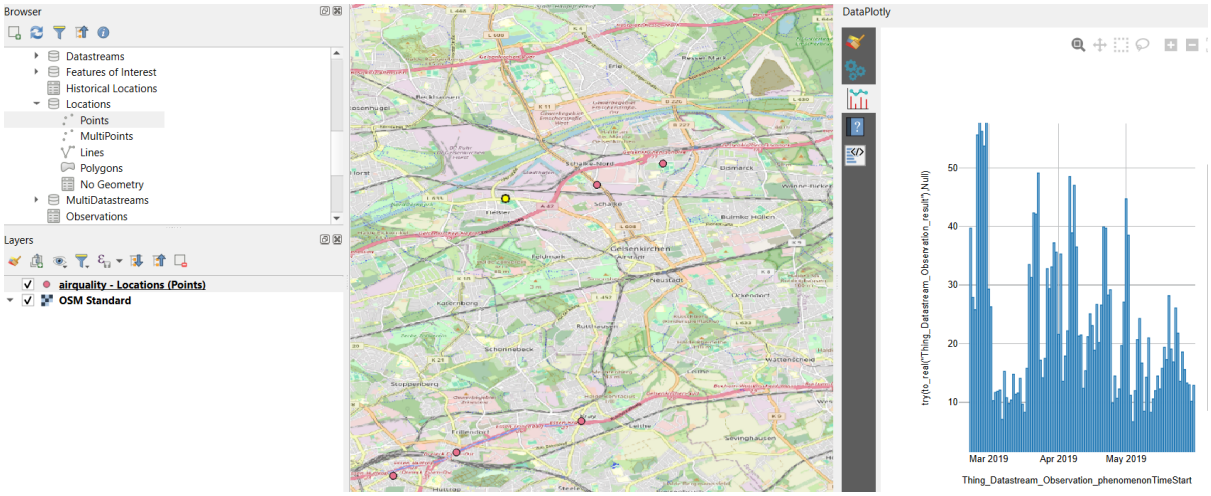


Figura 19.11: Utilice Data plotly para graficar las observaciones de la calidad del aire en una ubicación.

19.5 STAC (SpatioTemporal Asset Catalogs)

STAC (SpatioTemporal Asset Catalog) is a specification for describing geospatial data in a consistent and accessible way. It defines a standard structure for organizing and indexing spatial-temporal assets such as satellite imagery, drone photos, or sensor data so that you can search, preview, and use these datasets across different platforms and tools.

STAC enables QGIS users to connect to and browse geospatial datasets, either from static catalogs or dynamic APIs, and download or stream assets such as imagery or other spatial data.

More information at https://stacspec.org/en.

Key components of STAC in QGIS include:

- **STAC Items**: The basic unit of a STAC catalog. Each item represents one geospatial asset or a group of related assets (e.g. image bands) at a specific time and location. It includes metadata such as geometry, datetime, links to assets (e.g., TIFFs, JSON, COG), and properties.
- **STAC Collections**: A grouping of items that share common characteristics and metadata (e.g. satellite mission). Collections may define spatial and temporal extents, licensing, and keywords.
- **STAC Catalogs**: A hierarchical container that organizes items and/or collections. Catalogs allow navigation of STAC datasets but do not necessarily include search capabilities.
- **STAC API**: An implementation of the STAC specification that allows querying and filtering of STAC items using spatial and temporal based filters, as well as searching within specific collections. STAC APIs follow the OGC API - Features pattern and support dynamic access to datasets.

There is an important distinction between static STAC catalogs and STAC API endpoints:

- **Static STAC Catalogs**: are collections of JSON files without search capabilities. They can be browsed via the Browser panel.
- **STAC API**: provide search capabilities and can be accessed in QGIS through both the Browser panel and the Data Source Manager.

19.5.1 Configuración de la conexión

STAC connections can be added in QGIS using either the *Browser panel* or the *Data Source Manager*:

- **Browser Panel:** In the *Browser panel*, right-click on the *STAC* entry and select *New STAC Connection…*. In the dialog that appears, enter a *Name* for the connection, the *URL* of the STAC catalog and optionally fill in *Authentication* credentials and a *Referer*. Then click *OK*.

  Use this method for browsing static STAC catalogs that do not support search or filtering.

- **Data Source Manager:** For STAC APIs you can use *Data Source Manager* dialog.

  Open the *Data Source Manager*, choose the *STAC* tab and click the *New* button. Fill in the *Name* and *URL* fields, and (optional) the *Authentication* credentials and a *Referer*. Press *OK* and then *Connect* to establish the connection, after that you will be able to:

  - *Edit* the STAC connection settings
  - *Remove* the STAC connection

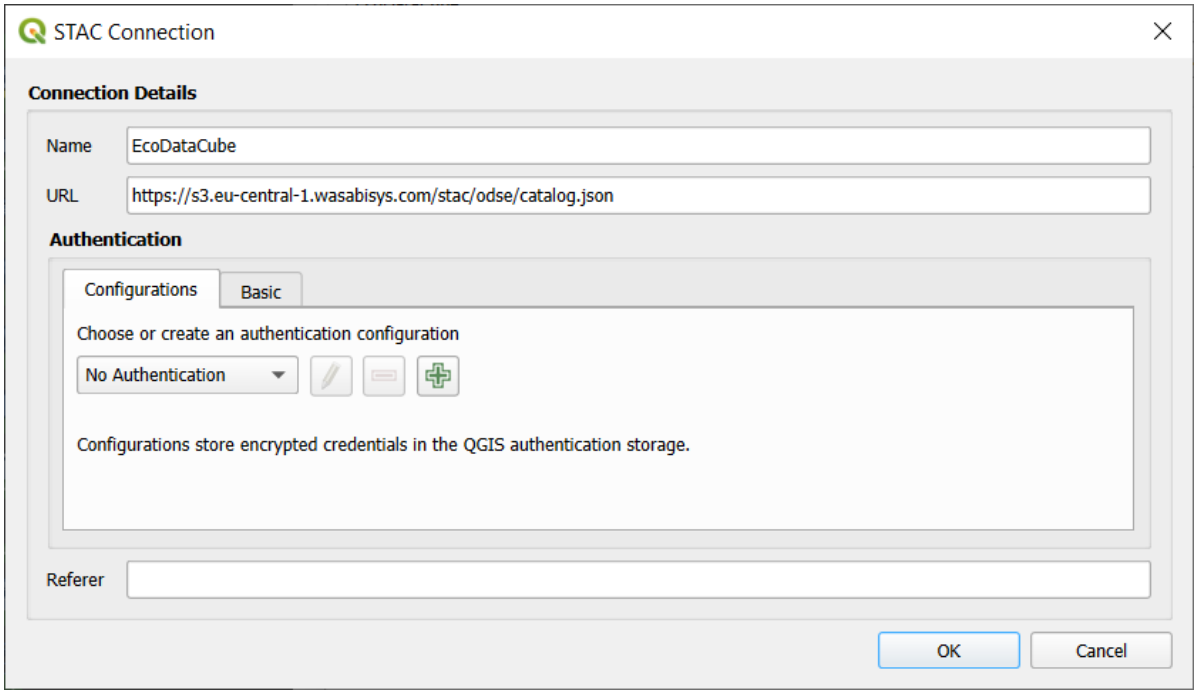


Figura 19.12: Creating a connection to a STAC server

19.5.2 Browsing STAC Catalogs

Static STAC catalogs are displayed as hierarchical structures within the *Browser panel*. Once connected, the STAC catalog appears under *STAC* in the *Browser* panel. You can expand the catalog node to see its *Collections*. Expanding a collection reveals the individual *Items* it contains.

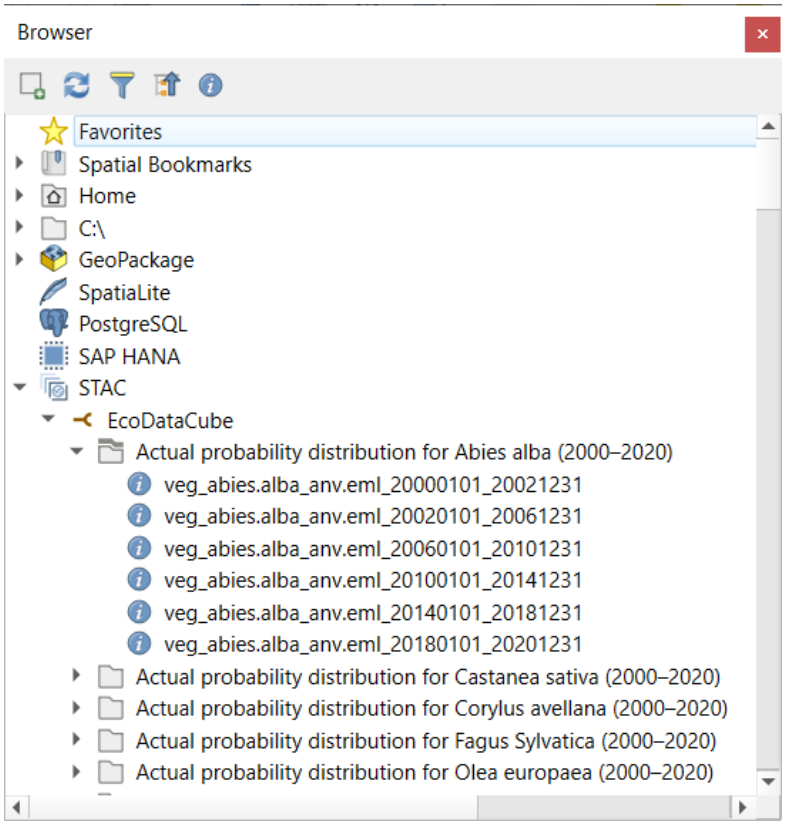


Figura 19.13: STAC connection expanded in the Browser, showing Collections and Items

Right-click any STAC *Item* and choose *Details…* to view its metadata. The details panel shows the item's JSON content and a map of its coverage. If an item's asset is a cloud-optimized format (e.g. a `.COG`), you can add it directly to the map canvas. Otherwise, *Download Assets…* to save it locally before use.

19.5.3 Filtering and Searching STAC Items

For STAC API endpoints, QGIS supports spatial and temporal filtering, as well as limiting searches to specific collections.

To search for items:

1. Open *Data Source Manager* and go to the *STAC*
2. Select the STAC connection you created earlier and click *Filters…*

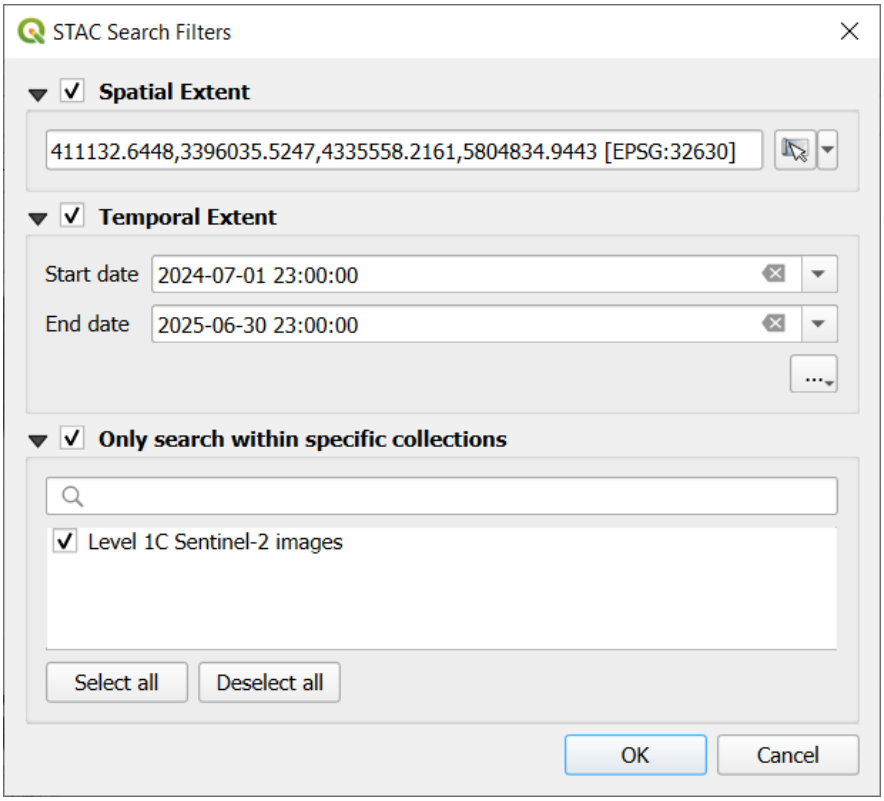


Figura 19.14: Define spatial and temporal filters for a STAC catalog search

3. In the filter dialog, you can:
   - Enable *Spatial Extent* to restrict results to a bounding box or drawn area.
   - Enable *Temporal Extent* and define a date or range.
   - Limit to specific collections using *Only search within specific collections*
4. After applying filters, results will be listed in the main *Data Source Manager* panel.
5. Optionally check *Show Footprints* to display the footprints of the results on the map.

Right-click a result item to access actions:

- *Zoom to Item*
- *Pan to Item*
- *Download Assets*
- *Details…*

If the item's asset requires download, use the *Download Assets* option.

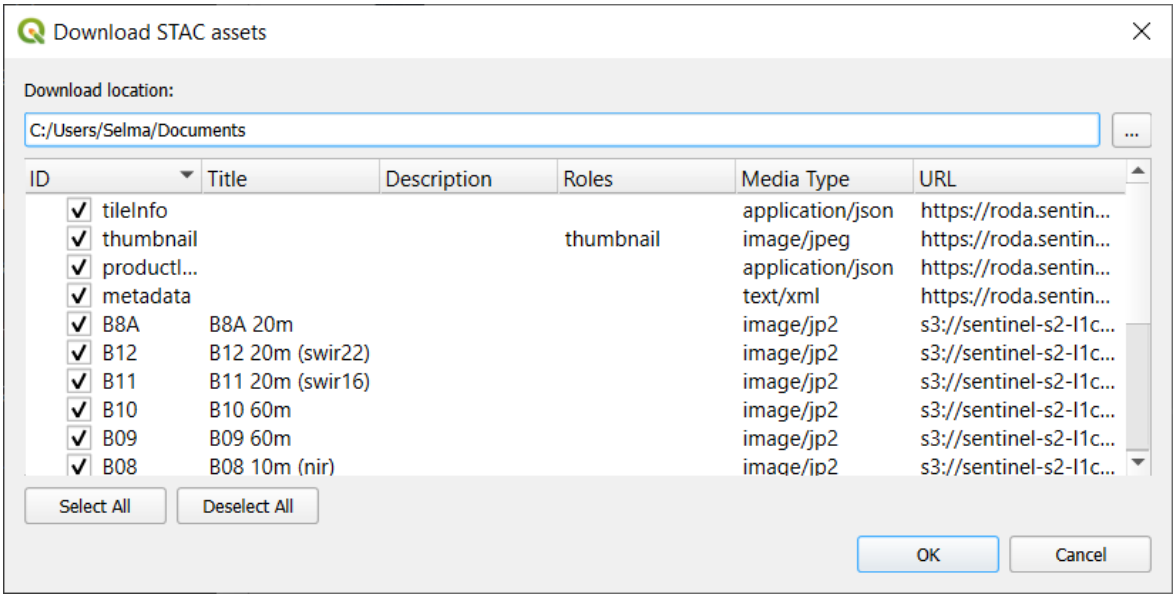


Figura 19.15: Download STAC item assets

Downloaded assets include the main dataset and any auxiliary files such as thumbnails or style files. After download, use standard QGIS tools (e.g., *Add Raster Layer…*) to load and display the data.

19.6 ArcGIS REST Server

An ArcGIS REST Server can host many different types of web services (feature service, map service, image service, …). Read *Uso de servidores REST de ArcGIS* for instructions to connect to an ArcGIS REST server.

When loading layers form an ArcGIS Feature Service, symbols are automatically converted to QGIS-native symbology, allowing QGIS to visually reproduce the same appearance as ArcGIS web maps.

Layers from an ArcGIS Feature Service can be editable in QGIS if:

1. The ArcGIS service has editing enabled
2. Your credentials grant write access
3. And the remote layer supports editing operations (Create, Update, or Delete capabilities). If these options are disabled on the server, the layer will be read-only in QGIS.

Editing these layers follow the *same procedure* as for other vector layers. To update your map with the latest changes, use *View* ► *Refresh* or press `F5`. If multiple users edit the same layer concurrently, it is recommended to refresh your data before performing edits to avoid conflicts.

CAPÍTULO 20

Trabajar con datos GPS

20.1 Introducción a los datos GNSS/GPS

20.1.1 ¿Qué es GPS?

GPS, el **Sistema de Posicionamiento Global**, es un sistema basado en satélites que permite a cualquier persona con un receptor GPS encontrar su posición exacta en cualquier parte del mundo. El GPS se utiliza como ayuda a la navegación, por ejemplo en aviones, barcos y excursionistas. El receptor GPS utiliza las señales de los satélites para calcular su latitud, longitud y (a veces) elevación. La mayoría de los receptores también tienen capacidad para almacenar:

- ubicaciones (conocidas como **puntas de vía**)
- secuencias de lugares que componen una **ruta** planificada
- y una **traza** del movimiento del receptor a lo largo del tiempo.

Los puntos de vía, las rutas y las trazas son los tres tipos básicos de características de los datos GPS. QGIS muestra los puntos de via en capas de puntos, mientras que las rutas y las trazas se muestran en capas de líneas.

> **Nota**
>
> QGIS también admite receptores GNSS. Pero seguimos usando el término GPS en esta documentación.

20.1.2 Transferencia o carga de datos GPS

Cargar un archivo GPX

Hay docenas de formatos de archivo diferentes para almacenar datos GPS. El formato que utiliza QGIS se llama GPX (formato de intercambio de GPS), que es un formato de intercambio estándar que puede contener cualquier número de waypoints, rutas y tracks en el mismo archivo.

Para cargar un archivo `GPX`:

1. Abra la pestaña *GPS* del cuadro de diálogo *Gestor de fuentes de datos*, es decir:
   - Haga clic en el botón Gestor de fuentes de datos abiertos de la barra de herramientas (o pulse `Ctrl+L`) y active la pestaña de destino
   - O seleccione *Capa* ► *Añadir Capa* ► *Añadir Capa GPX…*
2. Utilice el botón … Navegador situado junto a la opción *GPX conjunto de datos* para seleccionar el archivo GPX
3. Utilice las casillas de verificación para seleccionar los *Tipos de objetos espaciales* que desea cargar desde el archivo. Cada tipo de característica (*Puntos de vías*, *Trazas* o *Rutas*) se cargará en una capa separada.

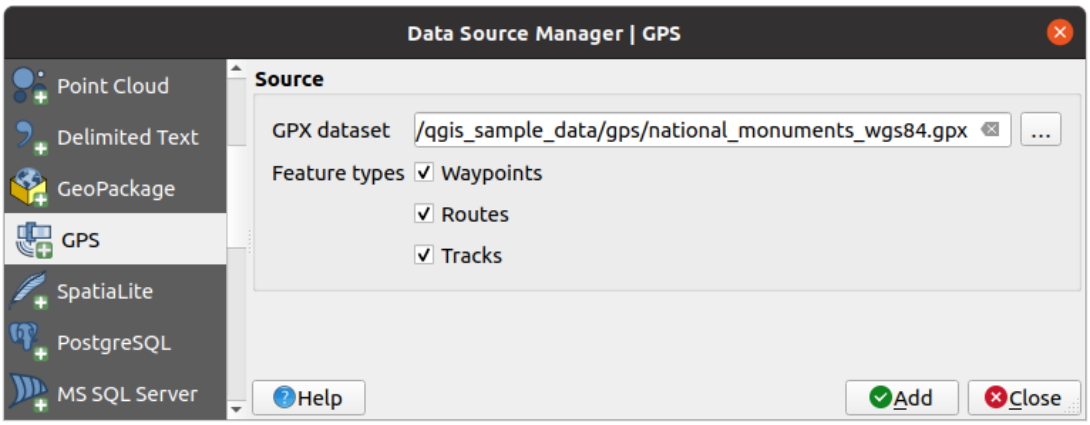


Figura 20.1: Diálogo de carga de datos GPS

Carga hacia o desde un dispositivo

Existen muchos tipos diferentes de dispositivos y formatos GPS. Dado que QGIS utiliza archivos GPX, se necesita una forma de convertir otros formatos de archivo GPS a GPX. QGIS puede hacerlo utilizando el programa gratuito GPSBabel. GPSBabel puede ayudarte a convertir waypoints, tracks y rutas entre receptores GPS populares como Garmin o Magellan y programas de mapas como Google Earth o Basecamp. Es compatible con cientos de receptores GPS y programas. También puede transferir datos GPS entre tu ordenador y un dispositivo GPS.

En *Configuración* ► *Opciones* ► *GPS* ► *GPSBabel*, QGIS le permite definir su propio tipo de dispositivo y establecer parámetros de conversión que más tarde podrían ser utilizados por el *Procesamiento de algoritmos GPS*.

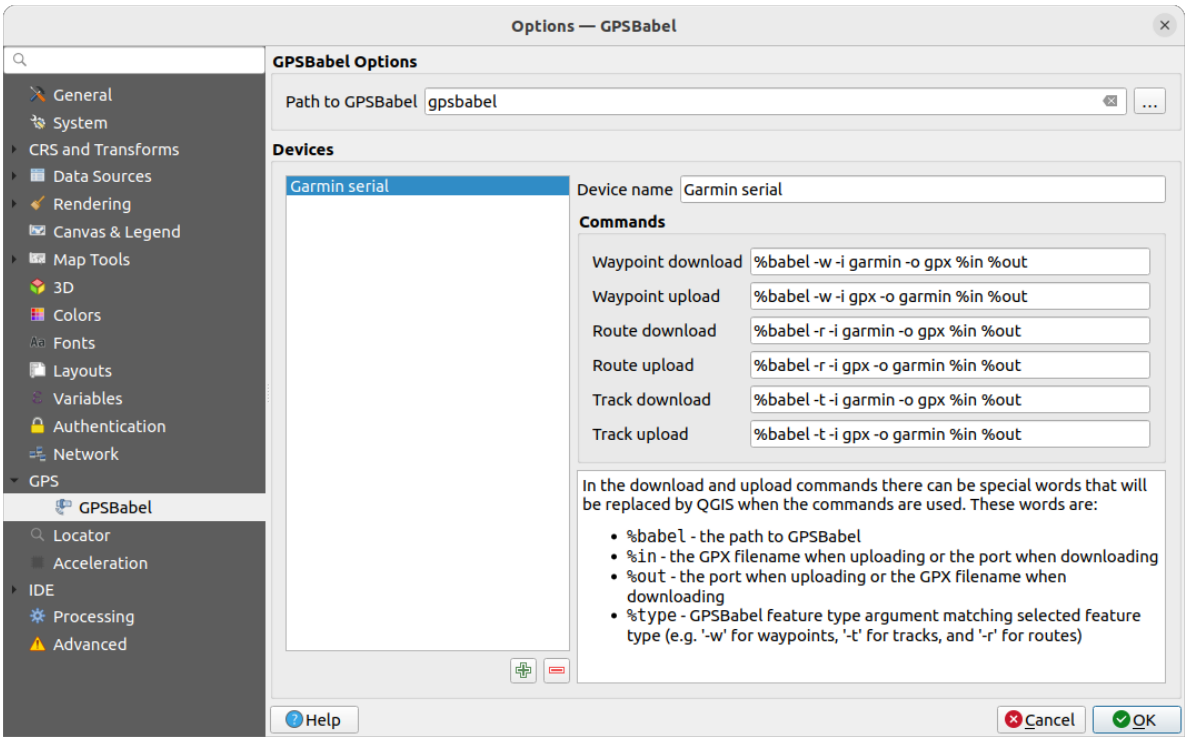


Figura 20.2: Ajustes de GPS Babel

1. Primero hay que definir la *Ruta a los binarios GPSBabel*.
2. A continuación, puede añadir su dispositivo. Puede actualizar la lista de dispositivos con el botón Añadir nuevo dispositivo o Eliminar dispositivo.
3. Para cada dispositivo:
   - indique una *Nombre del dispositivo*.
   - se configuran diferentes *Comandos* que QGIS utilizará al interactuar con él, tales como:
     - *Descarga de puntos de ruta* del dispositivo
     - *Subida de puntos de ruta* al dispositivo
     - *Descarga de Ruta* del dispositivo
     - *Subida de Ruta* al dispositivo
     - *Descarga de Traza* desde el dispositivo
     - *Subida de Traza* al dispositivo

     Aunque los comandos suelen ser los de GPSBabel, también puede utilizar cualquier otro programa de línea de comandos que pueda crear un archivo GPX. QGIS sustituirá las palabras claves `%type`,``%in``, y `%out` cuando ejecuta el comando.

     Por ejemplo, si crea un tipo de dispositivo con el comando de descarga `gpsbabel %type -i garmin -o gpx %in %out` y luego lo usa para descargar puntos de ruta desde el puerto `/dev/ttyS0` al archivo `output.gpx`, QGIS reemplazará las palabras clave y ejecutará el comando `gpsbabel -w -i garmin -o gpx /dev/ttyS0 output.gpx`.

     Lea el manual de GPSBabel para conocer las opciones de línea de comandos que pueden ser específicas para su caso de uso.

Una vez creado un nuevo tipo de dispositivo, aparecerá en las listas de dispositivos para los algoritmos de descarga y carga de GPS.

> **Nota**
>
> Las unidades GPS permiten almacenar datos en diferentes sistemas de coordenadas. Cuando descargue un archivo GPX (de su unidad GPS o de un sitio web) y luego lo cargue en QGIS, asegúrese de que los datos almacenados en el archivo GPX utilizan WGS 84 (latitud/longitud). QGIS lo espera, y es la especificación GPX oficial. Ver GPX 1.1 Schema Documentation.

20.2 Seguimiento de GPS en Vivo

QGIS puede ayudarle a realizar un seguimiento sobre el terreno con un receptor GPS. Esta operación de seguimiento en vivo se realiza utilizando la Barra de herramientas GPS 1. Alguna *configuración del dispositivo* puede ser necesitada antes de conectar QGIS y el receptor GPS.

20.2.1 Barra de herramientas GPS

La *Barra de herramientas GPS* proporciona las herramientas principales para controlar una sesión de seguimiento en directo. Puede activarse desde *Ver* ► *Barras de herramientas* ► *GPS*. Sigue el estado del proyecto, del GPS y del rastreo GPS actual, y activa acciones sólo cuando tienen sentido. Una vez que un dispositivo es detectado por QGIS, podrás interactuar con él:

-  Conectarse al GPS
-  Desconectar del GPS
-  Recentrar mapa en posición GPS: el mapa se recentrará inmediatamente en la posición GPS actual independientemente del parámetro *recentrado automático*.
-  Set destination layer for GPS digitized features: Predeterminado, QGIS *Seguirá la capa activa* para la digitalización de características, lo que significa que las herramientas de digitalización GPS se adaptarán a la capa seleccionada en el panel *Capas* y las características creadas se almacenarán en ella en consecuencia. Aunque esto puede ser útil en algunos contextos, también implica una interacción cuidadosa con otras capas para evitar almacenar inadvertidamente objetos espaciales en capas no deseadas. Esta opción le permite indicar explícitamente una capa para el almacenamiento de datos durante la sesión de rastreo en vivo, y podría cambiar dependiendo de sus necesidades. Las herramientas expuestas se adaptarán también al tipo de capa seleccionado.

  La capa de destino GPS se hace editable automáticamente cuando se está creando un objeto espacial, y se notifica al usuario en consecuencia.
- acciones para la creación de objetos espaciales:
  -  Añadir vértice a la traza GPS usando la localización GPS actual.
  -  Crear un nuevo objeto espacial de punto en la ubicación GPS actual.
  -  Crea un nuevo objeto espacial de línea utilizando la traza GPS actual.
  -  Crear un nuevo objeto espacial polígono usando la traza GPS actual.
  -  Restablecer traza
-  Mostrar información GPS: abre el panel *Información GPS*
- un cuadro de visualización para acceder rápidamente a algunos componentes de información del GPS que puedas necesitar:
  -  Mostrar ubicación

  - Mostrar altitud (Geoide)
  - Mostrar altitud (elipsoide WGS 84)
  - Mostrar velocidad de avance
  - Mostrar rumbo
  - Mostrar la longitud total de la traza
  - Mostrar distancia desde el inicio de la traza

- El botón Configuración abre un menú desplegable con configuraciones comunes que se espera que se modifiquen en mitad de la sesión:
  - *Mostrar marcador de localización*
  - *Mostrar línea de rumbo*
  - *Rotar el mapa para que coincida con la dirección GPS*
  - Opciones para controlar el recentrado del mapa:
    * *Siempre recentrar mapa* cuando la posición GPS se desplaza del centro del lienzo del mapa una cierta distancia (en relación a la extensión del lienzo del mapa)
    * *Recentrar mapa al salir de la extensión*
    * *Nunca recentrar*
  - *Añadir automáticamente vértices de seguimiento* cada vez que se reciban nuevas localizaciones desde el dispositivo GPS.
  - *Guardar automáticamente los objetos espaciales añadidos*: los objetos espaciales creados a partir de localizaciones GPS se asignan inmediatamente a sus capas de destino (omitiendo el búfer de edición de capas habitual)
  - *Destino de la marca de tiempo* para ajustar el campo de almacenamiento de la marca de tiempo
  - *Registrar en GeoPackage/SpatiaLite…*: Cuando se activa, se pedirá al usuario que seleccione un archivo GeoPackage/SpatiaLite existente o que introduzca un nuevo nombre de archivo. Se creará una tabla `gps_points` y `gps_tracks` en el archivo con una estructura predefinida.

    Todos los mensajes GPS entrantes se registrarán en la capa `gps_points`, junto con la información de velocidad, rumbo, altitud y precisión del GPS.

    Cuando se desconecta el GPS (o se cierra QGIS), toda la pista GPS registrada se añadirá a la tabla `gps_tracks` (junto con alguna información calculada como la longitud de la pista y las horas de inicio y fin).
  - *Registrar sentencias NMEA…*: activa el registro de todas las cadenas NMEA sin procesar desde el dispositivo a un archivo de texto.
  - *Configuración GPS…* para acceder al diálogo GPS *opciones globales*

> **Truco**
>
> **Información en tiempo real sobre la posición en la barra de estado**
>
> Cuando un dispositivo GPS está conectado y el usuario mueve el cursor sobre el lienzo del mapa, un mensaje de la barra de estado en vivo muestra la distancia y el rumbo desde el cursor hasta la posición GPS. En esta pantalla se respetan los ajustes de distancia y demora del proyecto. En un dispositivo de pantalla táctil, utilice un evento de pulsar y mantener para activar el mensaje de la barra de estado en directo.

20.2.2 Panel de información GPS

Para una monitorización completa de un seguimiento GPS en directo en QGIS, puede que necesite habilitar el *Panel de Información GPS* (*Ver* ► *Paneles* ► o pulse `Ctrl+0`).

En la esquina superior derecha del *Panel de información GPS*, pulse *Conectar* para iniciar la conexión entre QGIS y un receptor GPS conectado, o *Desconectar*.

En la parte superior izquierda del panel, se puede acceder a los siguientes botones:

- Posición: detalles en directo sobre la posición GPS y los sensores
- Signal: intensidad de la señal de las conexiones por satélite
- |opciones| Configuración: menú desplegable para *opciones de seguimiento en directo* que puede necesitar modificar durante una sesión

Posición y atributos adicionales

En la pestaña Posición, si el GPS está recibiendo señales de satélites, verás tu posición en latitud, longitud y altitud junto con atributos adicionales.

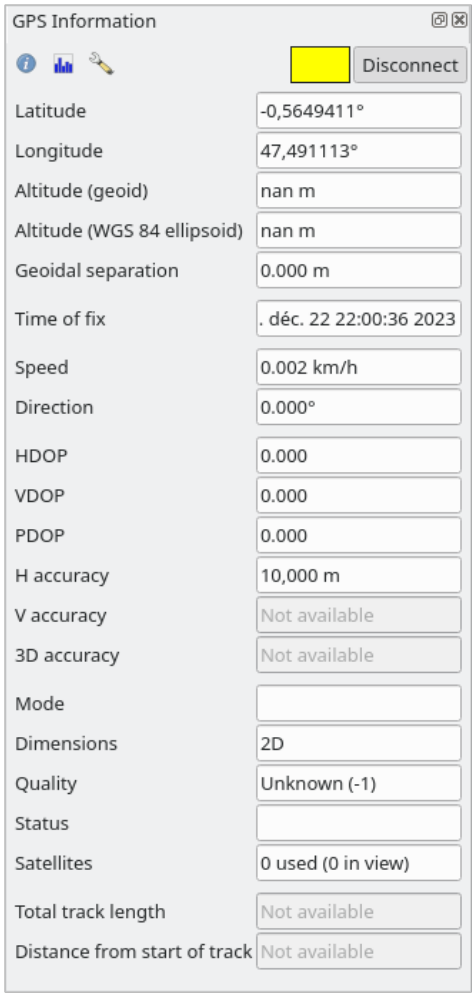


Figura 20.3: Posición de rastreo GPS y atributos adicionales

- *Latitud*

- *Longitud*
- *Altitud (Geoide)*: Altitud/elevación por encima o por debajo del nivel medio del mar
- *Altitud (elipsoide WGS 84)*: Altitud/elevación por encima o por debajo del elipsoide terrestre WGS-84.
- *Separación geoidal*: Diferencia entre el elipsoide terrestre WGS-84 y el nivel medio del mar (geoide), − significa que el nivel medio del mar está por debajo del elipsoide
- *Hora del arreglo*
- *Velocidad*: Velocidad del terreno
- *Dirección*: Rumbo medido en grados en el sentido de las agujas del reloj desde el norte verdadero hacia la dirección de desplazamiento.
- *HDOP*: Dilución horizontal de la precisión
- *VDOP*: Dilución vertical de la precisión
- *PDOP*: Dilución de la precisión
- *Exactitud H*: Precisión horizontal en metros
- *Exactitud V*: Precisión vertical en metros
- *Exactitud 3D*: Media cuadrática 3D (RMS) en metros
- *Modo*: Modo 2D/3D de configuración del receptor GPS, puede ser `automático` o `manual`.
- *Dimensiones*: Dimensión fija de la posición, puede ser `2D`, `3D` o `Sin fijar`.
- *Calidad*: Indicador de calidad de posicionamiento
- *Estado*: Estado de fijación de la posición, puede ser `Válido` o `No válido`.
- *Satélites*: Número de satélites utilizados para obtener la posición
- *Longitud total de la traza*: Distancia total de la traza GPS actual
- *Distancia desde el inicio de la pista*: Distancia directa desde el primer vértice de la traza GPS actual hasta el último vértice.

Señal

Con la pestaña  Señal, puede ver la intensidad de la señal de los satélites de los que está recibiendo señales.

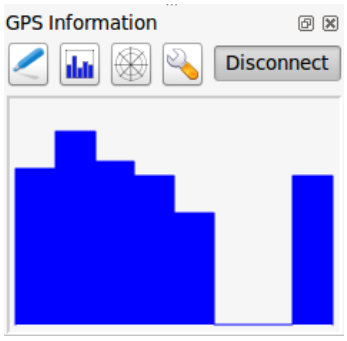


Figura 20.4: Intensidad de la señal de rastreo GPS

20.2.3 Conéctese a un GPS Bluetooth para seguimiento en vivo

Con QGIS puede conectar un GPS Bluetooth para la recopilación de datos de campo. Para realizar esta tarea, necesita un dispositivo GPS Bluetooth y un receptor Bluetooth en su computadora.

Al principio, debe permitir que su dispositivo GPS sea reconocido y emparejado con la computadora. Encienda el GPS, vaya al icono de Bluetooth en su área de notificación y busque un Dispositivo nuevo.

En el lado derecho de la máscara de selección de dispositivos, asegúrese de que todos los dispositivos estén seleccionados para que su unidad GPS probablemente aparezca entre los disponibles. En el siguiente paso debería estar disponible un servicio de conexión en serie, selecciónelo y haga clic en el botón *Configurar*.

Recuerde el número del puerto COM asignado a la conexión GPS como resultado de las propiedades de Bluetooth.

Una vez que se haya reconocido el GPS, realice el emparejamiento para la conexión. Normalmente, el código de autorización es `0000`.

Ahora abra el panel *Información GPS* y cambie a Pantalla de opciones de GPS. Seleccione el puerto COM asignado a la conexión GPS y haga clic en *Conectar*. Después de un rato, debería aparecer un cursor que indica su posición.

Si QGIS no puede recibir datos de GPS, entonces debe reiniciar su dispositivo GPS, esperar de 5-10 segundos y luego intentar conectarse nuevamente. Por lo general, esta solución funciona. Si vuelve a recibir un error de conexión, asegúrese de que no tiene otro receptor Bluetooth cerca de usted, emparejado con la misma unidad GPS.

20.2.4 Ejemplos de conexión de dispositivos GPS en QGIS

Utilizando GPSMAP 60cs

MS Windows

La forma más fácil de hacerlo trabajar es usar un software intermedio (software gratuito, no abierto) llamado`GPSGate <https://gpsgate.com/gpsgate-splitter>`_.

Inicia el programa, haz que busque dispositivos GPS (funciona tanto para los USB como para los BT) y luego en QGIS:

1. En *Configuración* ► *Opciones* ► *GPS*, detecte el dispositivo conectado. Puede utilizar el modo *Autodetectar*.
2. Pulse *Conectar* en el panel *Información GPS*

Ubuntu/Mint GNU/Linux

Para Windows la formas más fácil es usar un servidor en medio, en este caso GPSD, de manera

1. Instalar el programa

```
sudo apt install gpsd
```

2. Luego cargue el módulo del kernel `garmin_gps`

```
sudo modprobe garmin_gps
```

3. Y luego conecta la unidad y comprueba con `dmesg` el dispositivo real que está utilizando la unidad, por ejemplo `/dev/ttyUSB0`.
4. Ahora puedes lanzar gpsd

```
gpsd /dev/ttyUSB0
```

5. Y finalmente conectar con la herramienta de rastreo en vivo QGIS.

Usando el registrador de datos BTGP-38KM (solo Bluetooth)

Usar GPSD (bajo Linux) o GPSGate (bajo Windows) es sencillo.

Usando el registrador de datos BlueMax GPS-4044 (tanto BT como USB)

MS Windows

El seguimiento en directo funciona tanto para USB como para BT, usando GPSGate o incluso sin él, sólo tienes que usar el modo  *Autodetectar*, o apuntar la herramienta al puerto correcto.

Ubuntu/Mint GNU/Linux

Para USB

El seguimiento en vivo funciona tanto con GPSD

```
gpsd /dev/ttyACM3
```

o sin él, conectando la herramienta de seguimiento en vivo de QGIS directamente al dispositivo (por ejemplo `/dev/ttyACM3`).

Para Bluetooth

El seguimiento en vivo funciona tanto con GPSD

```
gpsd /dev/rfcomm0
```

o sin él, conectando la herramienta de seguimiento en vivo de QGIS directamente al dispositivo (por ejemplo `/dev/rfcomm0`).

CAPÍTULO 21

Sistema de autenticación

21.1 Vista general del sistema de Autenticación

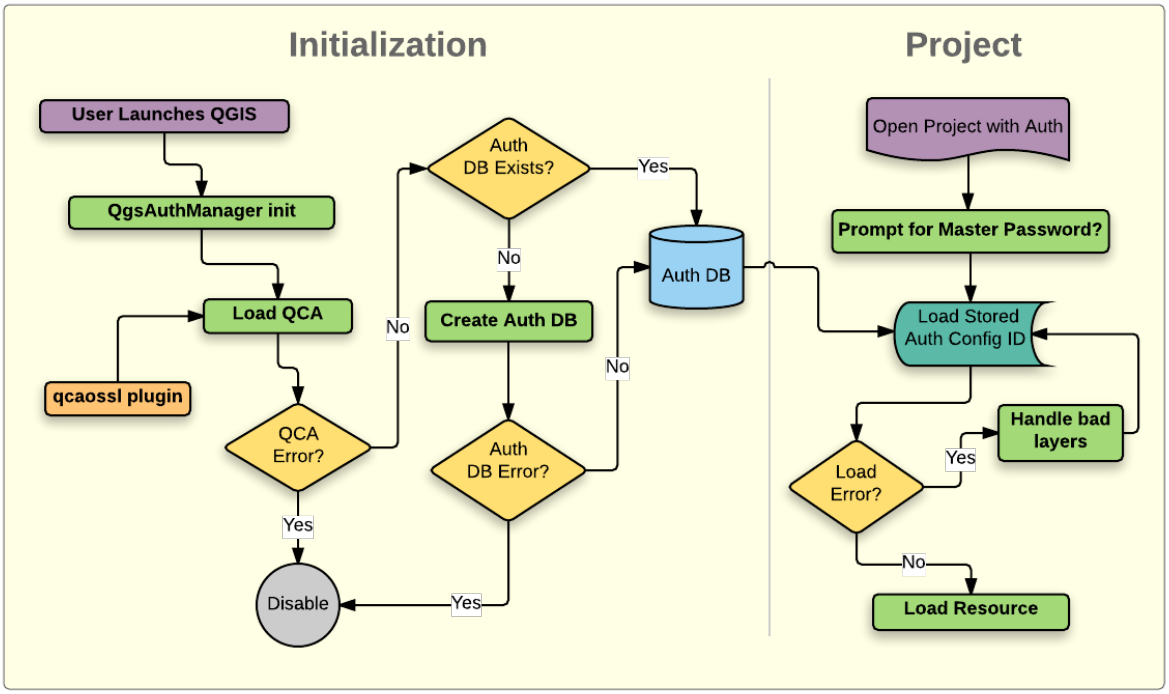


Figura 21.1: Anatomía del sistema de Autenticación

21.1.1 Base de datos de autenticación

El sistema de autenticación almacena las configuraciones de autenticación en un archivo de base de datos SQLite ubicado, por defecto, en `<profile directory>/qgis-auth.db`.

Esta base de datos de autenticación se puede mover entre las instalaciones de QGIS sin afectar otras preferencias de usuario actuales de QGIS, ya que está completamente separada de la configuración normal de QGIS. Se genera un ID de configuración (una cadena alfanumérica aleatoria de 7 caracteres) cuando se almacena inicialmente una configuración en la base de datos. Esto representa la configuración, lo que permite que la ID se almacene en componentes de la aplicación de texto sin formato (como archivos de proyecto, complemento o configuración) sin revelar sus credenciales asociadas.

Nota

El directorio principal de *qgis-auth.db* se puede configurar usando la siguiente variable de entorno, `QGIS_AUTH_DB_DIR_PATH`, o se puede configurar en la línea de comando durante el lanzamiento con la opción `--authdbdirectory`.

21.1.2 Bases de datos de autenticación personalizadas

QGIS se puede configurar para utilizar una base de datos de autenticación personalizada en lugar de la base de datos SQLite predeterminada mencionada anteriormente: se puede utilizar cualquier base de datos compatible con el módulo Qt SQL (por ejemplo, PostgreSQL, MySQL, etc.), siempre que el controlador Qt SQL correspondiente esté disponible en el sistema.

Esto puede resultar útil en situaciones en las que un usuario desea compartir la misma base de datos de autenticación entre varias instalaciones de QGIS, o cuando un usuario desea utilizar una base de datos de autenticación diferente a la base de datos SQLite predeterminada, o cuando el servidor QGIS utiliza una base de datos de autenticación centralizada.

La única forma de configurar una base de datos de autenticación personalizada es estableciendo la variable de entorno `QGIS_AUTH_DB_URI` con la URI de la conexión, que tiene el formato `driver://username:password@hostname:port/database?options`.

Dónde:

- `driver` es el nombre del controlador SQL de Qt que se va a utilizar, por ejemplo, `QPSQL` para PostgreSQL, `QMYSQL` para MySQL, etc.
- `username` es el nombre de usuario que se utilizará para conectarse a la base de datos.
- `password` es la contraseña que se debe utilizar para conectarse a la base de datos.
- `hostname` es el nombre de host del servidor de la base de datos
- `port` es el puerto del servidor de la base de datos
- `database` es el nombre de la base de datos que se va a utilizar.
- `options` Son las opciones que se pasan al controlador, por ejemplo, `sslmode=require` para PostgreSQL.

Nota

El `schema` se puede especificar en las opciones URI, por ejemplo, `QPSQL://nombre_usuario:contraseña@nombre_host:puerto/base_datos?esquema=nombre_esquema`

La base de datos debe existir antes de iniciar QGIS, y el usuario debe tener los permisos necesarios para conectarse a la base de datos y crear las tablas requeridas si no existen.

Advertencia

La contraseña en la URI se almacena en texto sin formato en la variable de entorno, por lo que se recomienda utilizar un usuario sin contraseña o un usuario con permisos limitados para conectarse a la base de datos.

Advertencia

Cualquier base de datos que no esté basada en SQLite se considera de solo lectura (esto se puede cambiar mediante complementos de Python si es necesario).

Esta es una función avanzada, diseñada para permitir que QGIS utilice una o más bases de datos de autenticación personalizadas o incluso implementaciones personalizadas de Python para el almacenamiento de credenciales.

El sistema también está diseñado para permitir el uso de múltiples bases de datos de autenticación, pero actualmente no existe una interfaz de usuario para gestionar múltiples almacenes de credenciales, por lo que su uso requiere una configuración y gestión manuales, normalmente desde un complemento de Python.

21.1.3 Contraseña maestra

Para almacenar o acceder a información confidencial dentro de la base de datos, un usuario debe definir una "contraseña maestra". Se solicita y verifica una nueva contraseña maestra cuando se almacena inicialmente cualquier dato cifrado en la base de datos. Cuando se accede a información confidencial, se solicita al usuario la contraseña maestra. Luego, la contraseña se almacena en caché durante el resto de la sesión (hasta que se cierra la aplicación), a menos que el usuario elija manualmente una acción para borrar su valor en caché. Algunas instancias de uso del sistema de autenticación no requieren la entrada de la contraseña maestra, como cuando se selecciona una configuración de autenticación existente o cuando se aplica una configuración a una configuración de servidor (como cuando se agrega una capa WMS).

Puede optar por guardar la contraseña en el `Wallet/Keyring` de su computadora.

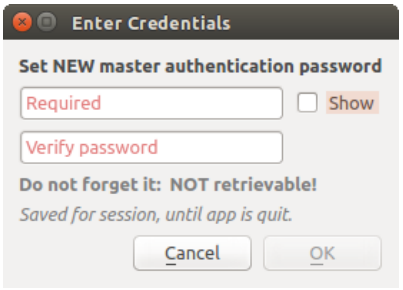


Figura 21.2: Ingrese la nueva contraseña maestra

Nota

Se puede establecer una ruta a un archivo que contiene la contraseña maestra mediante la siguiente variable de entorno, `QGIS_AUTH_PASSWORD_FILE`.

Administrar la contraseña maestra

Una vez configurada, la contraseña maestra se puede restablecer; se necesitará la contraseña maestra actual antes de restablecer. Durante este proceso, existe la opción de generar una copia de seguridad completa de la base de datos actual.



Figura 21.3: Restablecer contraseña maestra

Si el usuario olvida la contraseña maestra, no hay forma de recuperarla o anularla. Tampoco hay forma de recuperar información cifrada sin conocer la contraseña maestra.

Si un usuario ingresa su contraseña existente de manera incorrecta tres veces, el diálogo ofrecerá borrar la base de datos.

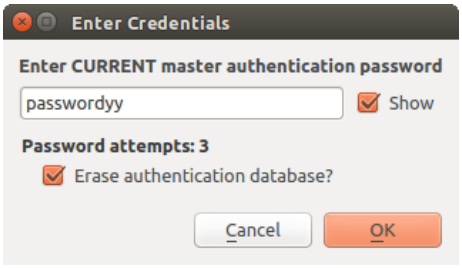


Figura 21.4: Solicitud de contraseña después de tres intentos no válidos

21.1.4 Configuraciones de autenticación

Puede administrar las configuraciones de autenticación desde *Configuraciones* en la pestaña *Autenticación* del cuadro de diálogo Opciones de QGIS (:menuselection:` Configuración -> Opciones`).

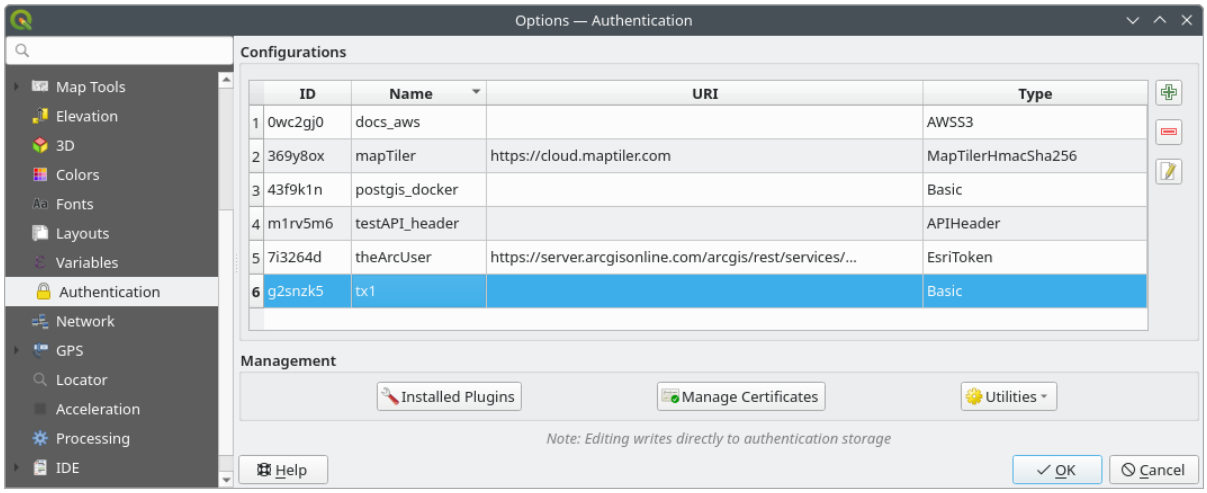


Figura 21.5: Editor de configuraciones

To create a new authentication configuration:

1. Press the Add a new authentication configuration
2. Provide a *Name* and optional *Resource* URL
3. A seven-characters *ID* will be assigned to the configuration for identification purpose. It can be customized after you pressed the Unlock to edit ID button. Be aware that updating that ID may break things.
4. Select the *target method* and fill the corresponding connection details
5. Once done, click *Save* to save the configuration. The saved configuration appears in the list of authentication configurations and can be used with supported connections.

The saved configuration can later be edited using Edit selected configuration or removed using Delete selected configuration.

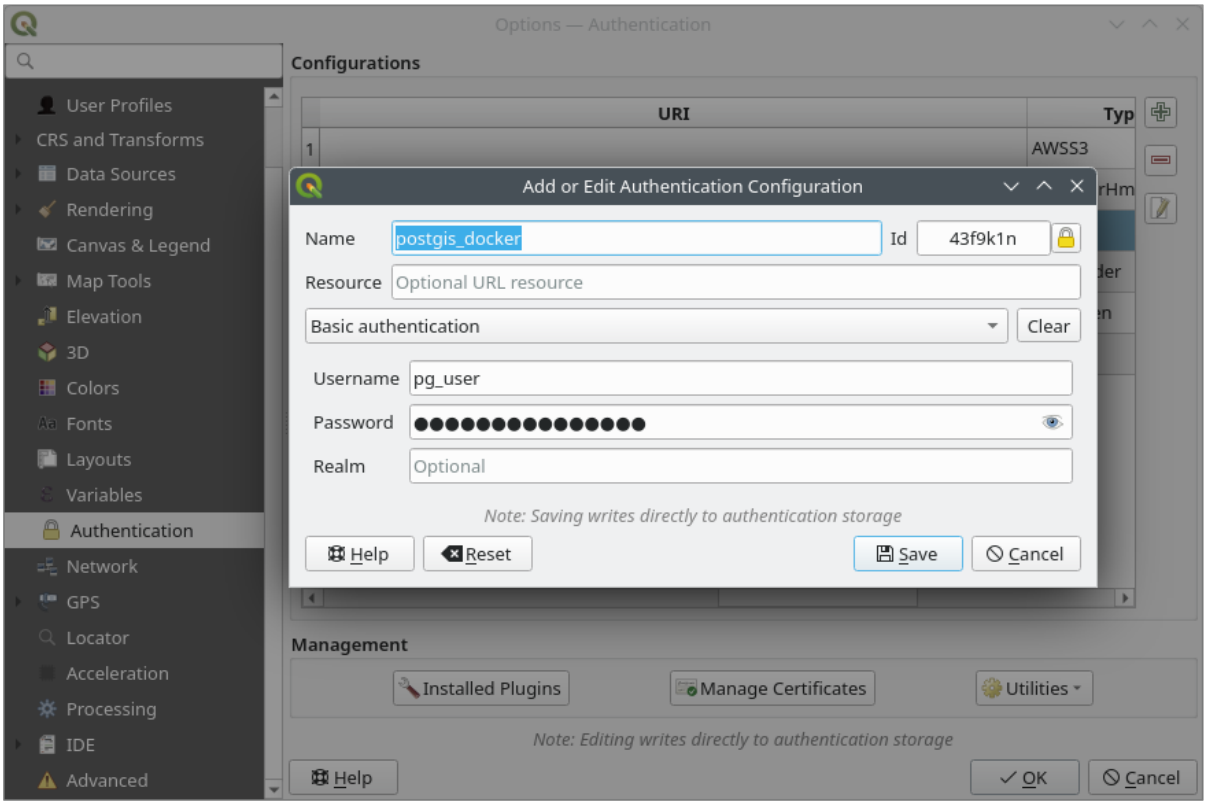


Figura 21.6: Agregar configuración desde el editor de configuración

The same type of operations for authentication configuration management (Add, Edit and Remove) can be done when configuring a given service connection, such as configuring an OWS service connection. For that, there are action buttons within the *configuration selector* for fully managing configurations found within the authentication database. In this case, there is no need to go to the *configurations* in *Authentication* tab of QGIS options unless you need to do more comprehensive configuration management.

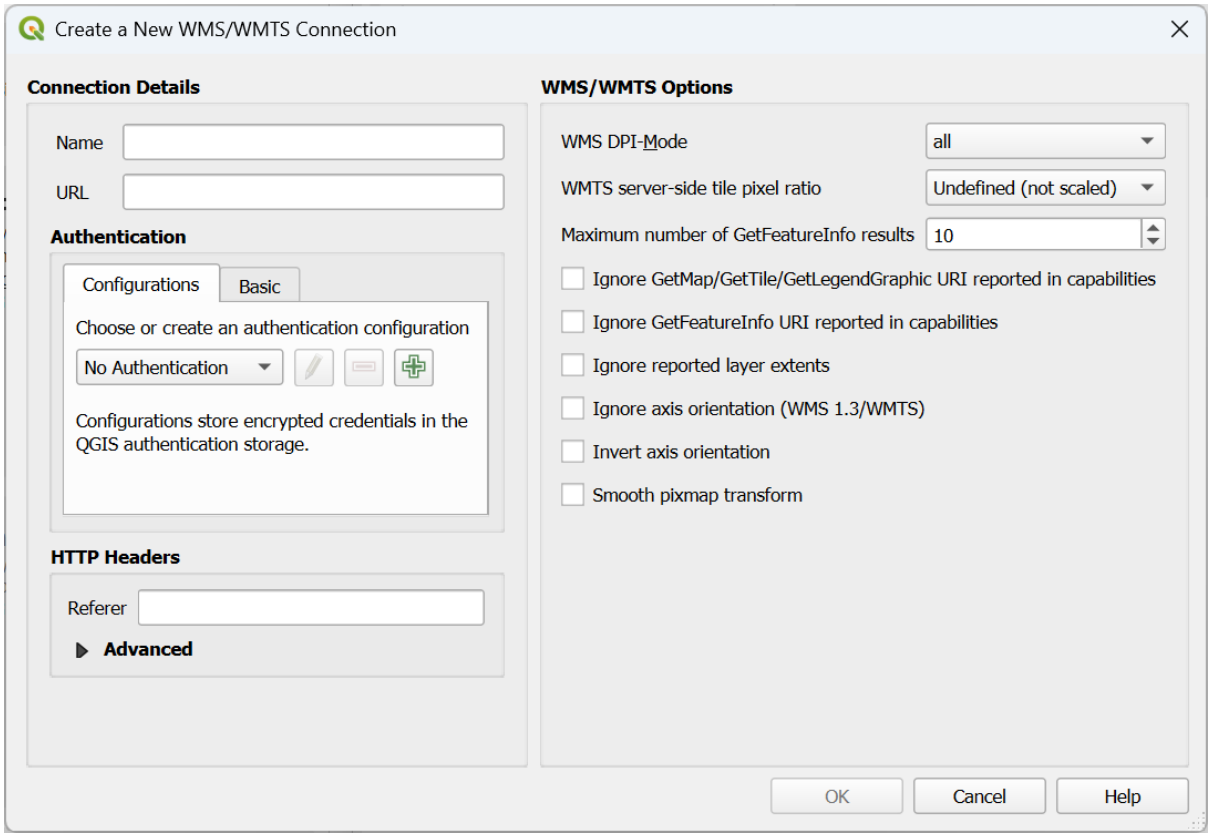


Figura 21.7: WMS connection dialog showing authentication configuration widget

Al crear o editar una configuración de autenticación, la información requerida es un nombre, un método de autenticación y cualquier otra información que requiera el método de autenticación (vea más sobre los tipos de autenticación disponibles en *Métodos de autenticación*).

21.1.5 Métodos de autenticación

Los complementos de C ++ proporcionan las autenticaciones disponibles de la misma manera que los complementos de proveedores de datos son compatibles con QGIS. El método de autenticación que se puede seleccionar es relativo al acceso necesario para el recurso/proveedor, p. ejemplo HTTP(S) o base de datos, y si hay soporte tanto en el código QGIS como en un complemento. Como tal, algunos complementos de métodos de autenticación pueden no ser aplicables en todos los lugares donde se muestra un selector de configuración de autenticación. Se puede acceder a una lista de complementos de métodos de autenticación disponibles y sus recursos / proveedores compatibles yendo a *Configuración -> Opciones* y, en la pestaña :guilabel:‘Autenticación`, haga clic en el botón *Complementos instalados*.

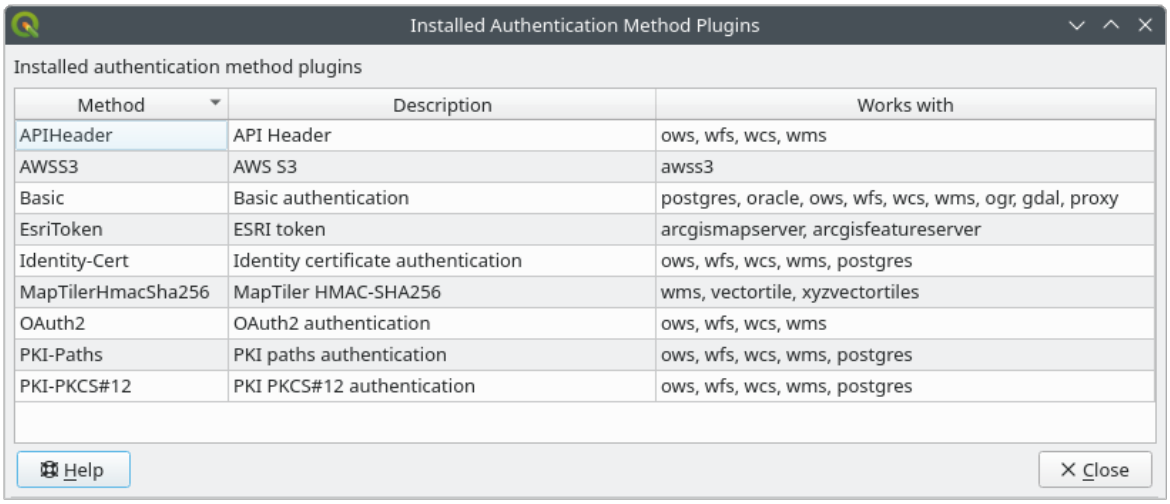


Figura 21.8: Lista de complementos de métodos disponibles

Se pueden crear complementos para nuevos métodos de autenticación que no requieren la recompilación de QGIS. Dado que actualmente el soporte para complementos es solo C ++, QGIS deberá reiniciarse para que el nuevo complemento incorporado esté disponible para el usuario. Asegúrese de que su complemento esté compilado con la misma versión de destino de QGIS si tiene la intención de agregarlo a una instalación de destino existente.

API Header Authentication

The API Header authentication method allows you to connect to services that require custom HTTP headers, such as API keys. For example, if a WMS service requires an API key sent as a header (e.g. `X-API-KEY`), you can use this method to provide it.

You can add one or more headers with the button; provide a *Header Name* and *Header Value* for each entry. Use to delete the selected header or the *Clear* button to remove all entries.

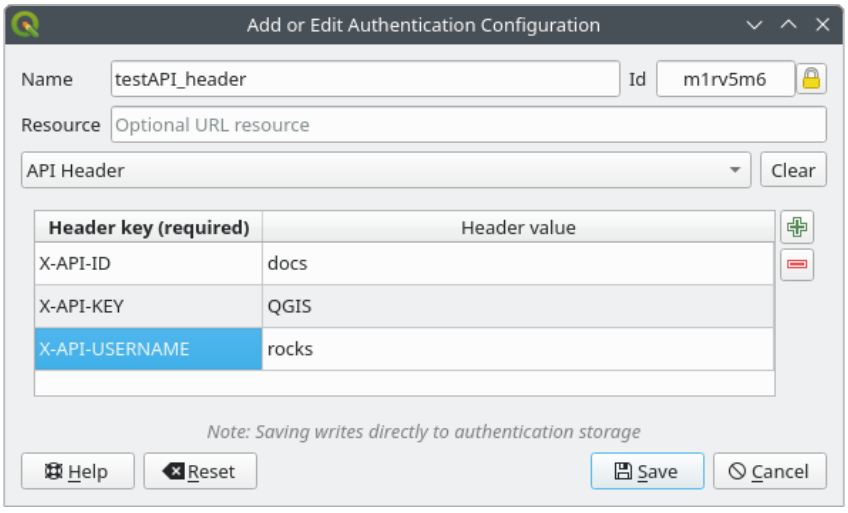


Figura 21.9: API Header authentication configs

AWS S3 Authentication

The AWS S3 authentication method allows you to connect to Amazon S3 resources by providing a *Name*, *Resource*, *Username*, *Password*, and *Region*.

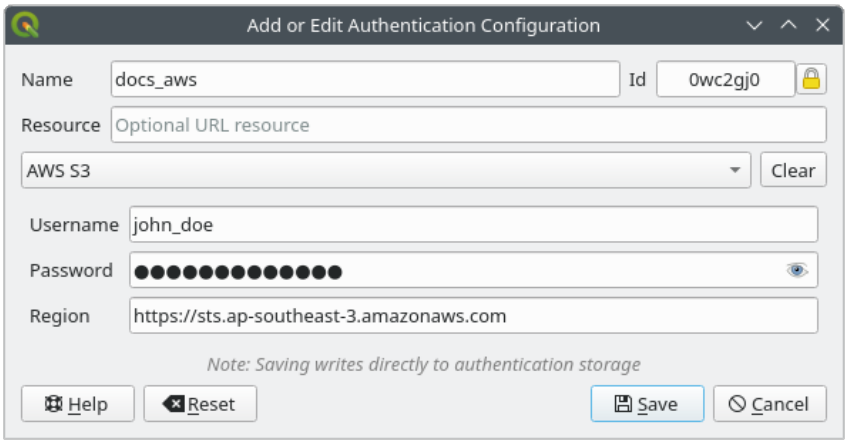


Figura 21.10: AWS S3 authentication configs

Basic Authentication

The Basic authentication method is used for services that require standard HTTP authentication with a *Username*, *Password* and *Realm*.

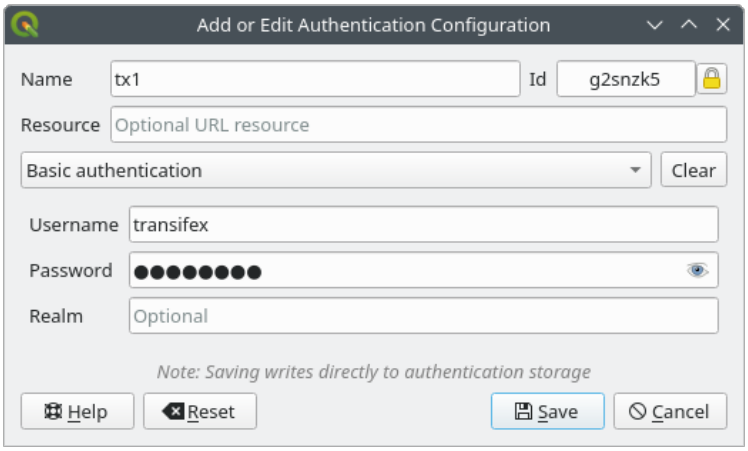


Figura 21.11: Configuraciones básicas de autenticación HTTP

ESRI Token Authentication

The ESRI token authentication method is used for *ArcGIS REST Servers* that require token-based authentication.

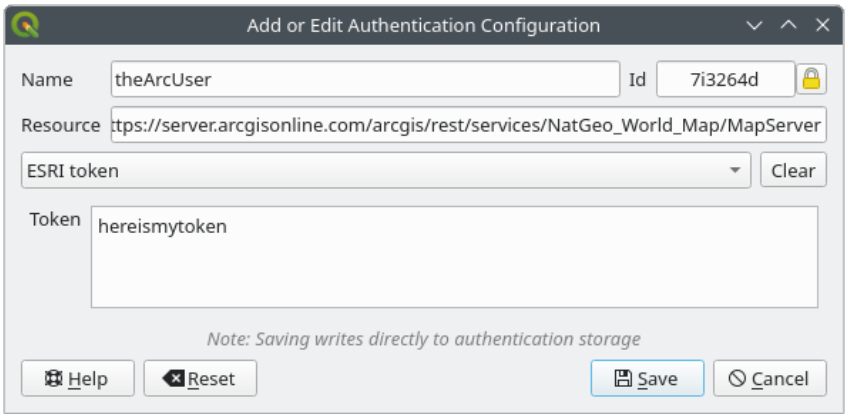


Figura 21.12: Configuraciones de autenticación de token ESRI

Identity certificate authentication

The Identity certificate authentication method allows you to connect using a client *identity certificate*.

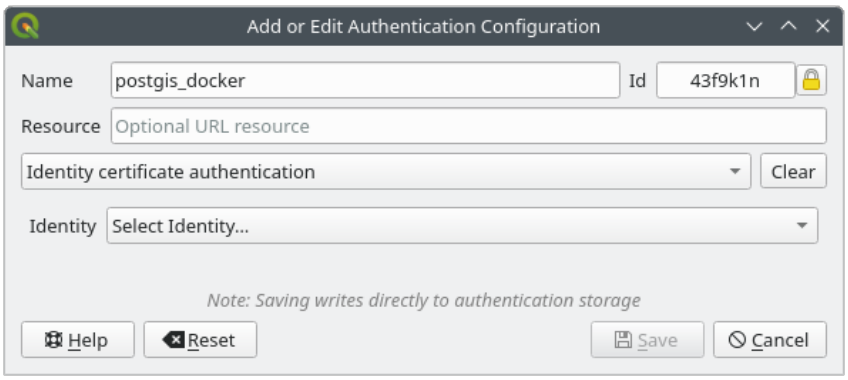


Figura 21.13: Identity-cert authentication configs

MapTiler HMAC-SHA256 Authentication

The MapTiler HMAC-SHA256 authentication method is used to connect to MapTiler services that require HMAC-SHA256 authorization.

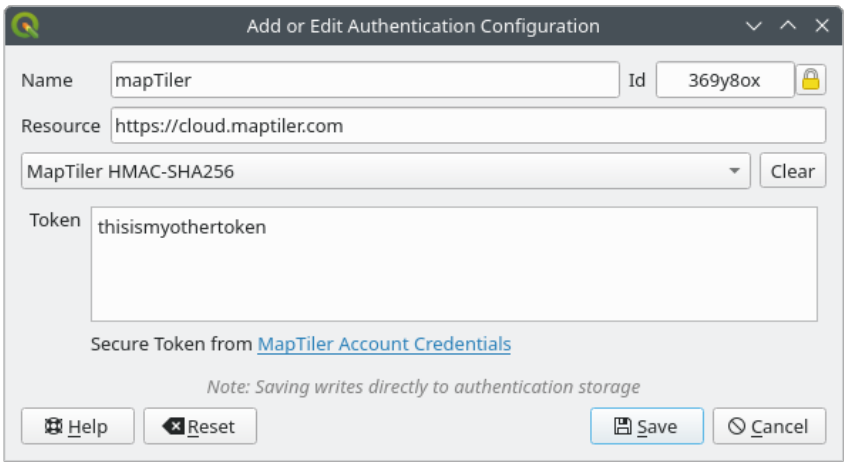


Figura 21.14: MapTiler HmacSha256 authentication configs

OAuth2 Authentication

The OAuth2 authentication method is used to connect to services that require OAuth2 2.0 authorization, allowing secure access using client credential and token-based authentication.

It supports using `localhost` as the redirect host, providing compatibility with services that do not accept `http://127.0.0.1` redirects (for example, Microsoft SharePoint).

An advanced OAuth2 configuration option, *Extra token header(s)*, allows attaching additional tokens returned by the OAuth2 token endpoint as HTTP(S) request headers.

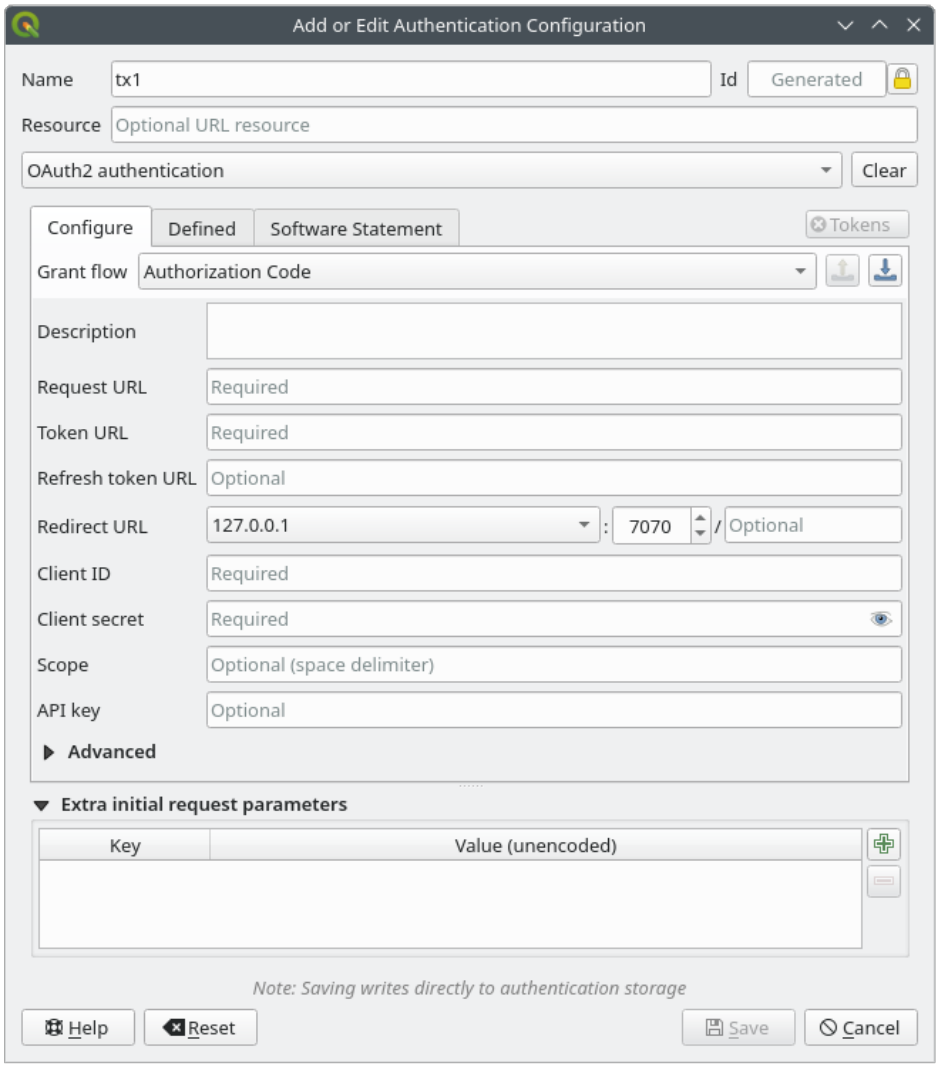


Figura 21.15: Configuraciones de autenticación OAuth2

PKI-Paths Authentication

The PKI-Paths authentication method allows you to connect using separate certificate and key files stored on your system.

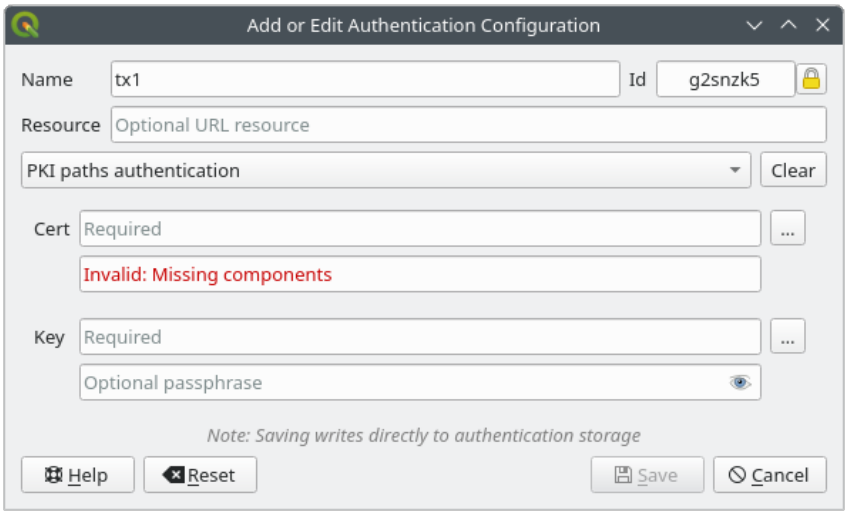


Figura 21.16: Configuraciones de autenticación de rutas PKI

PKI-PKCS#12 Authentication

The PKI-PKCS#12 authentication method allows you to connect using a single bundle file containing both the certificate and private key, with an optional passphrase for the key.

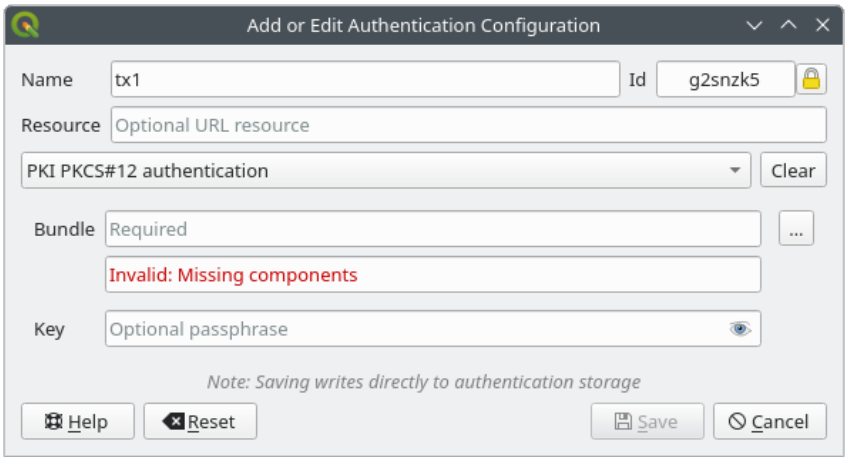


Figura 21.17: Configuraciones de autenticación de rutas de archivo PKI PKCS#12

21.1.6 Utilidades de contraseña maestra y configuración de autenticación

En el menú Opciones (*Configuración -> Opciones*) en la pestaña :guilabel:‘Autenticación`, hay varias acciones de utilidad para administrar la base de datos de autenticación y las configuraciones:

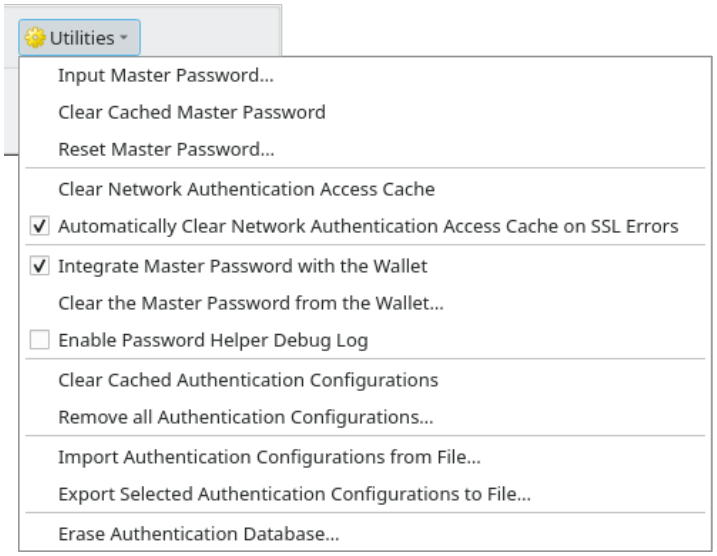


Figura 21.18: Menú utilidades

- *Input master password…*: opens the master password input dialog, independent of performing any authentication database command
- *Clear cached master password*: unsets the master password if it has been set
- *Reset master password…*: opens a dialog to change the master password (the current password must be known) and optionally back up the current database
- *Clear network authentication access cache*: clears the authentication cache of all connections
- *Automatically clear network authentication access cache on SSL errors*: the connection cache stores all authentication data for connections, also when the connection fails. If you change authentication configurations or certification authorities, you should clear the authentication cache or restart QGIS. When this option is checked, the authentication cache will be automatically cleared every time an SSL error occurs and you choose to abort the connection
- *Integrate master password with the Wallet*: adds the master password to your personal Wallet/Keyring
- *Clear the master password from the Wallet…*: deletes the master password from your Wallet/Keyring
- *Enable password helper debug log*: enables a debug tool that will contain all the log information of the authentication methods
- *Clear cached authentication configurations*: clears the internal lookup cache for configurations, used to speed up network connections. This does not clear QGIS's core network access manager's cache, which requires a relaunch of QGIS.
- *Remove all authentication configurations…*: clears the database of all configuration records, without removing other stored records.
- *Import authentication configurations from file…*: imports from an `.XML` file details for creating custom authentication configurations.
- *Export selected authentication configurations to file…*: exports the selected items to a possibly encrypted `.XML` file.

- *Erase authentication database…*: schedules a backup of the current database and complete rebuild of the database table structure. The actions are scheduled for a later time, to ensure that other operations, like project loading, do not interrupt the operation or cause errors due to a temporarily missing database.

21.1.7 Usando configuraciones de autenticación

Normalmente, se selecciona una configuración de autenticación en un cuadro de diálogo de configuración para servicios de red (como WMS). Sin embargo, el widget selector se puede incrustar en cualquier lugar donde se necesite autenticación o en funciones no básicas, como en complementos PyQGIS o C ++ de terceros.

When using the selector, *No authentication* is displayed in the pop-up menu control when nothing is selected, when there are no configurations to choose from, or when a previously assigned configuration can no longer be found in the database. Use the drop-down menu to select an existing authentication configuration or press 
Create a new autentication configuration to create a configuration you could use. More details at *Autenticación*.

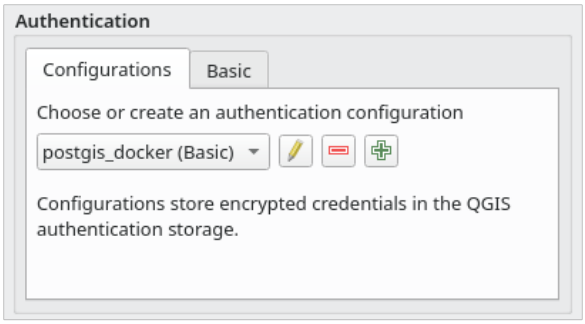


Figura 21.19: Selector de configuración de autenticación con configuración seleccionada

21.1.8 Python bindings

All classes and public functions have sip bindings, except QgsAuthCrypto, since management of the master password hashing and auth database encryption should be handled by the main app, and not via Python.

Una vez introducida la contraseña maestra, la API está abierta para acceder a las configuraciones de autenticación en la base de datos de autenticación, de forma similar a como funciona Firefox. Sin embargo, no se ha definido ninguna barrera contra el acceso de PyQGIS. Esto puede dar lugar a problemas cuando un usuario descarga/instala un complemento malicioso de PyQGIS o una aplicación independiente que obtiene acceso a las credenciales de autenticación.

21.2 Flujos de trabajo de autenticación de usuarios

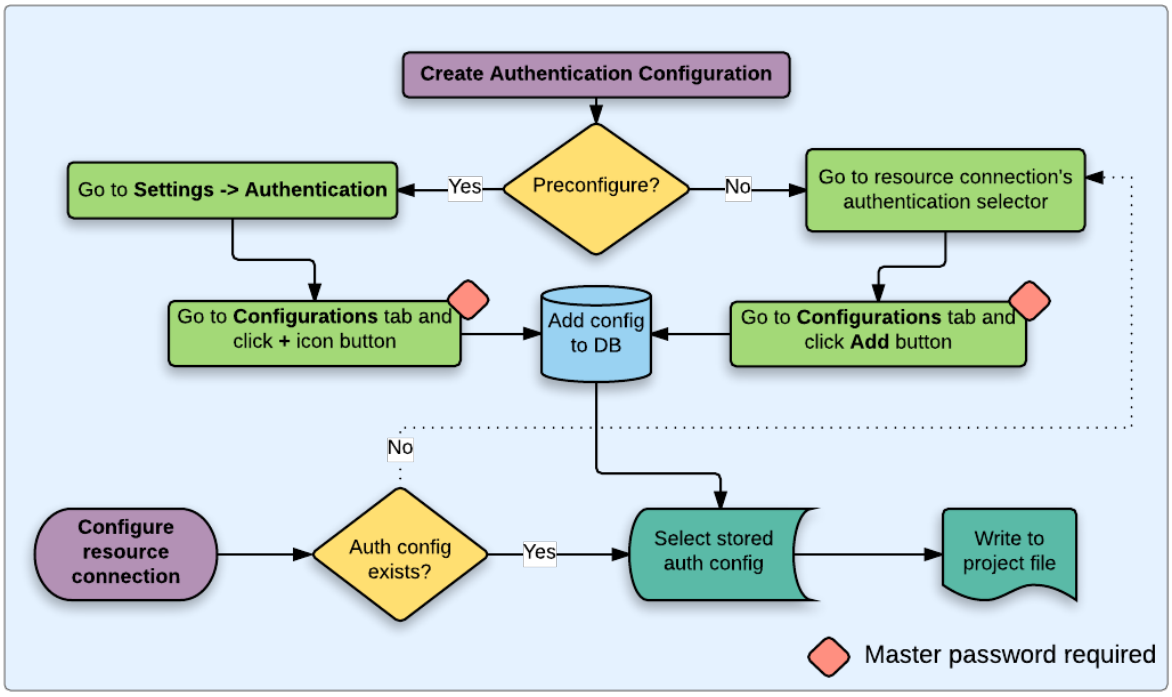


Figura 21.20: Flujo de trabajo de usuario genérico

21.2.1 Autenticación HTTP(S)

Una de las conexiones de recursos más comunes es a través de HTTP(S), p. Ej. Los servidores de mapas web y los complementos de métodos de autenticación a menudo funcionan para este tipo de conexiones. Los complementos de método tienen acceso al objeto de solicitud HTTP y pueden manipular tanto la solicitud como sus encabezados. Esto permite muchas formas de autenticación basada en Internet. Cuando se conecte a través de HTTP(S) utilizando el método de autenticación de nombre de usuario / contraseña estándar, se intentará la autenticación HTTP BASIC al realizar la conexión.

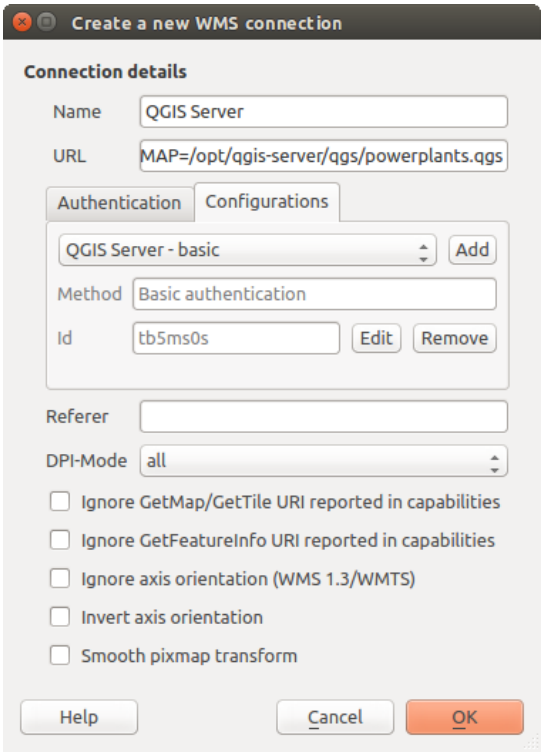


Figura 21.21: Configuración de una conexión WMS para HTTP BASIC

21.2.2 Base de datos de autenticación

Las conexiones a los recursos de la base de datos son almacenados generalmente como pares `key=value`, los cuales muestran nombres de usuario y (opcionalmente) contraseñas, si *no* se usa una configuración de autenticación. Al configurar con el sistema auth, la `key=value` será una representación abstracta de las credenciales, p.ej. `authfg=81t21b9`.

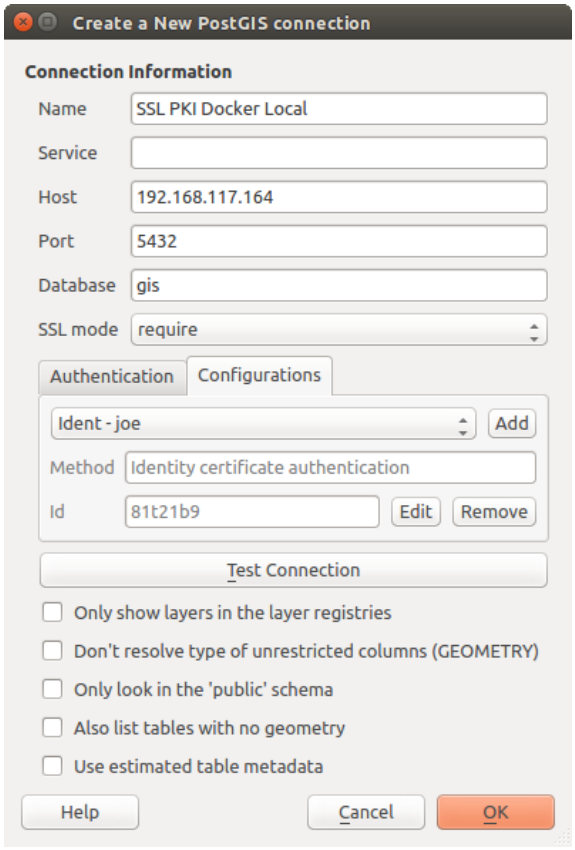


Figura 21.22: Configuración de una conexión Postgres SSL-con-PKI

21.2.3 Autenticación PKI

Al configurar componentes PKI dentro del sistema de autenticación, tiene la opción de importar componentes a la base de datos o hacer referencia a archivos de componentes almacenados en su sistema de archivos. Esto último puede ser útil si dichos componentes cambian con frecuencia o si los componentes serán reemplazados por un administrador del sistema. En cualquier caso, deberá almacenar cualquier frase de contraseña necesaria para acceder a las claves privadas dentro de la base de datos.

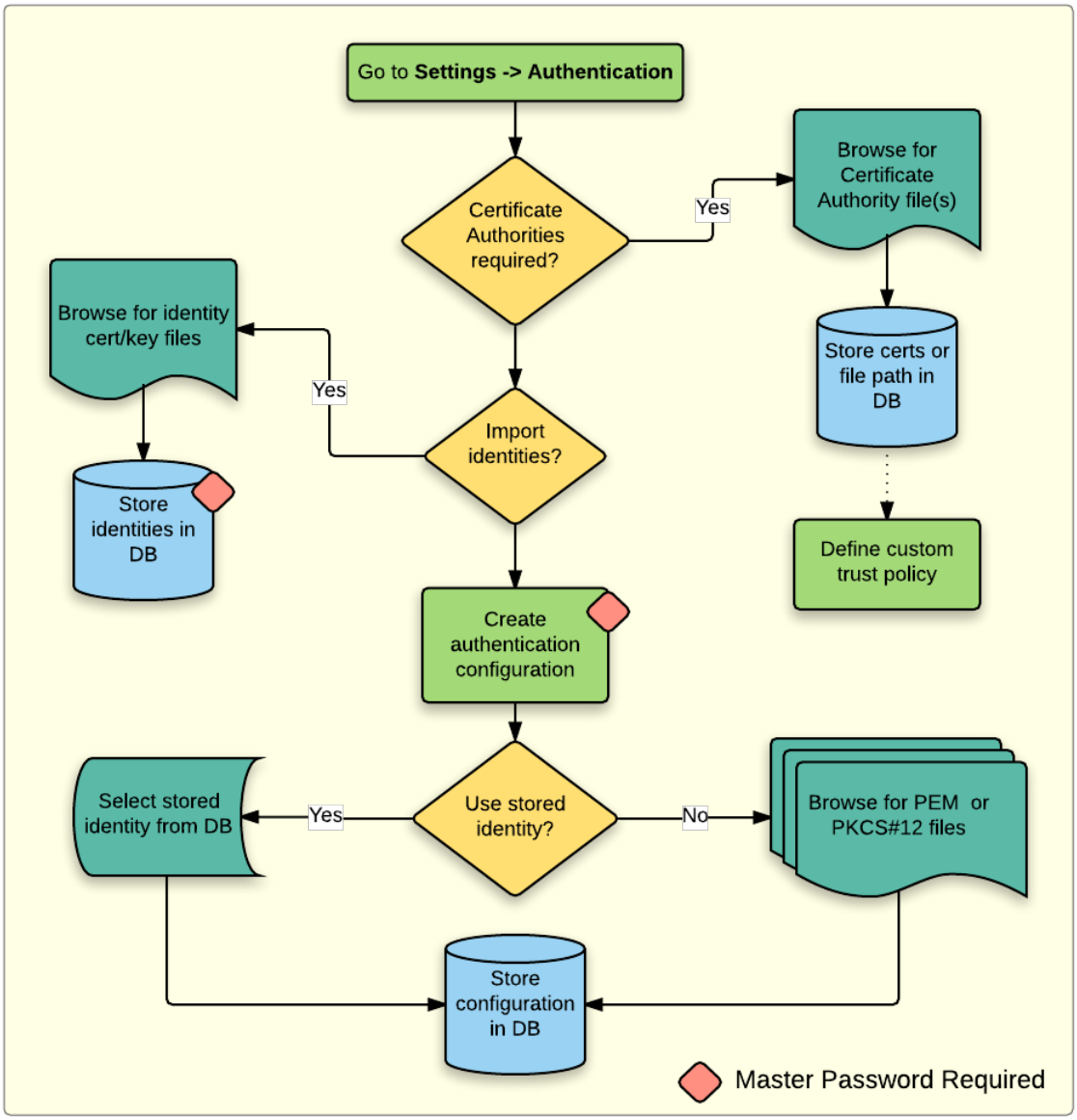


Figura 21.23: Flujo de trabajo de configuración de PKI

Todos los componentes de PKI se pueden administrar en editores separados dentro del **Administrador de certificados**, al que se puede acceder en la pestaña:guilabel:‘Authentication` en el cuadro de diálogo *Opciones* de QGIS (*Configuración -> Opciones*) haciendo clic en el botón *Administrar certificados*.

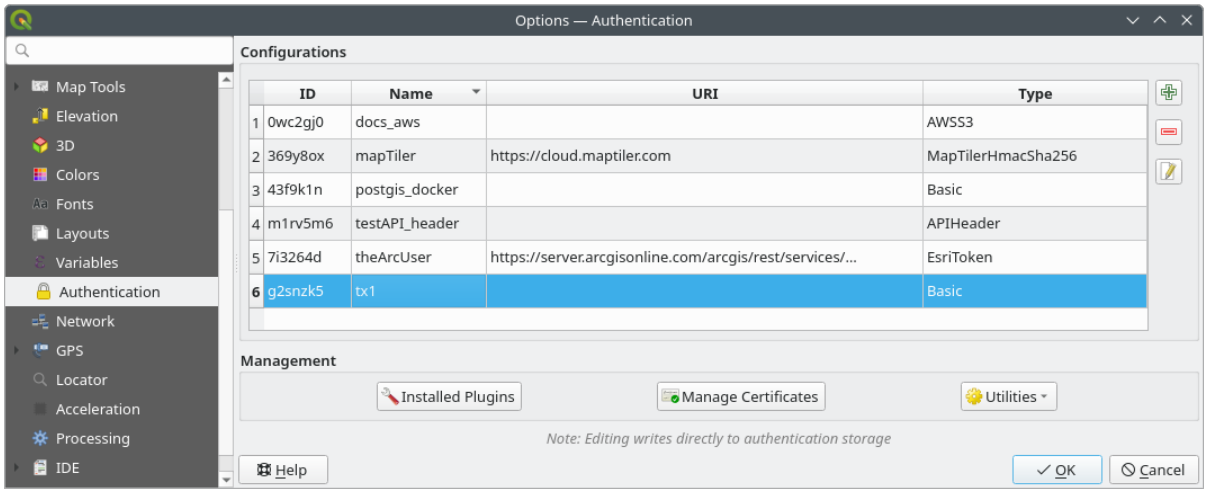


Figura 21.24: Abriendo el Administrador de Certificados

En *Administrador de Certificados*, hay editores para **Identidades**, **Servidores** y **Autoridades**. Cada uno de estos está contenido en sus propias pestañas y se describen a continuación en el orden en que se encuentran en el diagrama de flujo de trabajo anterior. El orden de las pestañas es relativo a los editores a los que se accede con frecuencia una vez que se acostumbre al flujo de trabajo.

Nota

Debido a que todas las ediciones del sistema de autenticación se escriben inmediatamente en la base de datos de autenticación, no es necesario hacer clic en el botón *Opciones Aceptar* para guardar los cambios. Esto es diferente a otras configuraciones en el cuadro de diálogo Opciones.

Autoridades

Puede administrar la certificación de Autoridades(CA) disponibles desde la pestaña **Autoridades** en el **Administrador de certificados** desde la pestaña **Autenticación** del cuadro de diálogo **Opciones** de QGIS.

Como se menciona en el diagrama de flujo de trabajo anterior, el primer paso es importar o hacer referencia a un archivo de CA. Este paso es opcional y puede que no sea necesario si su cadena de confianza de PKI se origina en las CA raíz ya instaladas en su sistema operativo (SO), como un certificado de un proveedor de certificados comerciales. Si su CA raíz de autenticación no se encuentra en las CA raíz de confianza del sistema operativo, deberá importarla o hacer referencia a la ruta del sistema de archivos. (Póngase en contacto con el administrador del sistema si no está seguro).

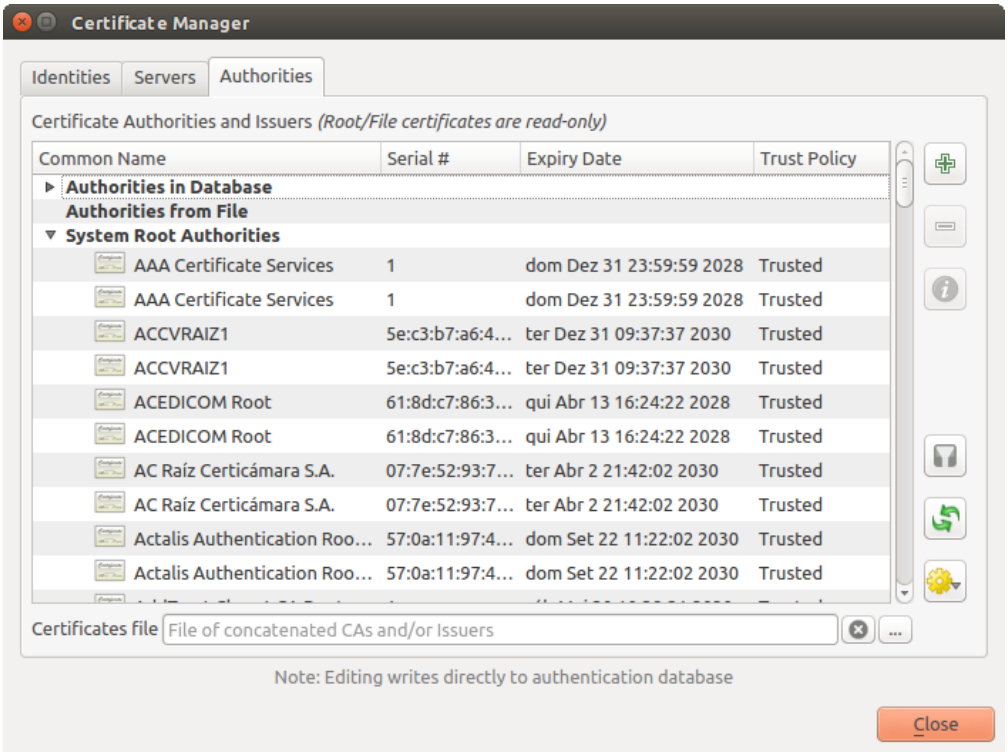


Figura 21.25: Editor de autoridades

De forma predeterminada, las CA raíz de su sistema operativo están disponibles; sin embargo, su configuración de confianza no se hereda. Debe revisar la configuración de la política de confianza del certificado, especialmente si las CA raíz de su sistema operativo han ajustado sus políticas. Cualquier certificado caducado se configurará como no confiable y no se utilizará en conexiones seguras de servidor, a menos que anule específicamente su política de confianza. Para ver la cadena de confianza detectable de QGIS para cualquier certificado, selecciónelo y haga clic en Mostrar información para el certificado.

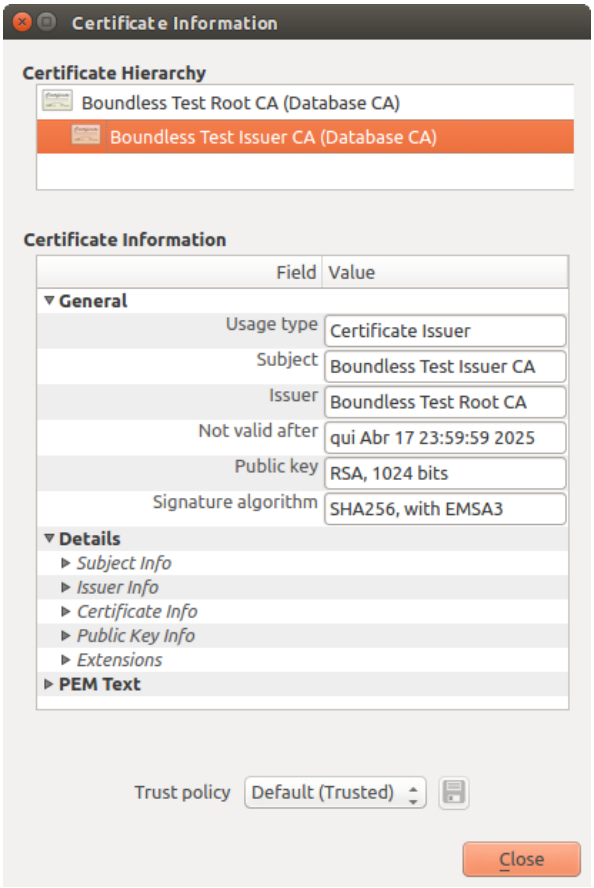


Figura 21.26: Dialogo de Información del certificado

Puede editar *Política de confianza*  para cualquier certificado seleccionado dentro de la cadena. Cualquier cambio en la política de confianza de un certificado seleccionado no se guardará en la base de datos a menos que el botón  Guardar cambio de política de confianza del certificado en la base de datos esté pulsado *para* certificación seleccionada. Cerrar el cuadro de diálogo **no** aplicará los cambios de política.

Puede revisar las CA filtradas, tanto certificados intermedios como raíz, que serán de confianza para conexiones seguras o cambiar la política de confianza predeterminada haciendo clic en el botón  **Opciones**.

Advertencia

Cambiar la política de confianza predeterminada puede resultar en problemas con las conexiones seguras.

Puede importar CA o guardar una ruta de sistema de archivos desde un archivo que contiene varias CA, o importar CA individuales. El formato PEM estándar para archivos que contienen múltiples certificaciones de cadena de CA tiene el certificado raíz en la parte inferior del archivo y todos los certificados secundarios firmados posteriormente arriba, hacia el comienzo del archivo.

El cuadro de diálogo de importación de certificados de CA encontrará todos los certificados de CA dentro del archivo, independientemente del orden, y también ofrece la opción de importar certificados que se consideran inválidos (en caso de que desee anular su política de confianza). Puede anular la política de confianza al realizar la importación o hacerlo más tarde en el editor de **Autoridades**.

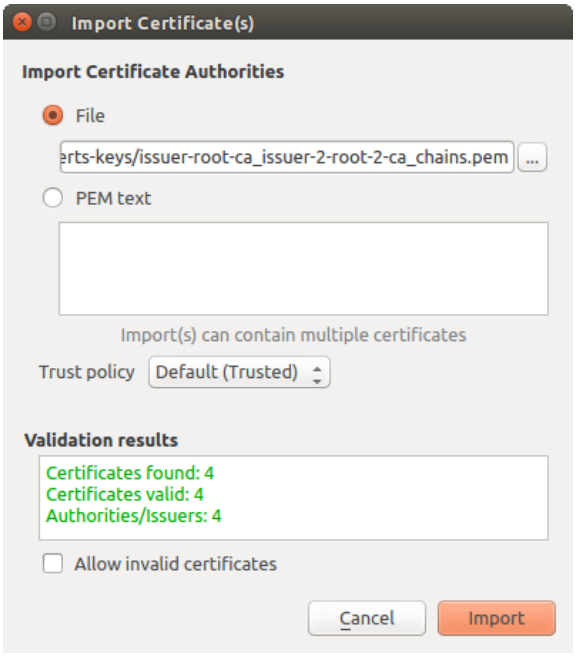


Figura 21.27: Diálogo importar certificados

Nota

Si está pegando información del certificado en el campo *Texto PEM*, tenga en cuenta que los certificados cifrados no son compatibles.

Identidades

Puede administrar los paquetes de identidad de cliente disponibles desde la pestaña *Identidades* en *Administrador de certificados* desde la pestaña ** Autenticación ** del cuadro de diálogo ** Opciones ** de QGIS. Una identidad es lo que lo autentica frente a un servicio habilitado para PKI y generalmente consta de un certificado de cliente y una clave privada, ya sea como archivos separados o combinados en un solo archivo «empaquetado». El paquete o la clave privada a menudo está protegida con una frase de contraseña.

Una vez que haya importado las Autoridades de certificación (CA), puede importar opcionalmente cualquier paquete de identidad a la base de datos de autenticación. Si no desea almacenar las identidades, puede hacer referencia a las rutas del sistema de archivos de sus componentes dentro de una configuración de autenticación individual.

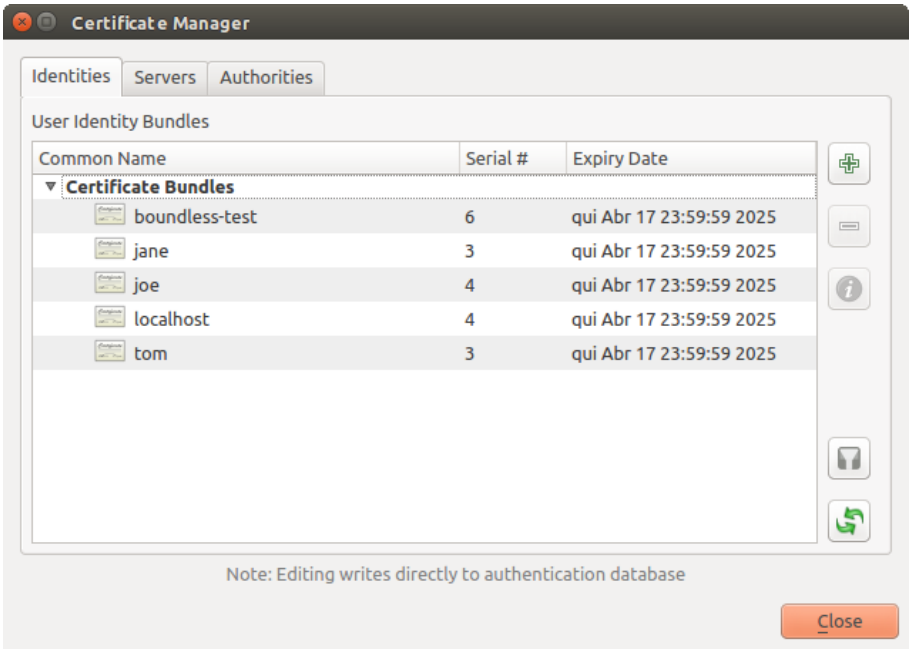


Figura 21.28: Editor de identidades

Al importar un paquete de identidad, puede estar protegido por contraseña o desprotegido, y puede contener certificados CA que forman una cadena de confianza. Las certificaciones de la cadena de confianza no se importarán aquí; se pueden agregar por separado en la pestaña *Autoridades*.

Tras la importación, el certificado y la clave privada del paquete se almacenarán en la base de datos, con el almacenamiento de la clave cifrado con la contraseña maestra de QGIS. El uso posterior del paquete almacenado de la base de datos solo requerirá la entrada de la contraseña maestra.

Se admiten paquetes de identidad personal que constan de componentes PEM/DER (.pem/.der) y PKCS#12 (.p12/.pfx). Si una clave o un paquete está protegido con contraseña, se requerirá la contraseña para validar el componente antes de la importación. Del mismo modo, si el certificado de cliente del paquete no es válido (por ejemplo, su fecha de vigencia aún no ha comenzado o ha transcurrido), el paquete no se puede importar.

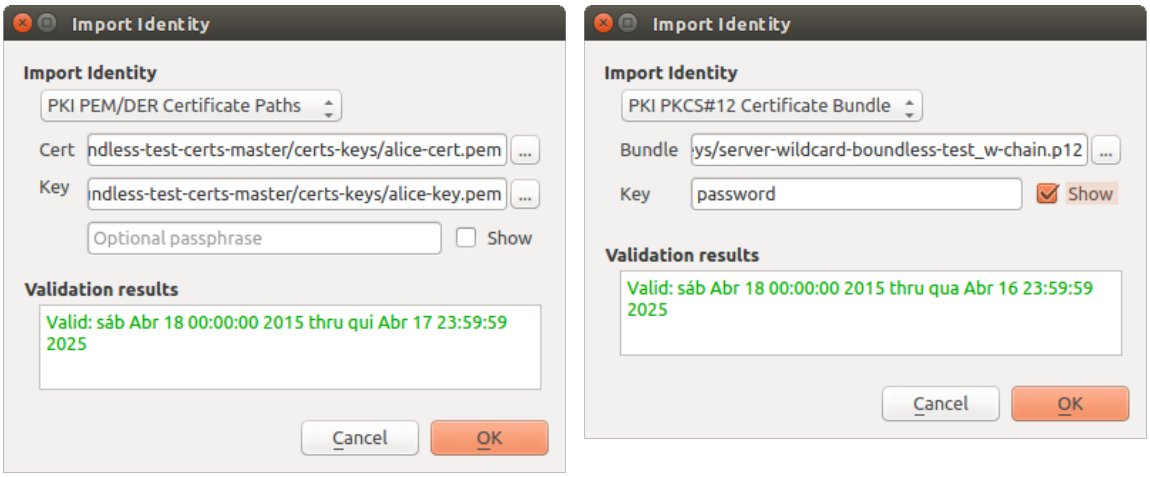


Figura 21.29: PEM/DER and PKCS#12 identities import

21.2.4 Manejar capas defectuosas

Ocasionalmente, el ID de configuración de autenticación que se guarda con un archivo de proyecto ya no es válido, posiblemente porque la base de datos de autenticación actual es diferente a la última vez que se guardó el proyecto, o debido a una discrepancia de credenciales. En tales casos, se presentará el diálogo *Manejar capas defectuosas* al iniciar QGIS.

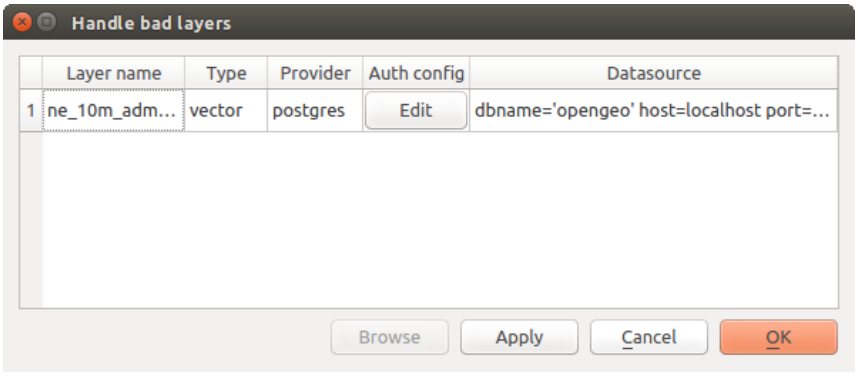


Figura 21.30: Manejar capas defectuosas con autenticación

Si se encuentra que una fuente de datos tiene un ID de configuración de autenticación asociado, podrá editarlo. Al hacerlo, se editará automáticamente la cadena de origen de datos, de la misma manera que se abre el archivo del proyecto en un editor de texto y se edita la cadena.

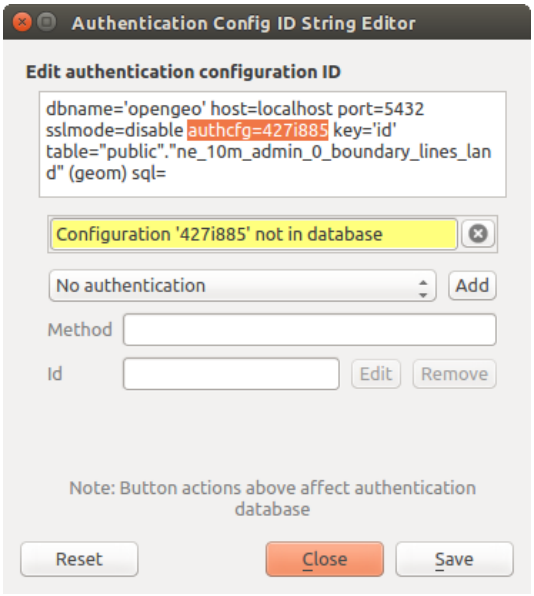


Figura 21.31: Editar ID de configuración de la autenticación

21.2.5 Cambiar el ID de configuración de autenticación

Ocasionalmente, deberá cambiar el ID de configuración de autenticación que está asociado con el acceso a un recurso. Hay casos en los que esto es útil:

- ** El ID de configuración de autenticación de recursos ya no es válido **: Esto puede ocurrir cuando ha cambiado las bases de datos de autenticación y agrega la necesidad de * alinear * una nueva configuración con el ID ya asociado con un recurso.
- ** Archivos de proyecto compartidos **: si pretendía compartir proyectos entre usuarios, p. Ej. a través de un servidor de archivos compartidos, puede * predefinir * un carácter de 7 (que contiene **a-z** y/o **0-9**) que está asociado con el recurso. Luego, los usuarios individuales cambian el ID de una configuración de autenticación que es específica para sus credenciales del recurso. Cuando se abre el proyecto, la identificación se encuentra en la base de datos de autenticación, pero las credenciales son diferentes por usuario.

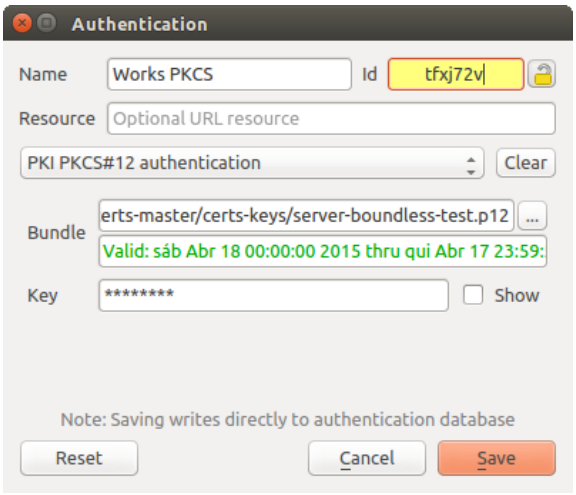


Figura 21.32: Cambiar el ID de configuración de autenticación de una capa (campo de texto amarillo desbloqueado)

Advertencia

Cambiar el ID de configuración de autenticación se considera una operación avanzada y solo debe realizarse con pleno conocimiento de por qué es necesario. Es por eso que hay un botón de bloqueo que debe hacer clic para desbloquear el campo de texto de la identificación antes de editar la identificación.

21.2.6 Soporte de Servidor QGIS

Cuando se usa un archivo de proyecto, con capas que tienen configuraciones de autenticación, como base para un mapa en QGIS Server, hay un par de pasos de configuración adicionales necesarios para que QGIS cargue los recursos:

- La base de datos de autenticación necesita estar disponible
- La contraseña maestra de la base de datos de autenticación debe estar disponible

Al instanciar el sistema de autenticación, el servidor creará o usará el archivo `qgis-auth.db` en el *user profile* activo, o el directorio definido por la variable de entorno `QGIS_AUTH_DB_DIR_PATH`. Puede ser que el usuario del servidor no tenga un directorio INICIO, en cuyo caso, use la variable de entorno para definir un directorio en el que el usuario del servidor tenga permisos de lectura/escritura y no se encuentre dentro de los directorios accesibles a través de la web.

Para pasar la contraseña maestra al servidor, escríbala en la primera línea del archivo en una ruta del sistema de archivos legible por el usuario de procesos del servidor y definida mediante la variable de entorno `QGIS_AUTH_PASSWORD_FILE`. Asegúrese de limitar el archivo como solo legible por el usuario del proceso del servidor y de no almacenar el archivo en directorios accesibles desde la web.

Nota

`QGIS_AUTH_PASSWORD_FILE` La variable se eliminará del entorno del servidor inmediatamente después de acceder.

21.2.7 excepciones de servidor SSL

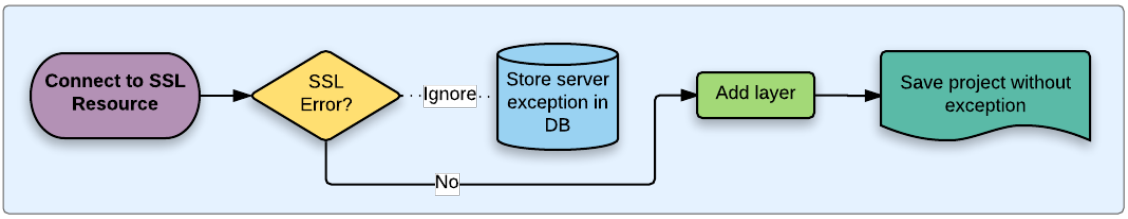


Figura 21.33: excepción de servidor SSL

Puede administrar las configuraciones y excepciones del servidor SSL desde la pestaña **Servidores** en la sección **Autenticación** del cuadro de diálogo **Opciones** de QGIS.

A veces, al conectarse a un servidor SSL, hay errores con el «protocolo de enlace» SSL o el certificado del servidor. Puede ignorar esos errores o crear una configuración de servidor SSL como excepción. Esto es similar a cómo los navegadores web le permiten anular los errores de SSL, pero con un control más granular.

Advertencia

No debe crear una configuración de servidor SSL a menos que tenga un conocimiento completo de toda la configuración SSL entre el servidor y el cliente. En su lugar, informe el problema al administrador del servidor.

Nota

Algunas configuraciones de PKI utilizan una cadena de confianza de CA completamente diferente para validar las identidades de los clientes que la cadena utilizada para validar el certificado del servidor SSL. En tales circunstancias, cualquier configuración creada para el servidor de conexión no necesariamente solucionará un problema con la validación de la identidad de su cliente, y solo el emisor de su identidad de cliente o el administrador del servidor pueden solucionar el problema.

Puede preconfigurar una configuración de servidor SSL haciendo clic en botón. Alternativamente, puede agregar una configuración cuando ocurre un error SSL durante una conexión y se le presenta un cuadro de diálogo **Error SSL** (donde el error puede ignorarse temporalmente o guardarse en la base de datos e ignorarse):

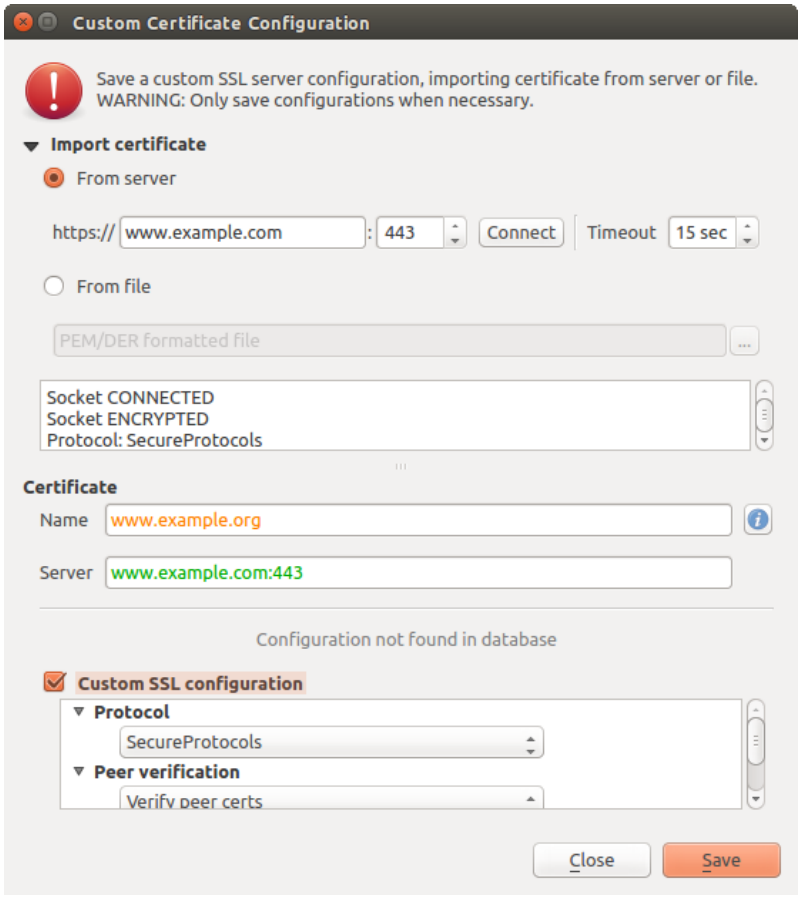


Figura 21.34: Adición manual de configuración

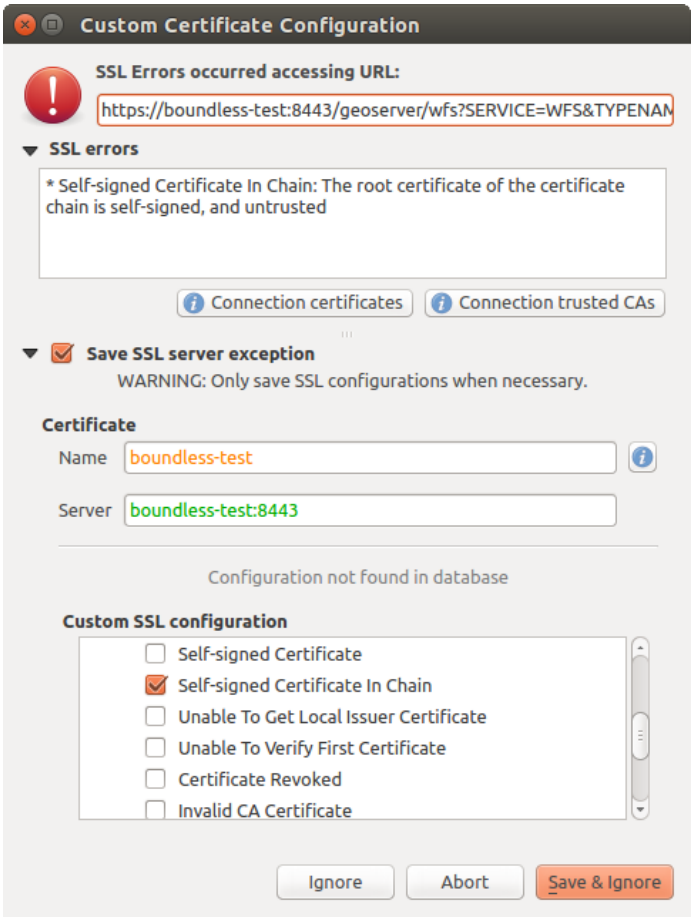


Figura 21.35: Adicionar configuración durante un error SSL

Una vez que se guarda una configuración SSL en la base de datos, se puede editar o eliminar.

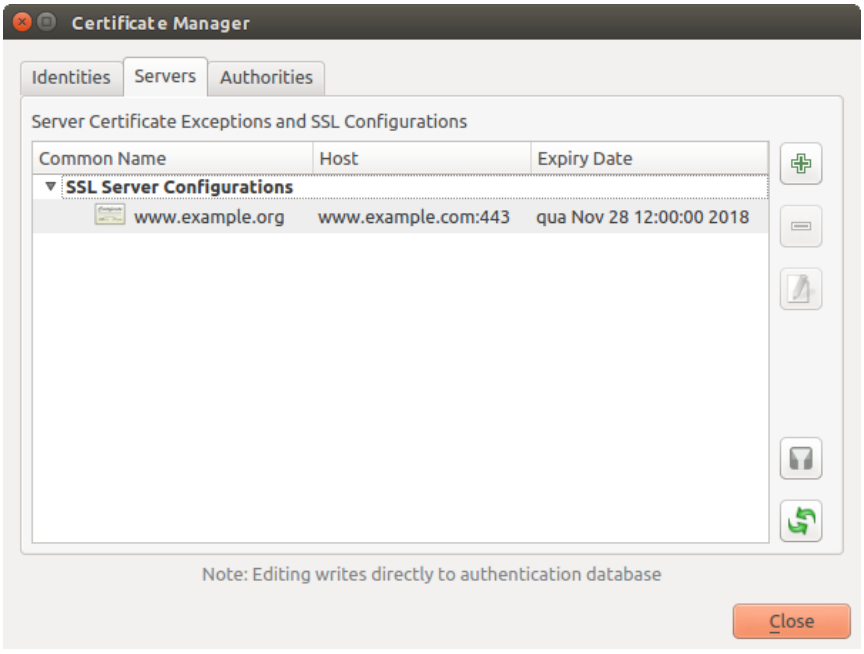


Figura 21.36: Configuración SSL existente

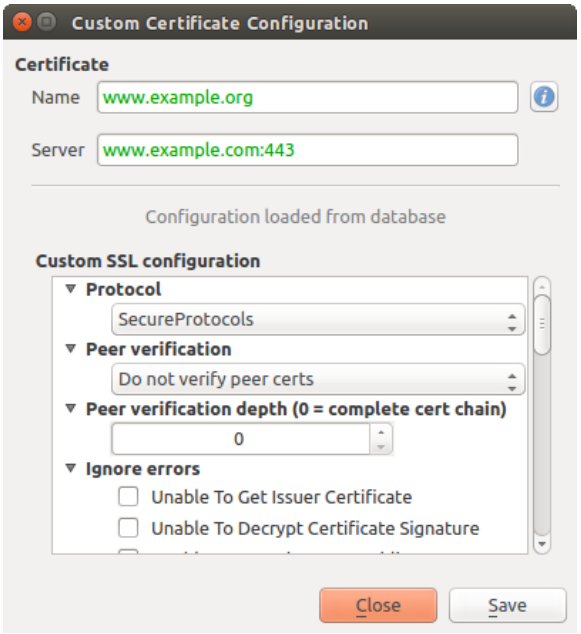


Figura 21.37: Editar una configuración SSL existente

Si desea preconfigurar una configuración SSL y el cuadro de diálogo de importación no funciona para la conexión de su servidor, puede activar manualmente una conexión a través de la **Consola Python** ejecutando el siguiente código (reemplace `https://bugreports.qt-project.org` con la URL de su servidor):

```
from qgis.PyQt.QtNetwork import QNetworkRequest
from qgis.PyQt.QtCore import QUrl
from qgis.core import QgsNetworkAccessManager

req = QNetworkRequest(QUrl('https://bugreports.qt-project.org'))
reply = QgsNetworkAccessManager.instance().get(req)
```

Esto abrirá un cuadro de diálogo de error SSL si se produce algún error, donde puede elegir guardar la configuración en la base de datos.

CAPÍTULO 22

Integracion GRASS SIG

La integración de GRASS proporciona acceso a las bases de datos y funcionalidades de GRASS GIS (ver GRASS-PROJECT en *Referencias bibliográficas y web*). La integración consta de dos partes: proveedor y complemento. El proveedor permite navegar, administrar y visualizar capas raster y vectoriales de GRASS. El complemento se puede utilizar para crear nuevas ubicaciones y conjuntos de mapas de GRASS, cambiar la región de GRASS, crear y editar capas vectoriales y analizar datos de GRASS 2-D y 3-D con más de 400 módulos de GRASS. En esta sección, presentaremos las funcionalidades del proveedor y el complemento y daremos algunos ejemplos de cómo administrar y trabajar con datos de GRASS.

El proveedor es compatible con GRASS versión 6 y 7, el complemento es compatible con GRASS 6 y 7 (a partir de QGIS 2.12). La distribución de QGIS puede contener un proveedor / complemento para GRASS 6 o GRASS 7 o para ambas versiones al mismo tiempo (los binarios tienen diferentes nombres de archivo). Sin embargo, solo se puede cargar una versión del proveedor/complemento en tiempo de ejecución.

22.1 Conjuntos de datos demostración

Como ejemplo, utilizaremos el conjunto de datos Alaska de QGIS (véase la sección *Descargando datos de muestra*). Incluye una pequeña muestra de GRASS `LOCATION` con tres capas vectoriales y un mapa de elevación raster. Cree una nueva carpeta llamada `grassdata`, descargue el conjunto de datos QGIS "Alaska" `qgis_sample_data.zip` de https://qgis.org/downloads-list/#data y descomprima el archivo en `grassdata`.

Más ejemplos de GRASS `LOCATIONs` están disponibles en el sitio web de GRASS.

22.2 Cargar capas ráster y vectorial de GRASS

Si el proveedor está cargado en QGIS, el elemento de ubicación con GRASS | grass | El icono se agrega en el árbol del navegador debajo de cada elemento de carpeta que contiene la ubicación de GRASS. Vaya a la carpeta `grassdata` y expanda la ubicación `alaska` y mapset `demo`.

Puede cargar capas ráster y vectoriales de GRASS como cualquier otra capa desde el navegador, ya sea haciendo doble clic en el elemento de capa o arrastrando y soltando al lienzo del mapa o la leyenda.

> **Truco**
>
> **Cargando datos GRASS**
>
> Si no ve el elemento de ubicación de GRASS, verifique en *Ayuda* ► *Acerca de* ► *Proveedores* si está cargado el proveedor de vectores GRASS.

22.3 Importar datos dentro de una UBICACIÓN DE GRASS mediante arrastrar y soltar

Esta sección ofrece un ejemplo de cómo importar datos ráster y vectoriales en un conjunto de mapas de GRASS.

1. En el navegador QGIS, navegue hasta el conjunto de mapas en el que desea importar datos.
2. En el navegador QGIS busque una capa que desee importar a GRASS, tenga en cuenta que puede abrir otra instancia del navegador (*Panel del navegador (2)*) si los datos de origen están demasiado lejos del conjunto de mapas en el árbol.
3. Arrastre una capa y suéltela en el conjunto de mapas de destino. La importación puede llevar algún tiempo para capas más grandes, verá el icono animado delante del nuevo elemento de capa hasta que finalice la importación.

Cuando los datos ráster están en diferentes SRC, pueden reproyectarse usando una transformación *Aproximada* (rápida) o *Exacta* (precisa). Si se crea un enlace al ráster de origen (utilizando `r.external`), los datos de origen están en el mismo CRS y el formato es conocido por GDAL, se utilizará el SRC de datos de origen. Puede configurar estas opciones en la pestaña *Navegador* en *Opciones GRASS*.

Si un ráster de origen tiene más bandas, se crea un nuevo mapa de GRASS para cada capa con el sufijo **.<band number>** y grupo de todos los mapas con Se crea el icono. Los rásteres externos tienen un icono diferente .

22.4 Administrar datos de GRASS en el navegador QGIS

- Copia de mapas: los mapas de GRASS se pueden copiar entre conjuntos de mapas dentro de la misma ubicación usando arrastrar y soltar.
- Eliminación de mapas: haga clic con el botón derecho en un mapa de GRASS y seleccione *Eliminar* del menú contextual.
- Cambiar el nombre de los mapas: haga clic con el botón derecho en un mapa de GRASS y seleccione *Renombrar* en el menú contextual.

22.5 Opciones GRASS

Las opciones de GRASS se pueden configurar en el cuadro de diálogo *Opciones de GRASS*, que se puede abrir haciendo clic con el botón derecho en la ubicación o el elemento del conjunto de mapas en el navegador y luego eligiendo *Opciones de GRASS*.

22.6 Iniciar el complemento GRASS

Para usar las funcionalidades de GRASS en QGIS, debe seleccionar y cargar el complemento de GRASS usando el Administrador de complementos. Para hacer esto, vaya al menú *Complementos* ->  *Administrar e instalar complementos* …, seleccione  *GRASS* y haga clic en *Aceptar*.

Las siguientes funciones principales se proporcionan con el menú de GRASS (*Complementos* -> *GRASS*) cuando inicia el complemento de GRASS:

-  Abrir directorio de mapas
-  Nuevo directorio de mapas
-  Cerrar directorio de mapas
-  Abrir herramientas de GRASS
-  Mostrar región actual de GRASS
-  Opciones de GRASS

22.7 Abrir directorio de mapas

Se debe abrir un conjunto de mapas de GRASS para obtener acceso a las herramientas de GRASS en el complemento (las herramientas están deshabilitadas si no hay ningún conjunto de mapas abierto). Puede abrir un conjunto de mapas desde el navegador: haga clic con el botón derecho en el elemento del conjunto de mapas y luego elija *Open mapset* del menú contextual.

22.8 LOCALIZACIÓN y DIRECTORIO DE MAPA GRASS

Los datos de GRASS se almacenan en un directorio llamado GISDBASE. Este directorio, a menudo llamado `grassdata`, debe ser creado antes de empezar a trabajar con el complemento de GRASS en QGIS. Dentro de este directorio, los datos SIG de GRASS están organizados por proyectos almacenados en subdirectorios llamados `LOCATIONs`. Cada `LOCATION` está definido por su sistema de coordenadas, proyección cartográfica y límites geográficos. Cada `LOCATION` puede tener varios `MAPSETs` (subdirectorios del `LOCATION`) que se utilizan para subdividir el proyecto en diferentes temas o subregiones, o como espacios de trabajo para miembros individuales del equipo (véase Neteler & Mitasova 2008 en *Referencias bibliográficas y web*). Para analizar capas vectoriales y raster con los módulos de GRASS, generalmente hay que importarlas en un `LOCATION` de GRASS. (Esto no es estrictamente cierto – con los módulos de GRASS `r.external` y `v.external` usted puede crear enlaces de solo lectura a datasets externos soportados por GDAL sin importarlos. Esta no es la forma usual de trabajar con GRASS para principiantes, por lo tanto esta funcionalidad no será descrita aquí).

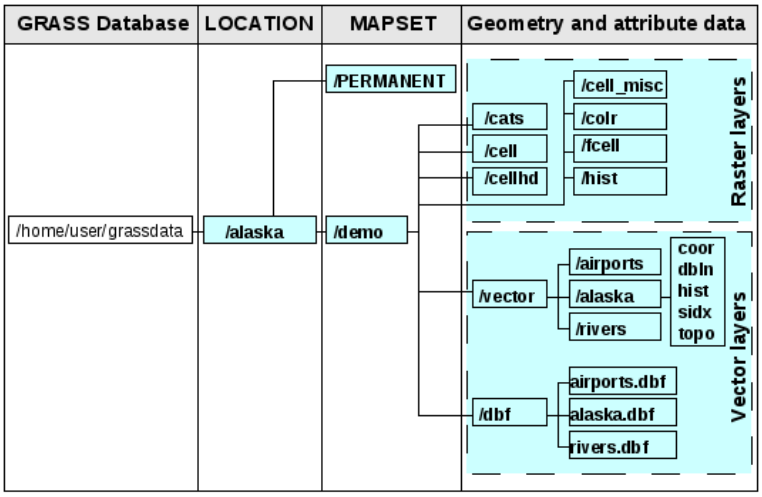


Figura 22.1: Datos GRASS en la LOCALIZACIÓN alaska

22.9 Importar datos dentro de una LOCALIZACIÓN DE GRASS

Ver sección *Importar datos dentro de una UBICACIÓN DE GRASS mediante arrastrar y soltar* para descubrir cómo se pueden importar fácilmente los datos arrastrando y soltando en el navegador.

Esta sección ofrece un ejemplo de cómo importar datos ráster y vectoriales en el archivo GRASS de "alaska" :file:"LOCATION" proporcionado por el conjunto de datos de QGIS "Alaska" de forma tradicional, utilizando módulos GRASS estándar. Por lo tanto, usamos el mapa ráster de cobertura terrestre `landcover.img` y el archivo vectorial GML :file:` lakes.gml` del conjunto de datos QGIS "Alaska" (ver *Descargando datos de muestra*).

1. Inicie QGIS y asegúrese que el complemento GRASS está cargado.
2. En la barra de herramientas de GRASS, haga clic en el icono Abrir directorio de mapas para que aparezca el asistente *DIRECTORIO DE MAPA*.
3. Seleccione como base de datos GRASS la carpeta: archivo: *grassdata* en el conjunto de datos de QGIS Alaska, como `LOCATION` "alaska", como `MAPSET` "demo" y haga clic en *Aceptar*.
4. Ahora haga clic en el icono Abrir herramientas de GRASS. El diálogo de la caja de herramientas de GRASS (vea la sección *La caja de herramientas GRASS*) aparece.
5. Para importar el mapa ráster `landcover.img`, haga clic en el modulo `r.in.gdal` en la pestaña *Árbol de módulos*. Este modulo GRASS le permite importar archivos ráster GDAL-admitidos en un `LOCALIZACIÓN` de GRASS. El diálogo del módulo para que `r.in.gdal` aparezca.
6. Navegue al navegador `raster` en el conjunto de datos de QGIS "Alaska" y seleccione el archivo `landcover.img`.
7. Como nombre de ráster saliente, define `landcover_grass` y click en *Ejecutar*. en la pestaña *Salida*, ve el comando GRASS que se está ejecutando actualmente `r.in.gdal -o input=/path/to/landcover.img output=landcover_grass`.
8. Cuando dice **finalizado satisfactoriamente**, click en *Ver salida*. La capa ráster `landcover_grass` ahora se importa a GRASS y se visualizará en el lienzo de QGIS.
9. Para importar el archivo vector GML `lakes.gml`, haga clic en el modulo `v.in.ogr` en la pestaña *Árbol de módulos*. Este modulo GRASS le permite importar archivos vectoriales OGR-admitidos en una `LOCAZACIÓN` de GRASS. El diálogo del modulo para que `v.in.ogr` aparezca.
10. Busque la carpeta `gml` en el conjunto de datos de QGIS "Alaska" y seleccione el archivo `lakes.gml` como archivo OGR.

11. Como nombre de salida del vector, defina `lakes_grass` y haga clic en *Ejecutar*. No tiene que preocuparse por las otras opciones en este ejemplo. En la pestaña *Salida*, verá el comando GRASS que se está ejecutando actualmente `v.in.ogr -o dsn=/path/to/lakes.gml output=lakes\_grass`.
12. Cuando diga **Finalizado satisfactoriamente**, haga clic en *Ver salida*. La capa vectorial `lakes_grass` ahora se importa a GRASS y se visualizará en el lienzo de QGIS.

22.9.1 Crear una nueva LOCALIZACIÓN GRASS

Como ejemplo, aquí está el ejemplo de GRASS `LOCATION alaska`, que se proyecta en la proyección Albers Equal Area usando pies como unidades. Este archivo de ejemplo de GRASS `LOCATION alaska` se utilizará para todos los ejemplos y ejercicios de las siguientes secciones relacionadas con GRASS. Es útil descargar e instalar el conjunto de datos en su computadora (ver:ref:*label_sampledata*).

1. Inicie QGIS y asegúrese que el complemento GRASS está cargado.
2. Visualice el shapefile `alaska.shp` (ver sección *Cargando una capa desde archivo*) del conjunto de datos QGIS Alaska (ver *Descargando datos de muestra*).
3. En la barra de herramientas de GRASS, haga clic en el icono  Nuevo mapaset para traer al wizard *MAPSET*.
4. Seleccione una carpeta de base de datos GRASS (GISDBASE) existente `grassdata`, o crea uno para la nueva `LOCATION` usando un administrador de archivos en su computadora. Luego haga clic en *Siguiente*.
5. Podemos usar este asistente para crear un nuevo `MAPSET` dentro de un `LOCATION` existente (ver sección *Añadir un nuevo DIRECTORIO DE MAPA*) o para crear un nuevo `LOCATION` por completo. Seleccione  *Crear nueva ubicación* (ver Figura 22.2).
6. Introduzca un nombre para `LOCATION` – nosostros usamos “alaska” – y haga click en *Siguiente*.
7. Defina la proyección haciendo clic en el botón de radio  *Proyección* para habilitar la lista de proyección.
8. Estamos usando la proyección Albers Equal Area Alaska (pies). Puesto que nos ha tocado saber que está representado por el EPSG ID 2964, ingresamos en el cuadro de búsqueda. (Nota: Si desea repetir este proceso para otra `LOCALIZACIÓN` y proyección y no ha memorizado el EPSG ID, haga clic en el icono  SRC actual en la esquina inferior derecha de la barra de estado (vea sección *Trabajar con Proyecciones*)).
9. En *Filtrar*, inserte 2964 para seleccionar la proyección.
10. Haz clic en *Siguiente*.
11. Para definir la región predeterminada, tenemos que ingresar los límites de `LOCATION` en dirección norte, sur, este, y oeste. Aquí, simplemente hacemos clic en el botón *Establecer la extensión actual de QGIS*, para aplicar la extensión de la capa cargada `alaska.shp` como la extensión de región predeterminada de GRASS.
12. Haz clic en *Siguiente*.
13. También necesitamos definir un `MAPSET` dentro de nuestro nueva `LOCATION` (esto es necesario al crear una nueva `LOCATION`). Puedes nombrarlo como quieras, usamos “demo”. GRASS crea automáticamente un especial `MAPSET` llamada `PERMANENT`, diseñado para almacenar los datos centrales del proyecto, su extensión espacial predeterminada y las definiciones del sistema de coordenadas (ver Neteler & Mitasova 2008 en *Referencias bibliográficas y web*).
14. Consulte el resumen para asegurarse de que sea correcto y haga clic en *Finalizar*.
15. La nueva `LOCALIZACIÓN`, “alaska”, y los dos `DIRECTORIO DE MAPAS`, “demo” y “PERMANENT”, son creados. El conjunto de trabajo abierto actualmente es “demo”, como se ha definido.
16. Tenga en cuenta que algunas de las herramientas en la barra de herramientas de GRASS que estaban inhabilitadas ahora están habilitadas.

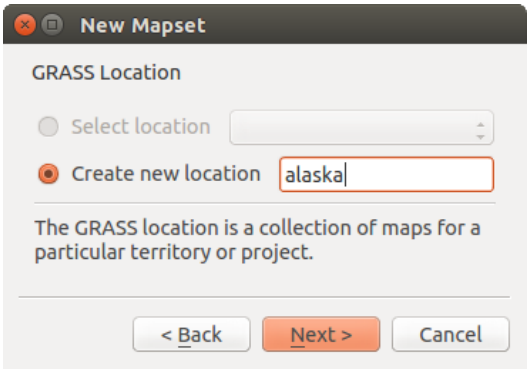


Figura 22.2: Cree una nueva LOCALIZACIÓN GRASS o un nuevo DIRECTORIO DE MAPA en QGIS

Si eso parecía como una gran cantidad de pasos, en realidad no es tan malo y una forma muy rápida de crear una LOCALIZACIÓN. La LOCALIZACIÓN "alaska" ahora está listo para la importación de datos (vea la sección *Importar datos dentro de una LOCALIZACIÓN DE GRASS*). También puede utilizar los datos ya existentes vectoriales y ráster en el ejemplo LOCALIZACIÓN "alaska" de GRASS, incluido en el conjunto de datos "Alaska" de QGIS *Descargando datos de muestra*, ya pasar a la sección *El modelo de datos vectoriales de GRASS*.

22.9.2 Añadir un nuevo DIRECTORIO DE MAPA

Un usuario sólo tiene acceso de escritura a un DIRECTORIO DE MAPA GRASS que él o ella crearon. Esto significa que además de acceder a su propio DIRECTORIO DE MAPA, se pueden leer mapas de otros usuarios" DIRECTORIO DE MAPAS (y ellos pueden leer los tuyos), pero sólo puede modificar o eliminar su propio DIRECTORIO DE MAPA.

Todos los DIRECTORIO DE MAPAS incluyen un archivo WIND que almacena los valores de las coordenadas de los límites actuales y la resolución ráster actualmente seleccionado (vea Neteler & Mitasova 2008 en *Referencias bibliográficas y web*, y la sección *La herramienta de región GRASS*).

1. Inicie QGIS y asegúrese que el complemento GRASS está cargado.
2. En la barra de herramientas de GRASS, haga clic en el icono  Nuevo mapaset para traer al wizard *MAPSET*.
3. Seleccione la base de datos GRASS (GISDBASE) la carpeta grassdata con la LOCALIZACIÓN "alaska", donde queremos añadir otro DIRECTORIO DE MAPA llamado "test".
4. Haz clic en *Siguiente*.
5. Podemos usar este asistente para crear un nuevo MAPSET dentro de una UBICACIÓN existente o para crear una nueva UBICACIÓN por completo. Haga click en el botón de opción  *Seleccionar ubicación* (ver Figura 22.2) y haga click en *Siguiente*.
6. Introduzca el nombre test para el nuevo MAPSET. Abajo en el asistente, verá una lista de MAPSETs y propietarios correspondientes.
7. Click en *Siguiente*, consulte el resumen para asegurarse de que esté todo correcto y haga clic en *Finalizar*.

22.10 El modelo de datos vectoriales de GRASS

Es importante comprender el Modelo de datos vectoriales de GRASS antes de digitalizar. En general, GRASS utiliza un modelo vectorial topológico. Esto significa que las áreas no se representan como polígonos cerrados, sino por uno o más límites. Un límite entre dos áreas adyacentes se digitaliza solo una vez y es compartido por ambas áreas. Los límites deben estar conectados y cerrados sin espacios. Un área se identifica (y etiqueta) por el **centroide** del área.

Además de los límites y centroides, un mapa vectorial también puede contener puntos y líneas. Todos estos elementos geométricos se pueden mezclar en una vector y serán representados en diferentes denominadas "capas" dentro de un mapa vectorial GRASS. Por lo que en GRASS, una capa no es un mapa vectorial o ráster pero un nivel dentro de una capa vectorial. Esto es importante para distinguir cuidadosamente. (aunque es imposible mezclar elementos, es inusual e incluso en GRASS, sólo se utiliza en casos especiales como análisis de redes vectoriales. Usualmente, se debe preferir almacenar diferentes elementos geométricos en diferentes capas.)

Es posible almacenar varias "capas" en un conjunto de datos vectoriales. Por ejemplo, campos, bosques y lagos se pueden almacenar en un vector. Un bosque y lago adyacente pueden compartir el mismo límite, pero tienen tablas de atributos separados. También es posible adjuntar atributos a límites. Un ejemplo podría ser el caso donde los límites entre un lago y un bosque es una carretera, por lo que puede tener una tabla de atributos diferente.

La "capa" del objeto espacial es definido por la "capa" dentro de GRASS. "Capa" es el número que define si hay más de una capa dentro del conjunto de datos (por ejemplo, si la geometría es bosque o lago). Por ahora, sólo puede ser un número. En el futuro, GRASS también implementara nombres como campos en la interfaz de usuario.

Los atributos se pueden almacenar dentro de la `LOCALIZACIÓN` en GRASS como dBase o SQLite3 o en tablas de base de datos externa, por ejemplo, PostgreSQL, MySQL, Oracle, etc.

Los atributos en la tabla de base de datos están enlazadas a los elementos geométricos utilizando un valor "categoría".

"Categoría" (llave, ID) es un entero adjunto a la geometría primitiva, y se utiliza como el enlace a una columna llave en la tabla de base de datos.

> **Truco**
>
> **Aprendizaje del modelo vectorial GRASS**
>
> La mejor manera de aprender el modelo vectorial GRASS y sus capacidades es descargar uno de los muchos tutoriales de GRASS donde se describe el modelo vectorial en más profundidad. Vea en https://grass.osgeo.org/learn/manuals/ más información, libros y tutoriales en diversos idiomas.

22.11 Crear una nueva capa vectorial GRASS

Para crear una nueva capa vectorial de GRASS, seleccione uno de los siguientes elementos del menú contextual del conjunto de mapas en el navegador:

- Nueva Capa de Puntos
- Nueva Capa de Líneas
- Nueva capa de polígonos

e ingrese un nombre en el diálogo. Se creará un nuevo mapa vectorial, se agregará una capa al lienzo y se iniciará la edición. La selección del tipo de capa no restringe los tipos de geometría que se pueden digitalizar en el mapa vectorial. En GRASS, es posible organizar todo tipo de tipos de geometría (punto, línea y polígono) en un mapa vectorial. El tipo solo se usa para agregar la capa al lienzo, porque QGIS requiere que una capa tenga un tipo específico.

También es posible agregar capas a mapas vectoriales existentes seleccionando uno de los elementos descritos anteriormente del menú contextual del mapa vectorial existente.

En GRASS, es posible organizar todo tipo de tipos de geometría (punto, línea y área) en una capa, porque GRASS utiliza un modelo de vector topológico, por lo que no es necesario seleccionar el tipo de geometría al crear un nuevo

vector de GRASS. Esto es diferente de la creación de shapefile con QGIS, porque los shapefiles usan el modelo vectorial de entidades simples (ver sección *Creando nuevas capas vectoriales*).

22.12 Digitalizar y editar una capa vectorial GRASS

Las capas vectoriales de GRASS se pueden digitalizar utilizando las herramientas de digitalización estándar de QGIS. Sin embargo, existen algunas particularidades que debe conocer debido a

- Modelo topológico de GRASS versus entidad simple de QGIS
- complejidad del modelo GRASS
  - múltiples capas en mapas únicos
  - múltiples tipos de geometría en un solo mapa
  - compartir geometría por múltiples entidades de múltiples capas

Las particularidades se discuten en las siguientes secciones.

Guardar, descartar cambios, deshacer, rehacer

> **Advertencia**
>
> Todos los cambios realizados durante la edición se escriben inmediatamente en el mapa vectorial y las tablas de atributos relacionados.

Los cambios se escriben después de cada operación, sin embargo, es posible deshacer/rehacer o descartar todos los cambios al cerrar la edición. Si se usa deshacer o descartar cambios, el estado original se reescribe en el mapa vectorial y las tablas de atributos.

Hay dos razones principales para este comportamiento:

- Es la naturaleza de los vectores GRASS que surge de la convicción de que el usuario quiere hacer lo que está haciendo y es mejor tener los datos guardados cuando el trabajo se interrumpe repentinamente (por ejemplo, apagón)
- La necesidad de una edición eficaz de los datos topológicos es la información visualizada sobre la corrección topológica; dicha información solo se puede adquirir del mapa vectorial de GRASS si se escriben cambios en el mapa.

Barra de herramientas

La “Barra de herramientas de digitalización” tiene algunas herramientas específicas cuando se edita una capa de GRASS:

Tabla 22.1: GRASS Digitizing Tools

Icono	Herramienta	Propósito
	Nuevo punto	Digitalizar un nuevo punto
	Nueva línea	Digitalizar nueva línea
	Nuevo límite	Digitalizar nuevo borde
	Nuevo centroide	Digitalizar nuevo centroide (etiqueta de área existente)
	Nuevo Contorno Cerrado	Digitalizar nuevo borde cerrado

Truco

Digitalizando polígonos en GRASS

Si desea crear un polígono en GRASS, primero debe digitalizar el límite del polígono. Luego agrega un centroide (punto de etiqueta) en el límite cerrado. La razón de esto es que un modelo vectorial topológico vincula la información de atributo de un polígono siempre al centroide y no al límite.

Categoría

La categoría, a menudo llamada gato, es una especie de identificación. El nombre proviene de tiempos en los que los vectores GRASS solo tenían un atributo «categoría». La categoría se utiliza como vínculo entre la geometría y los atributos. Una única geometría puede tener varias categorías y, por tanto, representar varias entidades en diferentes capas. Actualmente es posible asignar solo una categoría por capa usando las herramientas de edición de QGIS. Las nuevas funciones han asignado automáticamente una nueva categoría única, excepto los límites. Los límites generalmente solo forman áreas y no representan entidades lineales; sin embargo, es posible definir atributos para un límite más adelante, por ejemplo, en una capa diferente.

Las nuevas categorías siempre se crean solo en la capa que se está editando actualmente.

No es posible asignar más categorías a la geometría mediante la edición de QGIS, dichos datos se representan correctamente como múltiples entidades y las entidades individuales, incluso de diferentes capas, pueden eliminarse.

Atributos

Los atributos de la capa editada actualmente solo se pueden modificar. Si el mapa vectorial contiene más capas, las entidades de otras capas tendrán todos los atributos establecidos en "<not editable (layer #)>" para advertirle que dicho atributo no es editable. La razón es que otras capas pueden tener y generalmente tienen un conjunto de campos diferente, mientras que QGIS solo admite un conjunto fijo de campos por capa.

Si una primitiva de geometría no tiene una categoría asignada, se asigna automáticamente una nueva categoría única y se crea un nuevo registro en la tabla de atributos cuando se cambia un atributo de esa geometría.

Truco

Si desea realizar una actualización masiva de atributos en la tabla, por ejemplo, usando "Calculadora de campo" (*Usando la Calculadora de Campos*), y hay entidades sin categoría que no desea actualizar (generalmente límites), puede filtrar eliminarlos configurando "Filtro avanzado" en `cat is not null`.

Editar Estilo

La simbología topológica es esencial para la edición eficaz de datos topológicos. Cuando se inicia la edición, se establece un renderizador especializado "GRASS Edit" en la capa automáticamente y el renderizador original se restaura cuando se cierra la edición. El estilo se puede personalizar en la pestaña "Estilo" de las propiedades de capa. El estilo también se puede almacenar en un archivo de proyecto o en un archivo separado como cualquier otro estilo. Si personaliza el estilo, no cambie su nombre, ya que se utiliza para restablecer el estilo cuando se vuelve a iniciar la edición.

Truco

No guarde el archivo del proyecto cuando se edita la capa, la capa se almacenaría con "Editar estilo" que no tiene ningún significado si la capa no se edita.

El estilo se basa en información topológica que se agrega temporalmente a la tabla de atributos como campo "topo_symbol". El campo se elimina automáticamente cuando se cierra la edición.

Truco

No elimine el campo “topo_symbol” de la tabla de atributos, eso haría invisibles las entidades porque el renderizador se basa en esa columna.

Autoensamblado

Para formar un área, los vértices de los límites conectados deben tener **exactamente** las mismas coordenadas. Esto se puede lograr utilizando la herramienta de ajuste solo si el lienzo y el mapa vectorial tienen el mismo SRC. De lo contrario, debido a la conversión de coordenadas de mapa a lienzo y viceversa, la coordenada puede volverse ligeramente diferente debido a errores de representación y transformaciones de SRC.

Truco

Utilice el SRC de la capa también para el lienzo al editar.

Limitaciones

No se admite la edición simultánea de varias capas dentro del mismo vector al mismo tiempo. Esto se debe principalmente a la imposibilidad de manejar múltiples pilas de deshacer para una sola fuente de datos.

En Linux y macOS, solo se puede editar una capa de GRASS a la vez. Esto se debe a un error en GRASS que no permite cerrar los controladores de la base de datos en orden aleatorio. Esto se está resolviendo con los desarrolladores de GRASS.

Truco

Editar permisos de GRASS

Debe ser el propietario del `DIRECTORIO DE MAPA` de GRASS que desee editar. Es imposible editar capas de datos en un `DIRECTORIO DE MAPA` que no sea suyo, incluso si tiene permisos de escritura.

22.13 La herramienta de región GRASS

La definición de la región (ajuste una ventana de trabajo espacial) en GRASS es importante para trabajar con capas ráster. Análisis vectorial esta por defecto no limitado a cualquier definición de región definida. Pero todas los rásters recién creados tendrán la extensión espacial y resolución de la región GRASS definida actualmente, independientemente de su extensión y resolución original. La región GRASS actual es almacenada en el archivo `$LOCALIZACIÓN/$DIRECTORIO DE MAPA/WIND`, y define los límites norte, sur, este y oeste, número de columnas y filas, resolución espacial horizontal y vertical.

Es posible prender y apagar la visualización de la región GRASS en el lienzo de QGIS utilizando el botón 
Mostrar región actual de GRASS.

La región se puede modificar en la pestaña “Región” en el widget de muelle “GRASS Tolls”. Escriba los límites y la resolución de la nueva región y haga clic en *Aplicar*. Si hace clic en *Seleccionar la extensión arrastrando sobre el lienzo*, puede seleccionar una nueva región de forma interactiva con el ratón en el lienzo de QGIS arrastrando un rectángulo.

El modulo GRASS `g.region` proporciona muchos más parámetros para definir una extensión de región apropiada y resolución para su análisis ráster. Se puede utilizar estos parámetros con la caja de herramientas GRASS, descrito en la sección *La caja de herramientas GRASS*.

22.14 La caja de herramientas GRASS

La caja  Abrir herramientas GRASS proporciona funcionalidades de modulo GRASS para trabajar con datos dentro un `LOCALIZACIÓN` GRASS seleccionada y `DIRECTORIO DE MAPA`. Para utilizar la caja de herramientas GRASS necesita abrir una `LOCALIZACIÓN` y `DIRECTORIO DE MAPA` que tiene permisos de escritura para (normalmente se concede si ha creado el `DIRECTORIO DE MAPA`). Esto es necesario, porque las capas ráster y vector creadas durante análisis necesitan estar escritos a la `LOCALIZACIÓN` y `DIRECTORIO DE MAPA` actualmente

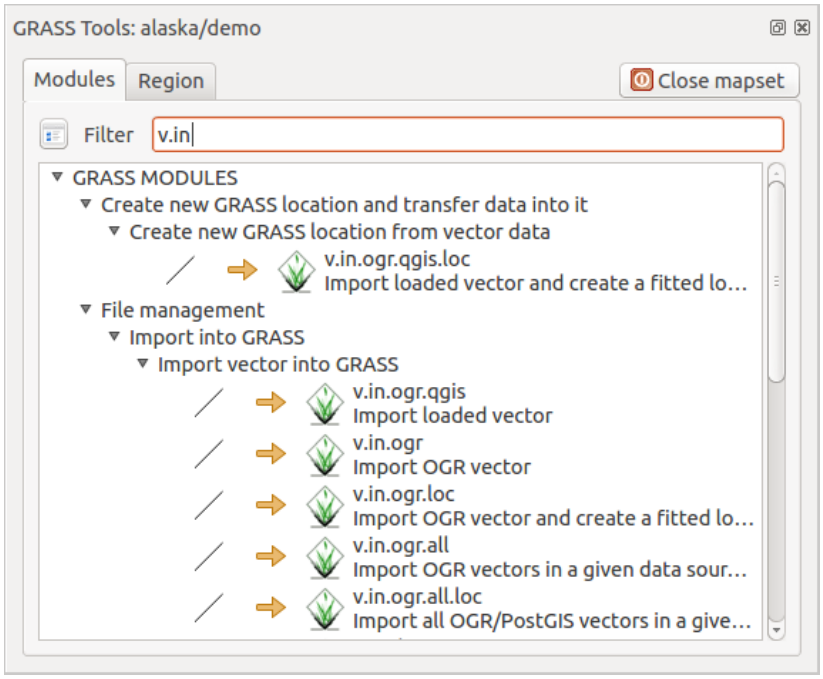


Figura 22.3: Caja de herramienta GRASS y módulo

22.14.1 Trabajando con módulos GRASS

The GRASS plugin provides a user-friendly toolbox with about 200 of the available GRASS modules and functionalities.

Una lista completa de los módulos GRASS disponibles en la caja de herramientas gráfica en la versión QGIS 3.44 está disponible en la wiki de GRASS en https://grasswiki.osgeo.org/wiki/GRASS-QGIS_relevant_module_list.

También es posible personalizar el contenido de la caja de herramientas GRASS. Este procedimiento se describe en la sección *Personalizar la caja de herramientas GRASS*.

Como se mostró en Figura 22.3, puede buscar el módulo GRASS apropiado usando la pestaña agrupada temáticamente *Árbol de módulos* o la pestaña de búsqueda *Lista de módulos*.

Al hacer clic en un icono de modulo gráfico, una nueva pestaña se añadirá al diálogo de Caja de herramientas, proporciona tres nuevas sub-pestañas *Opciones*, *Salida* y *Manual*.

Opciones

La pestaña *Opciones* proporciona un cuadro de diálogo de módulo simplificado donde normalmente puede seleccionar una capa ráster o vectorial visualizada en el lienzo de QGIS e ingresar más parámetros específicos del módulo para ejecutar el módulo.

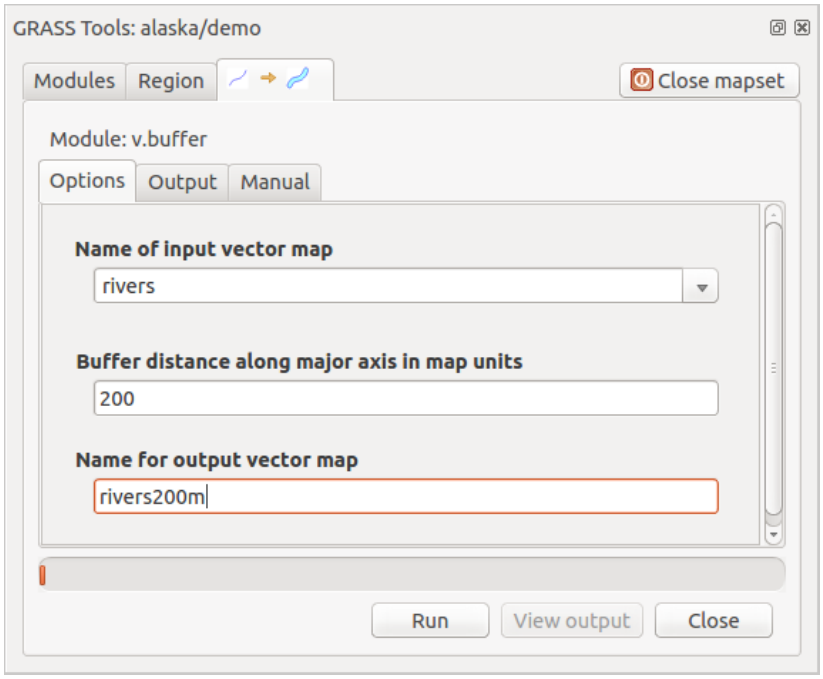


Figura 22.4: Opciones del módulo de la caja de herramientas GRASS

Los parámetros del módulo proporcionados a menudo no están completos para que el diálogo sea sencillo. Si desea utilizar más parámetros y banderas del módulo, debe iniciar el shell de GRASS y ejecutar el módulo en la línea de comandos.

Una nueva característica desde QGIS 1.8 es la implementación de un botón *Mostrar opciones avanzadas* abajo del diálogo del modulo simplificado en la pestaña *Opciones*. Por el momento, sólo se añade al modulo `v.in.ascii` como ejemplo de uso, pero probablemente es parte de más o todos los módulos en la caja de herramientas GRASS en futuras versiones de QGIS. Esto le permite usar las opciones del modulo de GRASS completo sin la necesidad de cambiar la consola de GRASS.

Salida

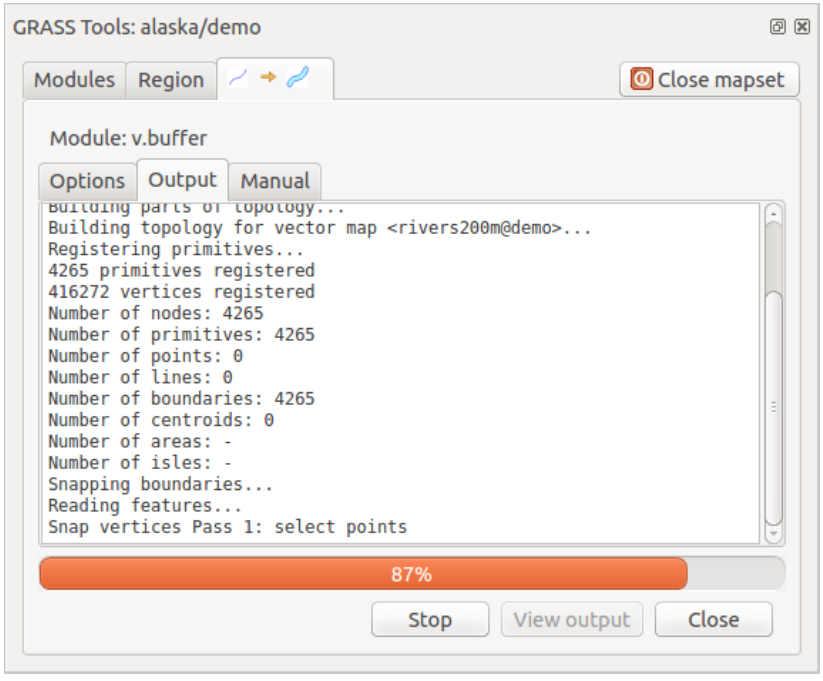


Figura 22.5: Salida del módulo de la caja de herramientas GRASS

La pestaña *Salida* proporciona información sobre el estado de salida del módulo. Cuando hace clic en el botón *Ejecutar*, el módulo cambia a la pestaña *Salida* y ve información sobre el proceso de análisis. Si todo funciona bien, finalmente verá un mensaje de `Finalizado correctamente`.

Manual

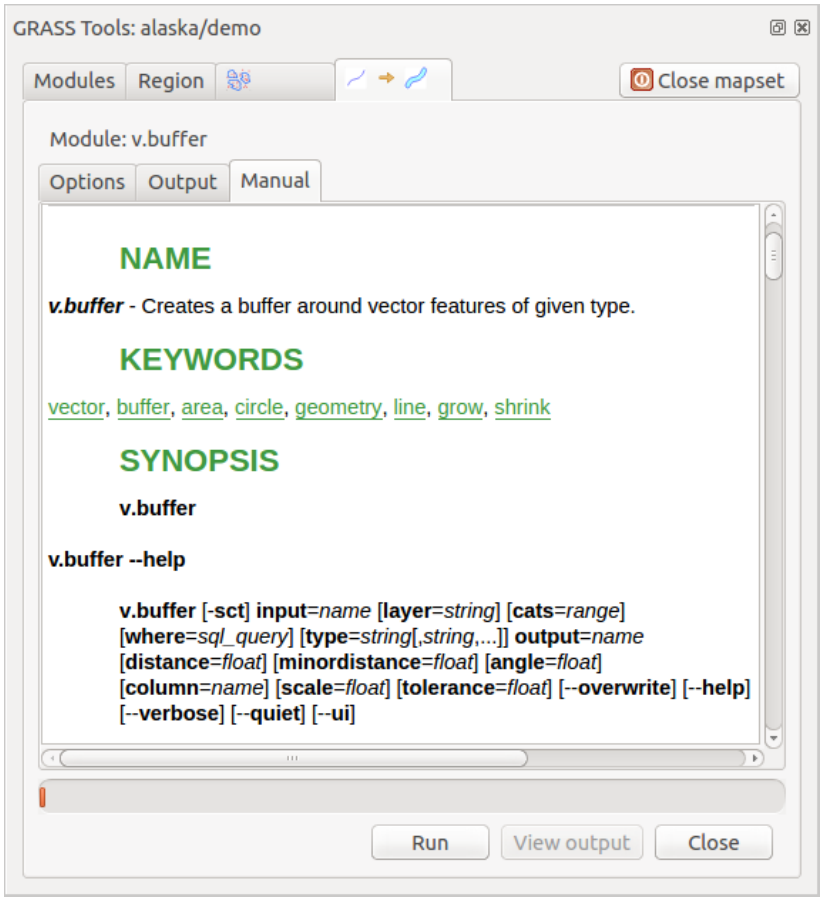


Figura 22.6: Módulo Manual de la caja de herramientas GRASS

La pestaña *Manual* muestra la página de ayuda HTML del modulo GRASS. Se puede utilizar para comprobar otros parámetros de los módulos y las banderas o para obtener un conocimiento más profundo acerca de la finalidad del módulo. Al final de cada página del manual del módulo, se ven otros enlaces al `Índice de ayuda principal`, al `Índice temático` y al `Índice completo`. Estos enlaces proporcionar información de ejemplo como el modulo `g.manual`.

Truco

Mostrar resultados inmediatamente

Si desea mostrar sus resultados de cálculo inmediatamente en su lienzo de mapa, se puede utilizar el botón “Ver Salida” en la parte inferior de la pestaña de módulo.

22.14.2 Ejemplos del módulo GRASS

Los siguientes ejemplos demostrarán el poder de algunos módulos GRASS.

Crear curvas de nivel

El primer ejemplo crea un mapa de curvas de nivel vectoriales de un ráster de elevación (DEM). Aquí, se asume que se tiene `LOCALIZACIÓN` Alaska configurado como se explica en la sección *Importar datos dentro de una LOCALIZACIÓN DE GRASS*.

- Primero, abra la ubicación haciendo clic en el botón  Abrir mapset y elegir la ubicación de Alaska.
- Ahora abra la caja de herramientas con el botón  Abrir herramientas de GRASS.
- En la lista de categorías de herramientas, haga doble clic *Ráster* ► *Administración de superficie*► *Generar curvas de nivel vectoriales*.
- Ahora, un solo clic en la herramienta **r.contour** abrirá el cuadro de diálogo de la herramienta como se explicó anteriormente (ver *Trabajando con módulos GRASS*).
- En el *Nombre del mapa ráster de entrada* introduce `gtopo30`.
- Escriba en *Incremento entre niveles de contorno*  el valor 100 (esto creará curvas de nivel a intervalos de 100 metros).
- Escriba en *Nombre de salida del mapa vectorial* el nombre `ctour_100`.
- Haga clic en *Ejecutar* para iniciar el proceso. Espere unos momentos hasta que aparezca el mensaje `Finalizado satisfactoriamente` en la ventana de salida. Luego haga clic en *Ver salida* y *Cerrar*.

Dado que esta es una región grande, tomará un tiempo para mostrarla. Después de que termine la presentación, puede abrir la ventana de propiedades de la capa para cambiar el color de línea así el contorno aparece claramente sobre el ráster de elevación, como en *El Diálogo de las Propiedades del Vector*.

El siguiente acercamiento a una pequeña y montañosa área en el centro de Alaska. Al acercarse, se puede observar que las curvas de nivel tienen esquinas afiladas. GRASS ofrece la herramienta **v.generalize** para alterar ligeramente mapas vectoriales, manteniendo su forma general. La herramienta utiliza varios algoritmos diferentes con propósitos diferentes. Algunos de los algoritmos (es decir, Douglas Peucker y Vertex Reduction) simplificar la línea mediante la eliminación de algunos de los vértices. El vector resultante se carga más rápido. Este proceso es útil cuando se tiene un vector muy detallado, pero va a crear un mapa de escala muy pequeña, por lo que el detalle es innecesario.

> **Truco**
>
> **La herramienta de simplificar**
>
> Tenga en cuenta que QGIS tiene una *Vectorial -> Herramientas de Geometría -> Simplificar Geometría* que funciona como el algoritmo de GRASS **v.generalize** Douglas-Peuker.

Sin embargo, el proposito de este ejemplo es diferente. Las líneas de curvas de nivel creadas por `r.contour` tiene ángulos agudos que deben ser suavizados. Entre el algoritmo **v.generalize** hay Chaiken's, lo que hace precisamente eso (también astillas de Hermite). Tenga en cuenta que estos algoritmos se pueden **añadir** vértices adicionales al vector, haciendo que se cargue más lentamente

- Abra la caja de herramientas GRASS y haga doble clic en las categorías *Vectorial* ► *Desarrollar mapa* ► *Generalización*,
- Compruebe que el vector "ctour_100" aparece como el *Nombre del vector de entrada*.
- De la lista de algoritmos, elija Chaiken's. Deje todas las demás opciones en sus valores predeterminados y desplácese hacia abajo hasta la última fila para ingresar en el campo *Nombre del mapa vectorial de salida* "ctour_100_smooth", y haga click en *Ejecutar*.

- El proceso lleva varios minutos. Una vez que aparezca `Finalizado con éxito` en las ventanas de salida, haga clic en *Ver salida* y luego *Cerrar*.
- Se puede cambiar el color del vector para que se muestre claramente sobre el fondo del ráster y para contrastar con las curvas de nivel originales. Se dará cuenta de que las nuevas curvas de nivel tienen esquinas más suaves que el original durante su estancia fiel a la original de forma general.



Figura 22.7: Módulo GRASS v.generalize para suavizar un mapa vectorial

Truco

Otros usos de r.contour

El proceso descrito anteriormente se puede utilizar en otras situaciones equivalentes. Si tiene un mapa ráster de datos de precipitación, por ejemplo, entonces el mismo método se utilizará para crear un mapa vectorial de líneas isoyetas (lluvia constante).

Crear un efecto sombreado de relieve 3-D

Varios métodos se utilizan para mostrar capas de elevación y da un efecto #-D a mapas. El uso de líneas de curvas de nivel, como se mostro anteriormente, es un método popular regularmente elejido para producir mapas topológicos. El efecto de sombreado se crea de un ráster (elevación) DEM calculando primero la pendiente y el aspecto de cada celda, entonces simula la posición del sol en el cielo y da un valor de reflectancia a cada celda. De este modo se obtienen pendientes frente al sol iluminadas; las pendientes orientadas lejos del sol(en la sombra) se oscurecen.

- Comience este ejemplo cargando el ráster de elevación `gtopo30`. Inicie la caja de herramientas GRASS y bajo la categoría ráster, haga doble clic para abrir *Análisis espacial* ► *Análisis del terreno*.
- A continuación haga clic en **r.shaded.relief** para abrir el módulo.
- Cambie el *ángulo del azimut* 1,00 270 a 315
- Ingrese `gtopo30_shade` para el nuevo ráster de sombreado y haga clic en *Ejecutar*.
- Cuando el proceso finalice, añada el ráster de mapa de sombras al mapa. Debe verlo desplegado en escala de grises.
- Para ver ambos sombreados y los colores de la *gtopo30*` juntos, mueva el mapa de sombreado abajo del mapa `gtopo30` en la tabla de contenido, a continuación abra la ventana *Propiedades* de `gtopo30`, cambie a la pestaña de *Transparencia* y establezca su nivel de transparencia a cerca de 25%.

Ahora debe tener la elevación `gtopo30` con su mapa de color y ajuste de transparencia mostrado **arriba** el mapa de sombras en escala de grises. Con el fin de ver los efectos visuales en el mapa de sombras, apague el mapa `gtopo30_shade`, a continuación, vuelva a encenderla.

Estadísticas de ráster en un mapa vectorial

El siguiente ejemplo muestra como un módulo GRASS puede agregar datos ráster y añadir columnas de una estadística para cada polígono en un mapa vectorial.

- Nuevamente usando los datos de Alaska, consulte *Importar datos dentro de una LOCALIZACIÓN DE GRASS* para importar el archivo `shapefiles/trees.shp` en GRASS.
- Ahora un paso intermedio es necesario: los centroides se deben añadir al mapa de árboles importado para que sea una zona de GRASS completa del vector (incluyendo ambos límites y centroides).
- De la caja de herramientas, elija *Vectorial* ► *Desarrollar mapa* -> *Administrar objetos espaciales*, y abra el modulo **v.centroids**.
- Introduzca como el *Mapa vectorial de salida* "forest_areas" y ejecute el módulo.
- Ahora cargue el vector `forest_areas` y muestre los tipos de bosque - caducifolio, árbol de hoja perenne, mixto - en diferentes colores: En la ventana *Propiedades* de la capa, la pestaña *Simbología*, elija de *Tipo de leyenda*  "Valor único" y establezca el *Campo de clasificación* a "VEGDESC". (Consulte la explicación de la pestaña de simbología en *Propiedades de simbología* de la sección vectorial.)
- A continuación vuelva a abrir la caja de herramientas de GRASS y abra *Vectorial* ► *Actualizacion vectorial por otros mapas*.
- Haga clic en el modulo **v.rast.stats**. Ingrese `gtopo30` y `forest_areas`.
- Solo se necesita un parámetro adicional: Ingrese *prefijo de columna* `elev` y haga clic en *Ejecutar*. Esta es una operación computacionalmente pesada, que se ejecutará durante mucho tiempo (probablemente hasta dos horas).
- Finalmente, abra la tabla de atributos `forest_areas`, y verifique que varias de las nuevas columnas se han añadido, incluyendo `elev_min`, `elev_max`, `elev_mean`, etc., para cada polígono de bosque.

22.14.3 Personalizar la caja de herramientas GRASS

Casi todos los módulos de GRASS se pueden añadir a la caja de herramientas de GRASS. Una interfaz XML se proporciona para analizar los archivos XML muy sencillos que configuran la apariencia y los parámetros de los módulos dentro de la caja de herramientas.

Un ejemplo del archivo XML para generar el módulo `v.buffer` (v.buffer.qgm) luce como esto:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE qgisgrassmodule SYSTEM "http://mrcc.com/qgisgrassmodule.dtd">

<qgisgrassmodule label="Vector buffer" module="v.buffer">
        <option key="input" typeoption="type" layeroption="layer" />
        <option key="buffer"/>
        <option key="output" />
</qgisgrassmodule>
```

El analizador lee esta definición y crea una nueva pestaña dentro de la caja de herramientas cuando se selecciona el módulo.

CAPÍTULO 23

Entorno de trabajo de procesamiento de QGIS

23.1 Introducción

Este capítulo introduce al marco de procesamiento de QGIS, un entorno de geoprosesamiento que se puede utilizar para llamar algoritmos nativos o de terceros de QGIS, haciendo su tarea de análisis espacial más productivo y fácil de lograr.

Como un *Complemento de núcleo*, El procesamiento está instalado de forma predeterminada, pero debe activarlo:

1. Vaya a *Complementos* ► *Administrar e instalar Complementos…*
2. Click en la pestaña *instalados* a la izquierda
3. Marque la casilla junto a la entrada  *Processing*
4. Cierra el cuadro de diálogo.

   Un menú *Processing* ahora está disponible en la barra de menú superior. Desde allí puede llegar a los principales componentes de este marco.

En las siguientes secciones, revisaremos cómo usar los elementos gráficos de este sistema y sacar el máximo provecho de cada uno de ellos.

Hay cuatro elementos básicos en el marco de la GUI, que se utilizan para ejecutar algoritmos para diferentes propósitos. Elegir una herramienta u otra dependerá del tipo de análisis que se realizará y las características particulares de cada usuario y proyecto. Se puede acceder a todos ellos (a excepción de la interfaz de procesamiento por lotes, que se llama desde la caja de herramientas o el diálogo de ejecución del algoritmo, como veremos) desde el elemento de menú *Processing* (verá más entradas; el resto los no se utilizan para ejecutar algoritmos y se explicarán más adelante en este capítulo).

- La *Toolbox*: El elemento principal de la GUI, se utiliza para ejecutar un solo algoritmo o ejecutar un proceso por lotes basado en ese algoritmo.

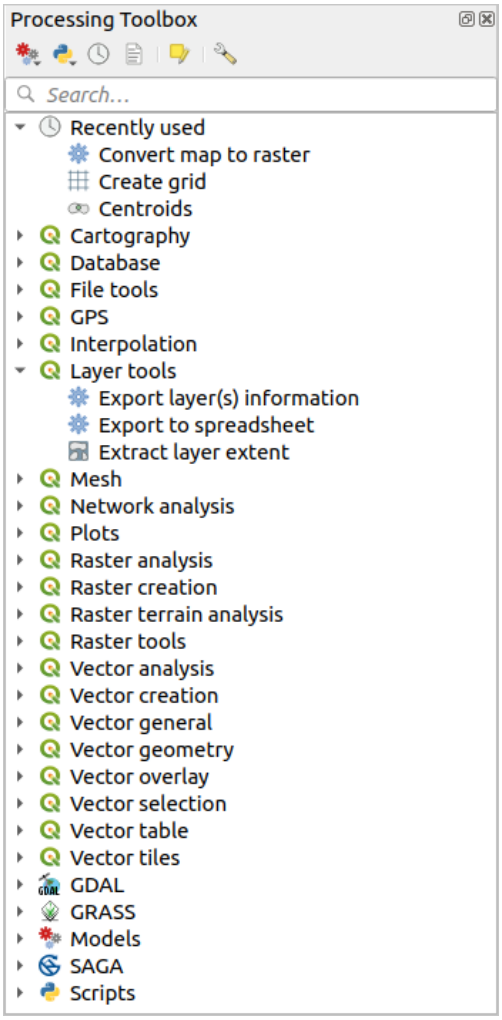


Figura 23.1: Caja de Herramientas de Procesamiento

- El *Diseñador de modelos*: Varios algoritmos pueden combinarse gráficamente utilizando el modelador para definir un flujo de trabajo, creando un único proceso que implique varios subprocesos.

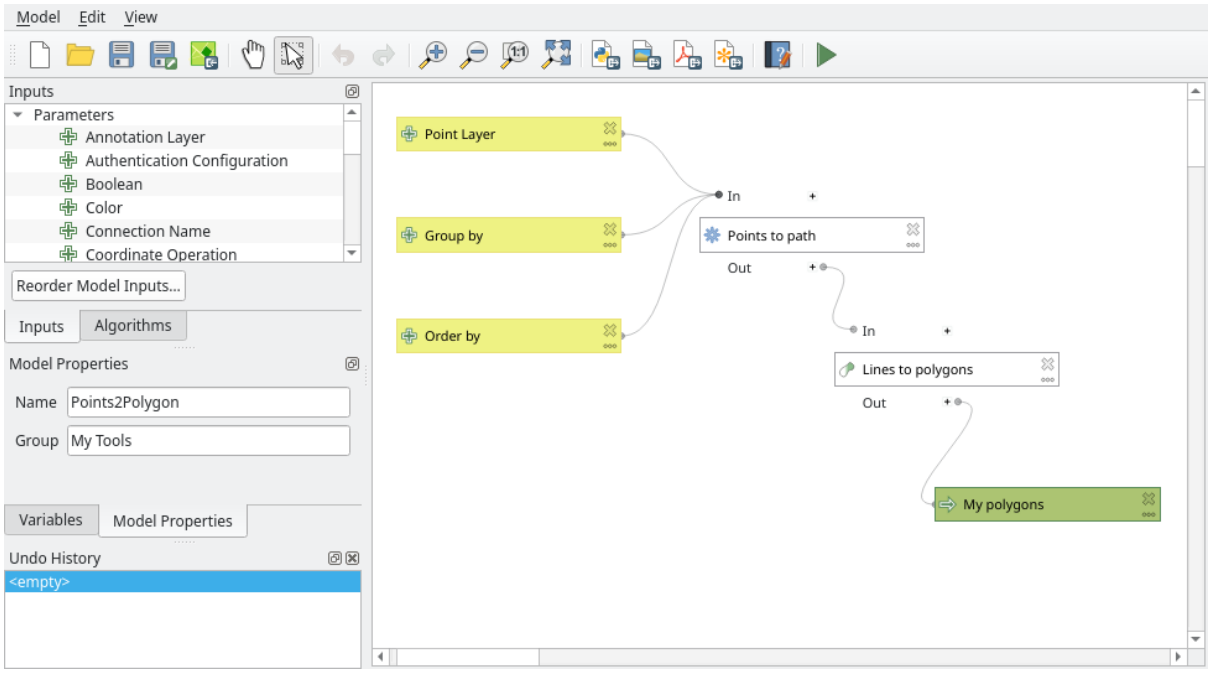


Figura 23.2: Modelador de Procesamiento

- El Admministrador *Historial*: Todas las acciones realizadas con cualquiera de los elementos antes mencionados se almacenan en un archivo de historial y luego se pueden reproducir fácilmente con el administrador de historial.

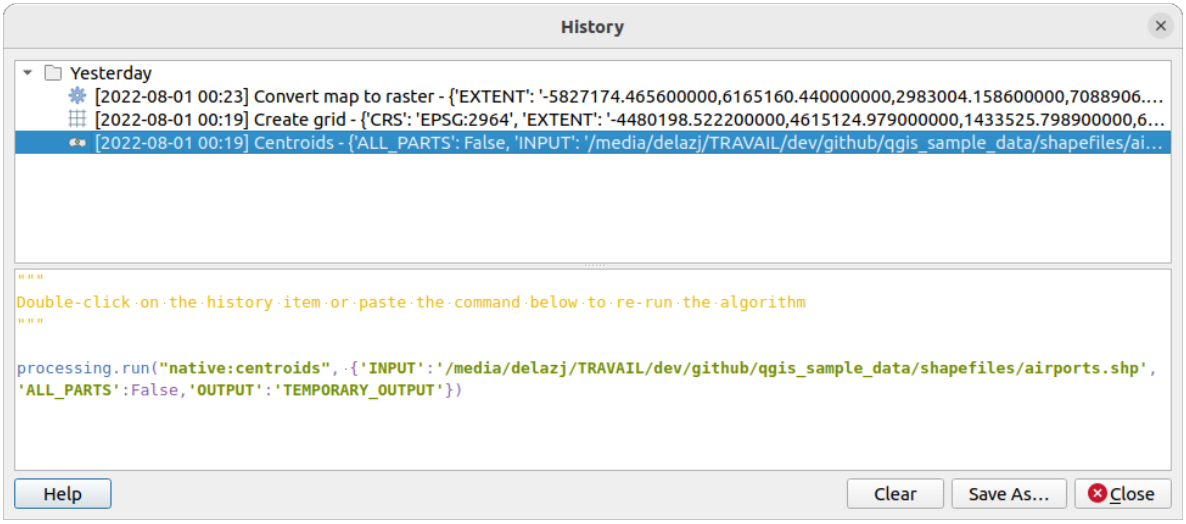


Figura 23.3: El historial del procesamiento

- La interfaz *Procesamiento por lotes* : Esta interfaz le permite ejecutar procesos por lotes y automatizar la ejecución de un solo algoritmo en múltiples conjuntos de datos.

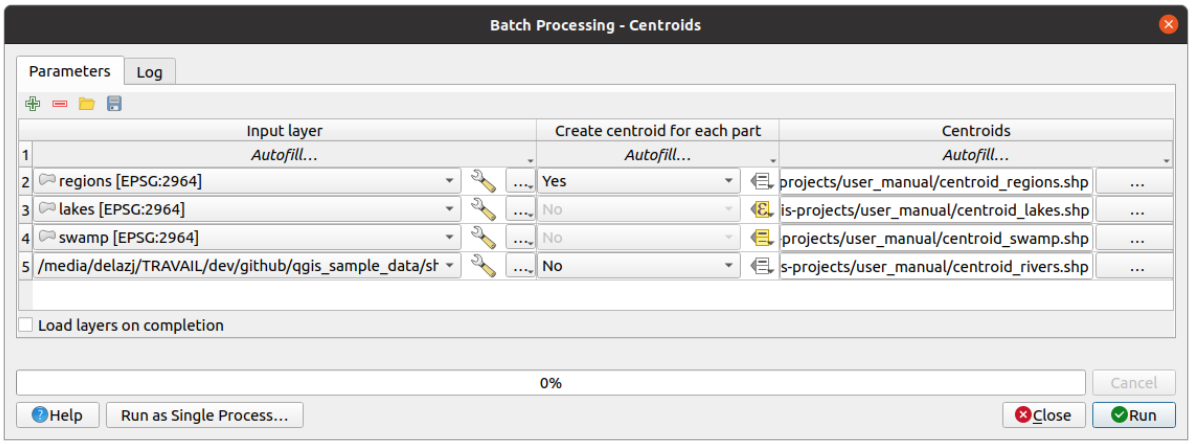


Figura 23.4: Interfaz de procesamiento por lote

En las siguientes secciones, revisaremos cada uno de los elementos a detalle.

23.2 Configurando el marco de procesamiento

El menú Opciones de Procesos (**:menuselección:`Configuración--> Opciones -->` |procesamientoAlgoritmo|** *Pestaña Procesos*) permite configurar el funcionamiento de los algoritmos. Los parámetros de configuración están estructurados en bloques separados que puede seleccionar en la parte izquierda del diálogo.

23.2.1 General

El bloque *General* contiene los ajustes predeterminados para controlar cómo debe comportarse el diálogo del algoritmo, así como los parámetros de entrada o salida. Sin embargo, algunos de los ajustes pueden anularse a nivel de algoritmo, por *ejecución del algoritmo* o para *parámetros individuales*.

- *Extensión de capa ráster saliente predeterminada* es por defecto `tif`
- *Extensión de capa vectorial saliente predeterminada* es por defecto `gpkg`
- *Invalid features filtering* al ejecutar el algoritmo:
  - *Do not filter (better performance)*: se procesan todas las características (con geometrías válidas e inválidas), pero el resultado puede ser erróneo dependiendo de cómo la invalidez de la geometría afecte las operaciones
  - *Skip (ignore) features with invalid geometries*, lo que significa que solo se procesará un subconjunto de su conjunto de datos (las características geométricas válidas)
  - *Stop algorithm execution when a geometry is invalid*: deberá rastrear y corregir las geometrías no válidas si desea que el algoritmo procese toda la capa. Algoritmos como *Comprobar validez* o *Corregir geometrías* pueden ayudarlo a lograr esto.

  La configuración de *Invalid features filtering* se puede anular por entrada, en tiempo de ejecución del algoritmo.
- *Mantener el diálogo abierto después de ejecutar el algoritmo*. Una vez que un algoritmo ha terminado de ejecutarse y sus capas de salida se cargan en el proyecto QGIS, se cierra el diálogo del algoritmo. Si desea mantenerlo abierto (para ejecutar el algoritmo nuevamente con diferentes parámetros, o para verificar mejor la salida que está escrita en la pestaña de registro), marque esta opción.
- *Hilos máximos*

- *Output folder* for non temporary outputs: If no folder path is provided for the Processing execution outputs, this is the folder in which they will be saved. Default is `processing` under the user home directory (e.g., `C:\Users\<YOURUSERNAME>\processing\` on Windows).
- *Anula la ruta de la carpeta de las salidas temporales*: Las salidas temporales son almacenadas de forma predeterminada en la carpeta `tmp` de la maquina. Esta opción le ayuda a establecer un sitio diferente de almacenamiento.
- *Script de ejecución previa* y *Script de ejecución posterior*. Estos parámetros apuntan a archivos que contienen secuencias de comandos escritas utilizando la funcionalidad de procesamiento de secuencias de comandos, que se explica en la sección que trata sobre secuencias de comandos y la consola.
- *Preferir el nombre del archivo de salida para los nombres de las capas*. El nombre de cada capa resultante creada por un algoritmo lo define el propio algoritmo. En algunos casos, se puede usar un nombre fijo, lo que significa que se usará el mismo nombre de salida, sin importar qué capa de entrada se use. En otros casos, el nombre puede depender del nombre de la capa de entrada o de algunos de los parámetros utilizados para ejecutar el algoritmo. Si esta casilla de verificación está marcada, el nombre se tomará del nombre del archivo de salida. Tenga en cuenta que, si la salida se guarda en un archivo temporal, el nombre de archivo de este archivo temporal suele ser largo y sin sentido destinado a evitar la colisión con otros nombres de archivo ya existentes.
- *Results group name*. Si desea obtener todas las capas de resultados de procesamiento en un grupo en el panel *Layers* , establezca un nombre de grupo para este parámetro. El grupo puede existir ya o no. QGIS agregará todas las capas de salida a dicho grupo. De forma predeterminada, este parámetro está vacío, por lo que todas las capas de salida se agregan a diferentes lugares en el panel *Layers*, según el elemento que esté activo cuando se ejecuta un algoritmo. Tenga en cuenta que las capas de salida se cargarán en el panel *Layers* solo si *Open output file after running algorithm* está marcado en el cuadro de diálogo del algoritmo.
- *Show algorithms with known issues*: De forma predeterminada, QGIS evita la visualización de algoritmos rotos (generalmente de proveedores externos). Si están marcados, estarán disponibles en la caja de herramientas Procesamiento, con un icono de advertencia y una información sobre herramientas que explica que tienen problemas. Úselo bajo su propio riesgo.
- *Mostrar el recuento de objetos espaciales para las capas vectoriales de salida*. Esta opción está desactivada por defecto, ya que el cálculo del recuento de objetos espaciales puede llevar algún tiempo en determinados formatos de datos.
- *Mostrar definición de SRC de la capa en recuadros de selección*
- *Mostrar información sobre herramientas cuando hay proveedores deshabilitados*
- *Estilo para capas de línea*, *Estilo para capas de puntos*, *Estilo para capas de polígonos* y *Estilo para capas ráster* se utilizan para establecer el estilo de renderizado predeterminado para las capas de salida (es decir, capas generadas por algoritmos de procesamiento). Simplemente cree el estilo que desea usando QGIS, guárdelo en un archivo y luego ingrese la ruta a ese archivo en la configuración para que los algoritmos puedan usarlo. Siempre que Processing cargue una capa y la agregue al lienzo de QGIS, se renderizará con ese estilo.

  Los estilos de representación se pueden configurar individualmente para cada algoritmo y cada una de sus salidas. Simplemente haga clic derecho en el nombre del algoritmo en la caja de herramientas y seleccione *Edit rendering styles for outputs*. Verá un cuadro de diálogo como el que se muestra a continuación.

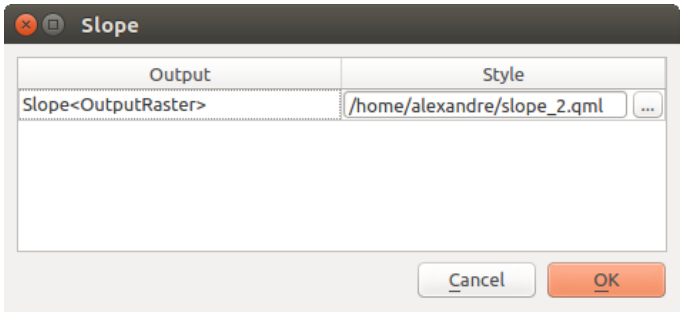


Figura 23.5: Estilos de representación

Seleccione el archivo de estilo (`.qml`) que quiera para cada salida y presione *OK*.

- *Advertir antes de ejecutar si los SRC de los parámetros no coinciden*: De forma predeterminada, los algoritmos nativos de QGIS (es decir, los listados en el grupo *Menús* ► *QGIS (C++ nativo)*) reproyectan de forma transparente las capas de entrada al SRC de la primera antes de la ejecución. Marque esta opción para obtener una notificación de las otras herramientas que no soportan la reproyección, cuando los SRC de entrada no son idénticos. Los proveedores de terceros no se ven afectados.

23.2.2 Menús

barra de hEl bloque *Menús* controla si un algoritmo, script o modelo (incorporado o proporcionado por complementos) debe estar disponible a través de un menú dedicado o una barra de herramientas (junto con la Caja de herramientas de procesamiento). Para cada elemento de cada proveedor, usted puede:

- *Add button in toolbar*, haciéndolo disponible en la barra de herramientas *Processing Algorithms*
- Asignar un *Icon* al algoritmo
- establezca *Ruta del menú*: el algoritmo estará entonces disponible a través de un menú existente o personalizado, por ejemplo, `Vect&or/Mis mejores algoritmos` añadirá un acceso directo en un submenú *Mis mejores algoritmos* del menú predeterminado *Vector*.

Reinicie QGIS para aplicar la configuración. En cualquier momento, sus cambios pueden ser *Reset to defaults*.

23.2.3 Modelos y scripts

En los bloques *Modelos* y *Scripts*, se utiliza una ruta predeterminada, establecida en la carpeta *Perfil de usuario* , se proporciona para almacenar modelos y scripts, respectivamente. En cuanto a las demás opciones, puede modificar la ruta, por ejemplo, para que conduzca a una carpeta remota o compartida para todos sus usuarios.

23.2.4 Proveedores

También encontrará un bloque para el algoritmo *Proveedores*. Aquí es donde los proveedores instalados muestran su configuración.

De forma predeterminada, QGIS se instala con dos proveedores de algoritmos:

- *GDAL* cuyos algoritmos puede *activar* (o no) en este cuadro de diálogo. Más información sobre los *algoritmos GDAL*.
- *GRASS* cuyos algoritmos requieren la instalación del *Proveedor de procesamiento GRASS GIS* desde el *Administrador de complementos*. A continuación, puede establecer algunos ajustes específicos, tales como:
  - **|casilla de verificación|** *Para capas raster, utilice r.external (más rápido) en lugar de r.in.gdal*
  - **|casilla de verificación|** *Para capas vectoriales, utilice v.external (más rápido) en lugar de v.in.ogr*
  - *Ubicación de los documentos GRASS*: por defecto, y dependiendo de su sistema operativo, QGIS utiliza una lista de rutas codificadas para encontrar la documentación local de GRASS y abrirla cada vez que pulsa el botón *Ayuda* del cuadro de diálogo de un algoritmo GRASS. Al rellenar esta opción se anulará la ruta predeterminada, lo que le permitirá utilizar una ubicación personalizada o documentación remota.
  - *Registrar salida de consola*
  - *Registrar comandos de ejecución*

Los complementos instalados que proporcionan algoritmos de procesamiento también pueden tener su proveedor incluido en este grupo, con ajustes personalizados.

23.3 La caja de Herramientas

La *Caja de Herramientas de Procesamiento* es el elemento principal de la GUI de procesamiento y el que es más probable que utilice en su trabajo diario. Muestra la lista de todos los **algoritmos** disponibles agrupados en diferentes bloques llamados *Proveedores*, y **modelos** y **scripts** personalizados que puede agregar para ampliar el conjunto de herramientas. Por lo tanto, la caja de herramientas es el punto de acceso para ejecutarlos, ya sea como un proceso único o como un proceso por lotes que implica varias ejecuciones del mismo algoritmo en diferentes conjuntos de entradas.

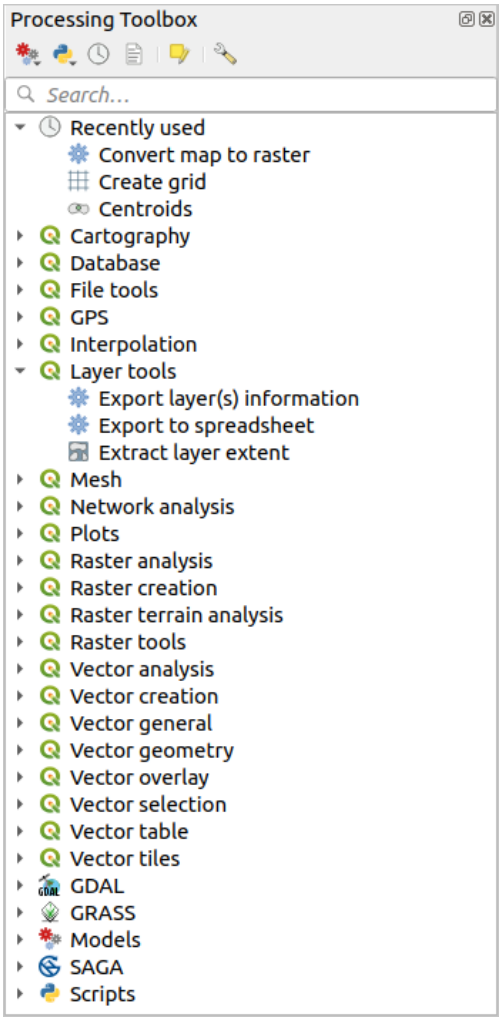


Figura 23.6: Caja de Herramientas de Procesamiento

Los proveedores se pueden (des)activar en *Diálogo de configuración de procesamiento*. De forma predeterminada, solo los proveedores que no dependen de aplicaciones de terceros (es decir, aquellos que solo requieren que se ejecuten elementos QGIS) están activos. Los algoritmos que requieren aplicaciones externas pueden necesitar una configuración adicional. La configuración de proveedores se explica en un *capítulo posterior* de este manual.

En la parte superior del cuadro de diálogo de la caja de herramientas, encontrará un conjunto de herramientas para:

- Trabajar con Modelos: *Crear Nuevo Modelo…*, *Abrir Modelo existente…* y *Agregar Modelo a la caja de Herramientas…*;
- trabajar con Scripts: *Craer Nuevo Script…*, *Crear Nuevo Script desde Plantilla…*, *Abrir Script existente…* y *Agregar Script a Caja de Herramientas…*;
- Abra el panel Historial;

- abra el panel  Visor de Resultados;
- Conmutar la caja de herramientas a *modo de modificación in situ* usando el botón  Editar objetos de la capa activa : solo se muestran los algoritmos que son adecuados para ejecutarse en la capa activa sin generar una nueva capa;
- abra el diálogo  Opciones.

Debajo de esta barra de herramientas hay un cuadro  *Buscar…* para ayudarle a encontrar fácilmente las herramientas que necesita. Puede ingresar cualquier palabra o frase en el cuadro de texto. Observe que, a medida que escribe, la cantidad de algoritmos, modelos o scripts en la caja de herramientas se reduce a solo aquellos que contienen el texto que ha ingresado en sus nombres o palabras clave.

La parte superior de la lista de algoritmos muestra las herramientas más utilizadas  Recientemente utilizadas y las herramientas  Favoritas, lo que resulta muy práctico para volver a ejecutarlas. La lista  Recientemente utilizadas aparece por defecto y muestra los algoritmos que ha ejecutado recientemente. La lista  Favoritas se crea cuando se añade manualmente un algoritmo a ella.

Puede hacer clic con el botón derecho del ratón en cualquier algoritmo de la caja de herramientas y seleccionar en el menú contextual:

- *Ejecutar…* para ejecutar el algoritmo.
- *Ejecutar como proceso por lotes…* para ejecutar el algoritmo en modo por lotes.
- *Editar estilos de representación para salidas…* para personalizar los estilos de representación de las salidas del algoritmo.
- *Añadir a favoritos* para añadir el algoritmo a la lista de favoritos.
- *Eliminar de favoritos* para eliminar el algoritmo de la lista de favoritos.

A diferencia de la lista Recientemente utilizados, la lista Favoritos permanece estática y no se ve afectada por los algoritmos que se hayan ejecutado.

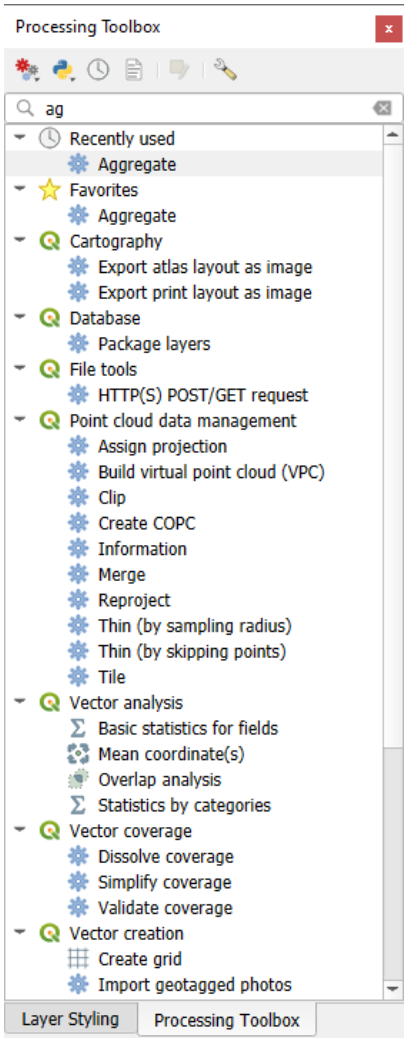


Figura 23.7: Caja de herramientas de procesamiento que muestra los resultados de la búsqueda

Para ejecutar una herramienta, simplemente haga doble clic en su nombre en la caja de herramientas.

23.3.1 El cuadro de diálogo de algoritmo

Una vez que haga doble clic sobre el nombre del algoritmo que desea ejecutar, se mostrará un diálogo similar al del Figura 23.8 que aparece a continuación (en este caso, el diálogo corresponde al algoritmo `Centroides`). El título del diálogo incluirá el nombre del grupo del que procede el algoritmo.

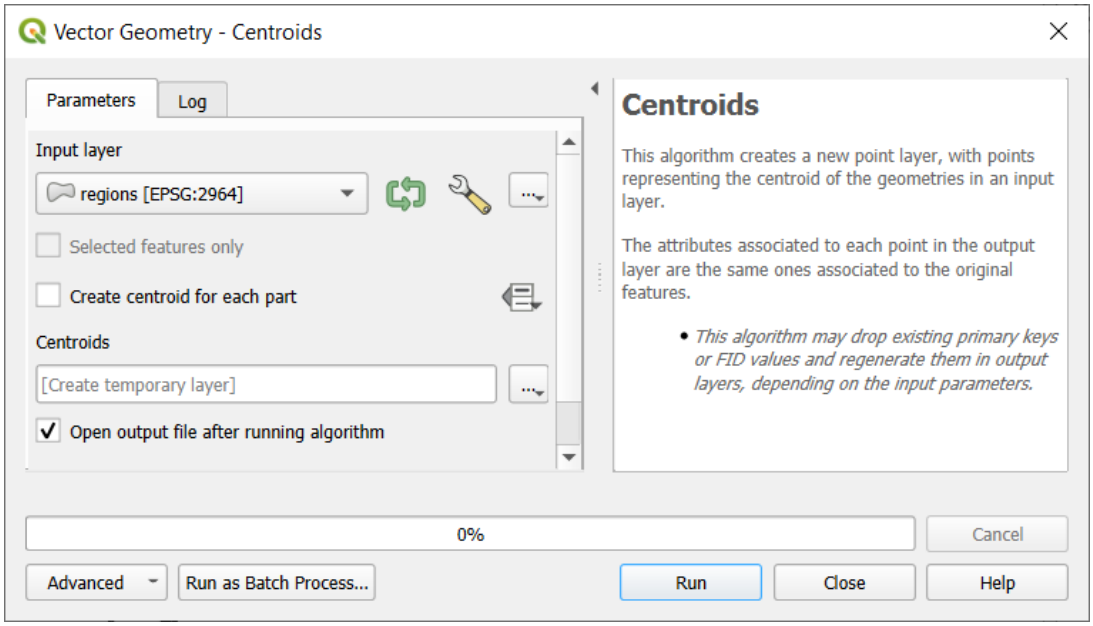


Figura 23.8: Dialogo Algoritmo - Parámetros

El diálogo muestra dos pestañas (*Parameters* y *Log*) en la parte izquierda, la descripción del algoritmo a la derecha y un conjunto de botones en la parte inferior.

Tipos de parámetros

La pestaña *Parámetros* se usa para establecer los valores de entrada que el algoritmo necesita para ser ejecutado. Muestra una lista de valores de entrada y parámetros de configuración a configurar. Por supuesto, tiene un contenido diferente, dependiendo de los requisitos del algoritmo a ejecutar, y se crea automáticamente en función de esos requisitos.

Truco

**Establecer sus propios valores por defecto para los parámetros del algoritmo.

Los diálogos de los algoritmos se abren con algunos parámetros prefijados con los valores de la instalación de QGIS. Sin embargo, es posible establecer *sus propios valores por defecto* para parámetros específicos del algoritmo, de modo que se utilicen al iniciarse el algoritmo.

Aunque el número y el tipo de parámetro dependen de las características del algoritmo, la estructura es similar para todos ellos. Los parámetros encontrados en la tabla pueden ser de uno de los siguientes tipos.

- Una **capa vectorial**, para seleccionar de una lista de todas las capas vectoriales disponibles (actualmente abiertas) en QGIS. También puede utilizar capas no cargadas: pulse el botón … en la parte derecha del widget, y seleccione:
  - *Seleccionar archivo…*: selecciona un archivo del disco utilizando el explorador de archivos del sistema operativo
  - *Buscar capa…*: abre el *panel Navegador*, que permite tomar las capas directamente de fuentes de bases de datos (PostgreSQL, SQL Server, Oracle, …), servicios web (WFS, AFS, …) o archivos en disco.



Figura 23.9: Widget de entrada de vectores

Truco

Drag-and drop as convenient alternative for layer selection

Besides selecting from the drop-down menu or browsing sources, you can also drag and drop a layer from the *Layers* or *Browser* panel, or from the Operating System file explorer directly into the combobox.

Nota

De forma predeterminada, el widget de capa muestra el SRC de la capa junto con su nombre. Si no desea ver esta información adicional, puede desactivar esta funcionalidad en el cuadro de diálogo Configuración de procesamiento, desmarcando la opción *General* ► *Mostrar definición de SRC de capa en cuadros de selección*.

El widget de entrada de vectores también tiene las siguientes características:

- un botón iterador : Si está activado, el algoritmo se ejecutará iterativamente en cada una de sus objetos espaciales, en lugar de una sola vez para toda la capa, produciendo tantas salidas como veces se ejecute el algoritmo. Esto permite automatizar el proceso cuando todas las características de una capa deben procesarse por separado. Si el algoritmo contiene varios vectores de entrada sobre los que se puede iterar, la iteración se procesará sólo sobre el primer parámetro conmutado, en el orden en que se declaran los parámetros en el algoritmo.
- Opciones avanzadas para ajustar la configuración a utilizar para ese parámetro específico. Estos ajustes se refieren a:
  - *Filtrado de objetos espaciales no válidos*: permite el *método predeterminado* para el tratamiento de objetos espaciales con geometrías no válidas que deben anularse
  - *Límite de objetos espaciales procesados*: límite opcional del número de objetos espaciales procesados desde el origen.
  - *Filtro de objeto espacial*: permite introducir una expresión para subconjuntar la capa dinámicamente al ejecutar la herramienta, evitando la necesidad de pasos separados para establecer filtros de capa o crear subconjuntos de capa.

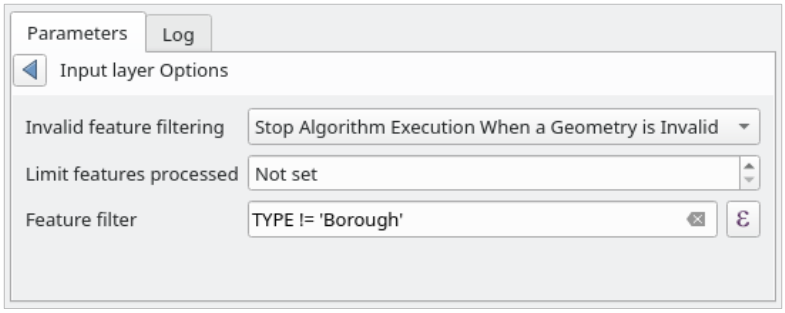


Figura 23.10: Opciones avanzadas para el widget de entrada vectorial

- También es posible limitar la ejecución del algoritmo en la capa vectorial a su *Sólo objetos espaciales*

seleccionados.

- Una **tabla**, para seleccionar de una lista de todas las disponibles en QGIS. Las tablas no espaciales se cargan en QGIS como capas vectoriales, y utilizan el *mismo widget*.
- Una **capa ráster**, para seleccionar de una lista de todas las capas ráster disponibles en QGIS. El selector contiene también un botón … en su parte derecha, para permitirle seleccionar nombres de archivos que representan capas actualmente no cargadas en QGIS.

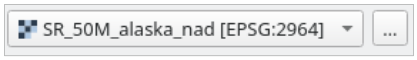


Figura 23.11: Widget de entrada Raster

- Una **opción**, para elegir de una lista de selección de posibles opciones.
- Un **valor numérico**, que se introducirá en una casilla de número. En algunos contextos (cuando el parámetro se aplica a nivel de entidad y no a nivel de capa), encontrará un botón Anulación definida por datos a un lado, que le permite abrir el :ref:‘constructor de expresiones<vector_expressions>` e ingresar una expresión matemática para generar valores variables para el parámetro. Algunas variables útiles relacionadas con los datos cargados en QGIS se pueden agregar a su expresión, por lo que puede seleccionar un valor derivado de cualquiera de estas variables, como el tamaño de celda de una capa o la coordenada más al norte de otra.

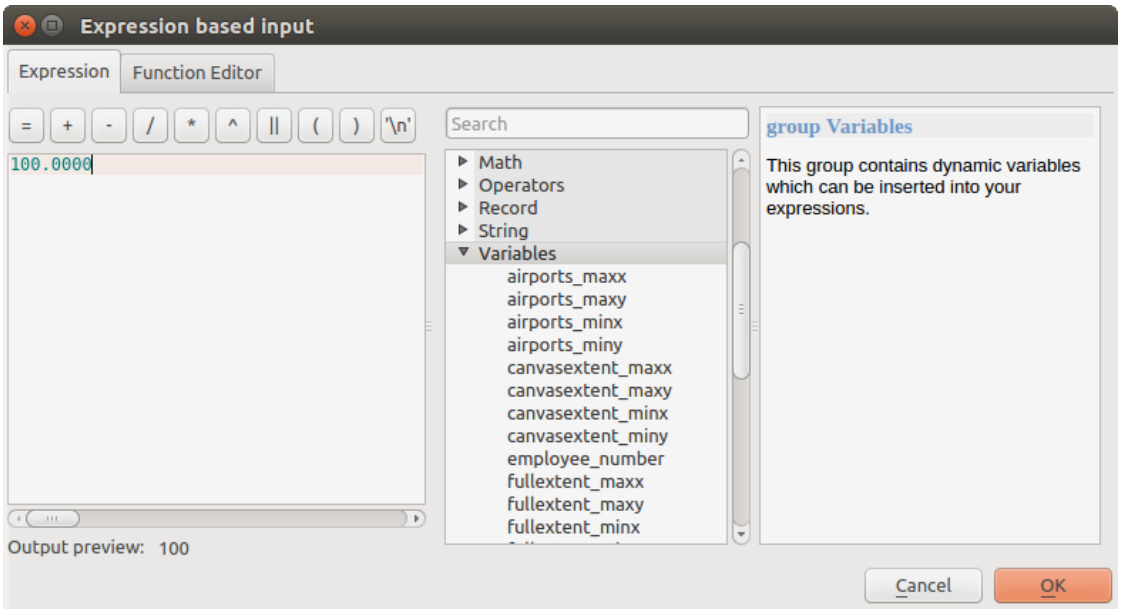


Figura 23.12: Entrada basada en expresión

- Un **rango**, con valores mínimos y máximos que se introducirán en dos cuadros de texto.
- Una **cadena de texto**, que se introducirá en un cuadro de texto.
- Un **campo**, para elegir de la tabla de atributos de una capa vectorial o de una sola tabla seleccionada en otro parámetro.
- Un **sistema de referencia de coordenadas**. Puede seleccionarlo entre los usados recientemente en la lista desplegable o en el diálogo *Selección de SRC* que aparece al hacer clic en el botón del lado derecho.

> **Truco**
>
> **Drag-and-drop as convenient shortcut for CRS selection**

You can also drag a layer from the Layers panel and drop it onto the CRS widget. The widget will automatically set its CRS to match the dropped layer's.

- Una **extensión**, un cuadro de texto que define un rectángulo a través de sus esquinas coordinadas en el formato `xmin, xmax, ymin, ymax`. Pulse el botón Establecer la extensión actual del lienzo del mapa para utilizar la extensión del lienzo del mapa. Al hacer clic en la flecha situada a la derecha del selector de valores, aparecerá un menú emergente con opciones para:
  - *Calcular a partir de capa* ►: rellena la caja de texto con las coordenadas del cuadro delimitador de una capa a seleccionar entre las cargadas.
  - *Calcular a partir de mapa de disposición* ►: rellena la caja de texto con las coordenadas de un elemento de mapa seleccionado de una disposición del proyecto actual.
  - *Calcular a partir de marcador* ►: rellena la caja de texto con las coordenadas de un marcador guardado
  - *Utilizar la extensión actual del lienzo del mapa*
  - *Dibujar sobre lienzo*: la ventana de parámetros se ocultará, por lo que puede hacer click y arrastrar sobre el lienzo. Una vez que haya definido el rectángulo de extensión, volverá a aparecer el cuadro de diálogo que contiene los valores en el cuadro de texto de la extensión.

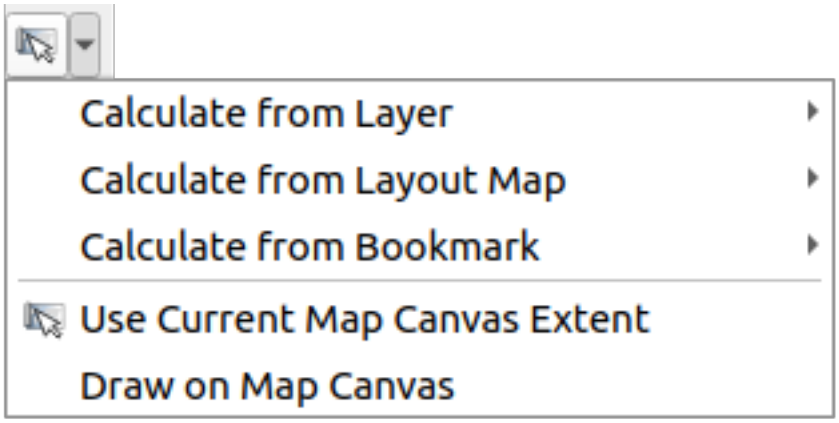


Figura 23.13: Selector de extensión

- Una **lista de elementos** (ya sean capas raster o vectoriales, tablas, campos) para seleccionar. Haga clic en el botón … a la izquierda de la opción para ver un diálogo como el siguiente. Se permite la selección múltiple y cuando se cierra el cuadro de diálogo, el número de elementos seleccionados se muestra en el widget del cuadro de texto del parámetro.

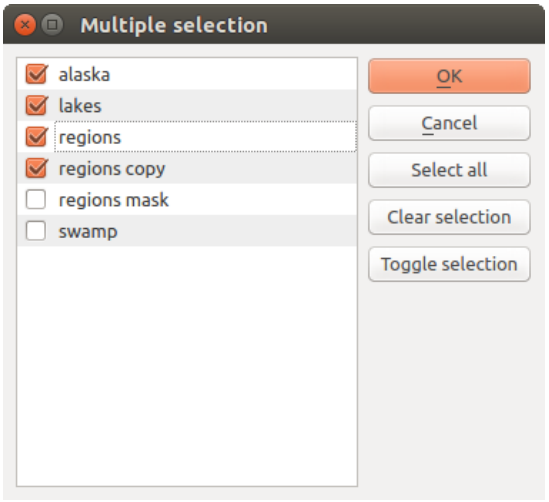


Figura 23.14: Selección Múltiple

- Una **tabla pequeña** para ser editada por el usuario. Estos se utilizan para definir parámetros como tablas de búsqueda o núcleos de convolución, entre otros.

  Click en el botón del lado derecho para ver la tabla y editar sus valores.

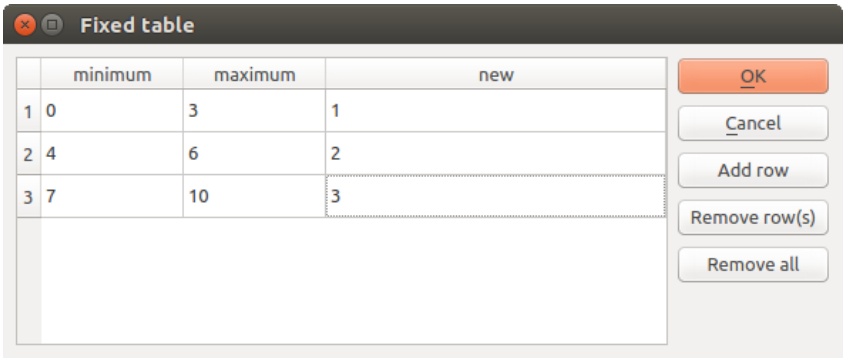


Figura 23.15: Tabla Fija

Dependiendo del algoritmo, el número de filas que pueden ser modificadas o no al utilizar los botones del lado derecho de la ventana.

Nota

Algunos algoritmos requieren muchos parámetros para ejecutarse, por ejemplo, en el *Calculadora ráster* hay que especificar manualmente el tamaño de la celda, la extensión y el CRS. Puede evitar elegir todos los parámetros manualmente cuando el algoritmo tiene el parámetro `Capas de referencia`. Con este parámetro puede elegir la capa de referencia y se utilizarán todas sus propiedades (tamaño de celda, extensión, SRC).

Registro de la ejecución

Junto con la pestaña *Parámetros*, hay otra pestaña llamada *Registro* (véase Figura 23.16 más abajo). La información proporcionada por el algoritmo durante su ejecución se escribe en esta pestaña, lo que le permite realizar un seguimiento de la ejecución, así como conocer y obtener más detalles sobre el algoritmo mientras se ejecuta. Puede hacer clic directamente en los nombres de los archivos de salida, carpetas o archivos HTML que aparecen en el registro. Al hacerlo, se abrirá la carpeta que contiene el archivo generado y se seleccionará automáticamente. La información sobre la ejecución del algoritmo también se muestra en *Ver* ► *Paneles* ► *Panel de mensajes de registro*.

Tenga en cuenta que no todos los algoritmos escriben información en la pestaña *Registro*, y muchos de ellos pueden ejecutarse silenciosamente sin producir ningún resultado más que los archivos finales. Compruebe el *Panel de registro de mensajes* en ese caso.

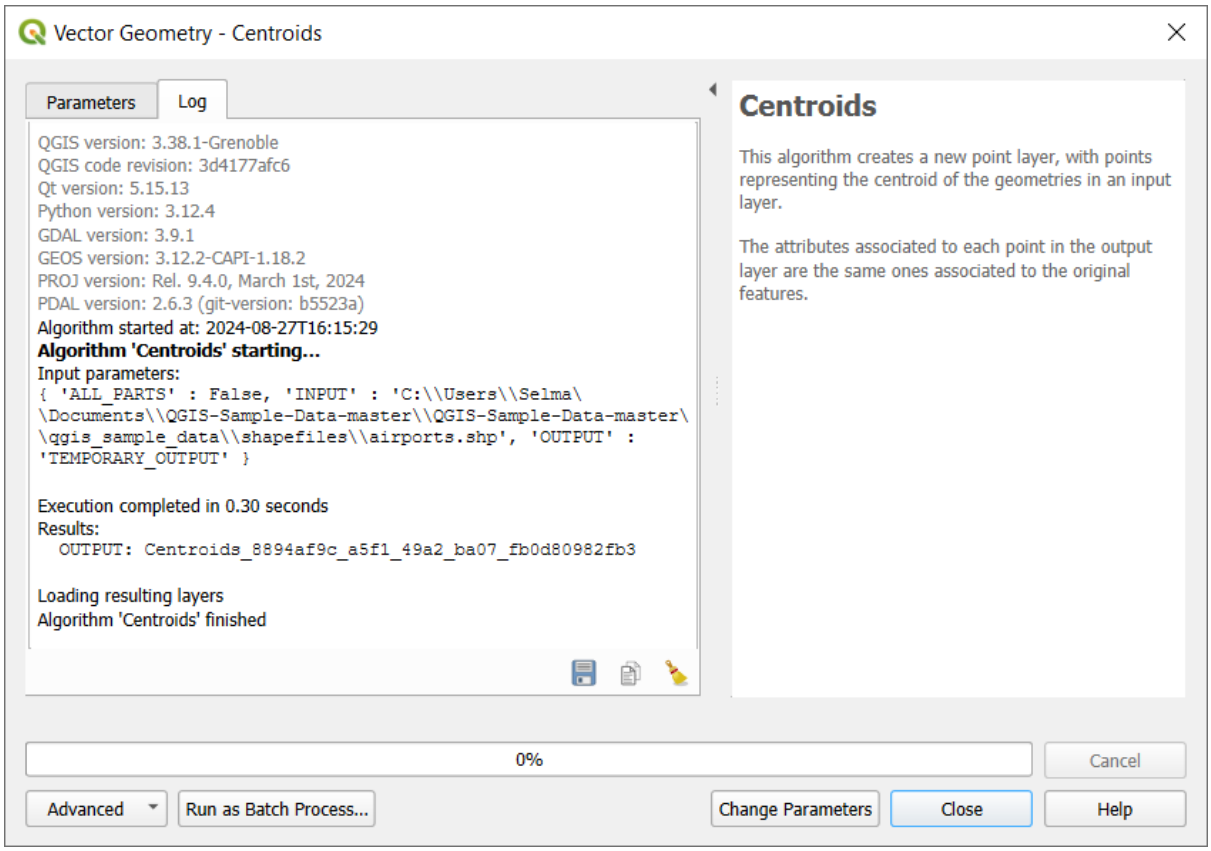


Figura 23.16: Diálogo Algoritmo - Registro

En la parte inferior de la pestaña *Registro* encontrará botones para *Guardar registro en archivo*, *Copiar registro al portapapeles* y *Borrar registro*. Estos son particularmente útiles cuando ha marcado *Mantener el diálogo abierto después de ejecutar el algoritmo* en la parte *General* de las opciones de Procesamiento.

Otras herramientas

En el lado derecho del cuadro de diálogo encontrará una breve descripción del algoritmo, que lo ayudará a comprender su propósito y sus ideas básicas. Si dicha descripción no está disponible, no se mostrará el panel de descripción.

Para obtener un archivo de ayuda más detallado, que puede incluir una descripción de cada parámetro que utiliza, o ejemplos, encontrará un botón *Ayuda* en la parte inferior del diálogo que lo lleva a la :ref:`Documentación de algoritmos de procesamiento<processing_algs>`o a la documentación del proveedor (para algunos proveedores externos).

El menú *Avanzado* ► proporciona funciones para reutilizar la configuración definida en el diálogo sin ejecutar el algoritmo:

- *Configuración del algoritmo…*: permite anular la configuración de procesamiento para la ejecución actual del algoritmo. Más detalles en *Anular la configuración del algoritmo*.
- *Copiar como comando Python*: permite copiar fácilmente el *comando PyQGIS* equivalente para ejecutar la herramienta utilizando los parámetros definidos en el diálogo
- *Copiar como comando qgis_process*: permite generar fácilmente el *comando qgis_process*, incluida la configuración del entorno, como las unidades de distancia, las unidades de superficie, el elipsoide y cualquier parámetro complicado, como los resultados de GeoPackage con capas específicas.
- *Copiar como JSON*: todos los parámetros del comando se copian en formato `JSON`, listos para ser consumidos por qgis_process. Esta es una forma conveniente de ver el formato esperado de los comandos, incluso para parámetros complejos (como los parámetros de interpolación TIN). Puede almacenarlos fácilmente y restaurarlos más tarde pegando los valores.
- *Pegar configuración* en formato `JSON`

El botón *Ejecutar como proceso por lotes…* activa el *modo de procesamiento por lotes* permitiendo configurar y ejecutar múltiples instancias del algoritmo con una variedad de parámetros. El botón *Ejecutar como proceso único…* le ayuda a volver del modo por lotes.

Cuando finaliza la ejecución de un algoritmo (con o sin éxito), se muestra un nuevo botón *Cambiar parámetros* siempre que la pestaña *Registro* esté activa.

Anular la configuración del algoritmo

Activado desde el menú desplegable *Avanzado* en la parte inferior del cuadro de diálogo de un algoritmo, *Configuración del algoritmo…* muestra un panel que permite a los usuarios controlar la configuración general de procesamiento que se aplica sólo a la ejecución de ese algoritmo. Se pretende que sea un lugar donde el usuario pueda anular sus *configuraciones globales de procesamiento* de forma ad hoc sin tener que cambiar su configuración habitual por omisión.

Los ajustes que se pueden anular son:

- *Filtrado no válido de objetos espaciales*: a diferencia del ajuste por parámetro existente para esto, establecer el método de tratamiento aquí se aplicará a **TODAS las entradas** para el algoritmo.
- *Configuraciones de cálculo*, como *Unidades de distancia* y *Unidades de área* que se utilizarán para las mediciones de distancia/área.
- *Configuración del entorno*, como *Carpeta temporal* y *Número de hilos a utilizar*.

Un comentario sobre proyecciones

La ejecución del algoritmo de procesamiento siempre se realiza en el sistema de referencia de coordenadas de la capa de entrada (SRC). Debido a las capacidades de reproyección sobre la marcha de QGIS, aunque parezca que dos capas se superponen y coinciden, eso podría no ser cierto si se utilizan sus coordenadas originales sin reproyectarlas en un sistema de coordenadas común. Siempre que utilice más de una capa como entrada para un *algoritmo nativo de QGIS*, ya sea vectorial o ráster, todas las capas se volverán a proyectar para que coincidan con el sistema de referencia de coordenadas de la primera capa de entrada.

Sin embargo, esto es menos cierto para la mayoría de las aplicaciones externas cuyos algoritmos se exponen a través del marco de procesamiento, ya que asumen que todas las capas ya están en un sistema de coordenadas común y listas para ser analizadas.

De forma predeterminada, el cuadro de diálogo de parámetros mostrará una descripción del SRC de cada capa junto con su nombre, lo que facilita la selección de capas que comparten el mismo SRC para utilizarlas como capas de entrada. Si no desea ver esta información adicional, puede deshabilitar esta funcionalidad en el cuadro de diálogo Configuración de procesamiento, desmarcando la opción:guilabel:*Mostrar definición de capa SRC en cuadros de selección*.

Si intenta ejecutar un algoritmo utilizando como entrada dos o más capas con CRS no coincidentes, se mostrará un cuadro de diálogo de advertencia. Esto ocurre gracias a la opción *Advertir antes de ejecutar si la capa SRC no coincide*.

Aún se puede ejecutar el algoritmo, pero en la mayoría de los casos se producirán resultados incorrectos, como capas vacias debido a que las capas de entrada no se solapan.

> **Truco**
>
> **Usar algoritmos de procesamiento para realizar una reproyección intermedia**
>
> Cuando un algoritmo no puede funcionar con éxito en múltiples capas de entrada debido a SRC no coincidentes, use el algoritmo interno de QGIS como *Capa reproyectada* para realizar la reproyección de capas al mismo SRC antes de ejecutar el algoritmo usando estas salidas.

23.3.2 Resultados generados por algoritmos

Los objetos de datos generados por un algoritmo pueden ser de diferentes tipos. Pueden ser una capa (vector con o sin geometría, ráster, malla, nube de puntos, etc.), almacenados como un archivo sin formato en el disco o en una base de datos, archivos de texto como `qml` archivo de estilo o gráficos `.html` archivo, una carpeta, etc.

The parameters dialog will contain a widget corresponding to the needed output, where you can type the output channel to use for saving it. An output channel contains the information needed to save the resulting object somewhere. In the most usual case, you will save it to a file, but in the case of vector layers, and when they are generated by native algorithms (algorithms not using external applications) you can also save to a PostgreSQL, GeoPackage or SpatiaLite database, or a memory layer.

El parámetro **salida** para configurar el resultado de la ejecución del algoritmo proporciona un cuadro de texto para rellenar la ruta al resultado de salida y un botón desplegable … para opciones adicionales (la disponibilidad depende del algoritmo que ejecute):

- *Skip Output*: if you are not interested in a given output of the algorithm
- *Crear capa temporal* (`TEMPORARY_OUTPUT`): la salida se almacena en un vector *capa temporal de trabajo*.
- *Guardar en archivo…*: se le mostrará un cuadro de diálogo para guardar el archivo, donde podrá seleccionar la ruta de archivo deseada. Las extensiones de archivo compatibles se muestran en el selector de formato de archivo del cuadro de diálogo, dependiendo del tipo de salida y del algoritmo.

  El formato de la salida está definido por la extensión del nombre de archivo. Los formatos admitidos dependen de lo que admita el propio algoritmo. Para seleccionar un formato, simplemente seleccione la extensión de archivo correspondiente (o agréguela, si en su lugar está escribiendo directamente la ruta del archivo). Si la extensión de la ruta del archivo que ingresó no coincide con ninguno de los formatos admitidos, se agregará una

extensión predeterminada a la ruta del archivo y se usará el formato de archivo correspondiente a esa extensión para guardar la capa o tabla. Las extensiones predeterminadas son `.dbf` para tablas, `.tif` para capas ráster y `.gpkg` para capas vectoriales. Estos se pueden modificar en el cuadro de diálogo de configuración, seleccionando cualquier otro de los formatos compatibles con QGIS.

Puede establecer una carpeta predeterminada para los objetos de datos de salida. Vaya al diálogo de configuración (puede abrirlo desde el menú *Configuración -> Opciones -> Procesamiento*), y en el grupo *General*, encontrará un parámetro llamado:guilabel:` Carpeta de salida`. Esta carpeta de salida se utiliza como ruta predeterminada en caso de que escriba solo un nombre de archivo sin ruta (es decir, `myfile.shp`) al ejecutar un algoritmo.

- *Save to a Temporary File*: if you do not enter any filename in the output text box (or select the corresponding option in the context menu), the result will be saved as a temporary file in the corresponding default file format, and it will be deleted once you exit QGIS (take care with that, in case you save your project and it contains temporary layers). The file is saved in the default `TEMP` folder on the machine, if *not overridden*.
- *Guardar en Geopackage…*: la capa de salida se guardará como una tabla con nombre en una base de datos GeoPackage nueva o existente.
- *Guardar en tabla de base de datos…*: la capa de salida se guardará como una nueva tabla en una base de datos como GeoPackage, SpatiaLite, PostgreSQL o MS SQL Server. Ya debe existir una conexión. Deberá proporcionar el nombre de la nueva tabla y, dependiendo del tipo de base de datos de destino, seleccionar un esquema.
- *Añadir a capa…*: añade las características generadas por el algoritmo a una capa existente. Al seleccionar la capa de destino, se abre un panel de asignación de campos que permite configurar manualmente cómo se asignan los campos de la capa de salida a los campos de la capa de destino.

- *Guardar en directorio*: Utilice un directorio nuevo o existente para almacenar los resultados del algoritmo.
- *Guardar en un directorio temporal*: crea un directorio dentro de la carpeta `TEMP` establecida para almacenar los resultados del algoritmo. Los archivos se eliminarán una vez que salga de QGIS.

Al ejecutar un algoritmo que utiliza una capa vectorial en modo iterativo, la ruta del archivo introducido se utiliza como la ruta de la base para todos los archivos generados, los cuales se denominan utilizando el nombre base y añadiendo un número que representa el índice de la iteración. La extensión del archivo (y el formato) se utiliza para todos los archivos generados.

Los algoritmos también generan gráficos y texto como archivos HTML. Estos resultados se muestran al final de la ejecución del algoritmo en el cuadro de diálogo *Visor de resultados*. Este cuadro de diálogo conservará los resultados producidos por cualquier algoritmo durante la sesión actual y se puede mostrar en cualquier momento seleccionando *Procesamiento* ► *Visor de resultados* en el menú principal de QGIS.

Para las salidas basadas en capas, es posible marcar la casilla *Abrir archivo de salida después de ejecutar el algoritmo* directamente en QGIS, como una nueva entrada en el panel *Capas*. Por defecto, los archivos se abren.

23.4 El administrador del historial

23.4.1 El historial del procesamiento

Cada vez que ejecuta un algoritmo, la información sobre el proceso se almacena en el administrador de historial. Se guarda la fecha y hora de ejecución, junto con los parámetros utilizados, lo que facilita el seguimiento y control de todo el trabajo que se ha desarrollado utilizando el marco de Procesamiento, y su reproducción.

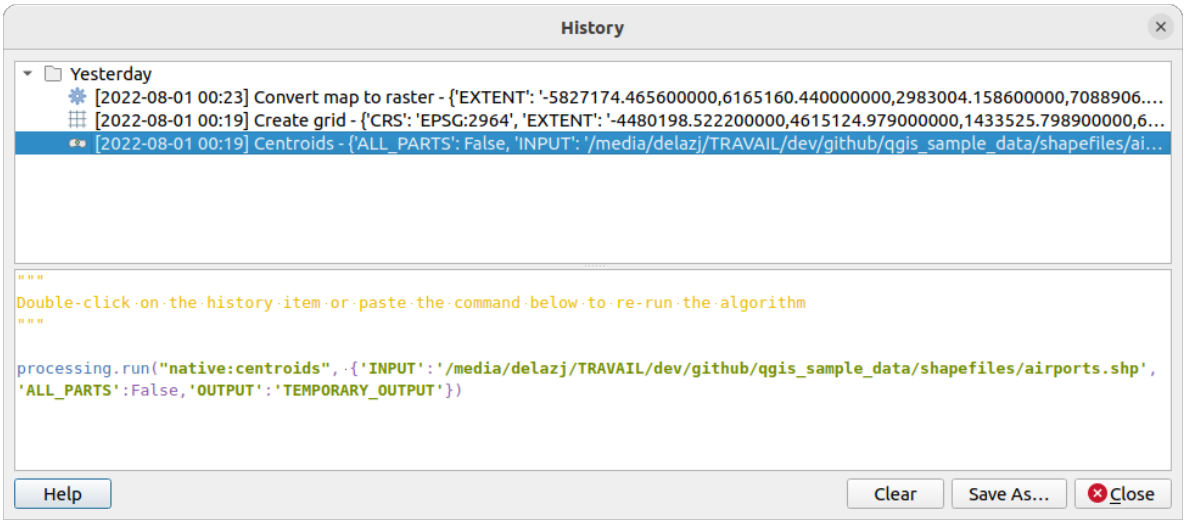


Figura 23.17: Historial

La información del proceso se mantiene como una expresión de línea de comandos, incluso si el algoritmo se inició desde la caja de herramientas. Esto lo hace útil para aquellos que están aprendiendo a usar la interfaz de línea de comandos, ya que pueden llamar a un algoritmo usando la caja de herramientas y luego consultar el administrador de historial para ver cómo se puede llamar desde la línea de comandos.

Botón derecho sobre una fila y debería ser capaz de:

- *Copiar como Comando Python*: le permite la copia fácil del comando *PyQGIS equivalente* corre sobre el dialogo. Lo mismo que el código mostrado debajo de la lista de comandos.
- *Copiar como Comando qgis_process*: permite la generación fácil de comandos *qgis_process*, incluyendo sus ajustes de entorno como las unidades de distancia, unidades de áreas, elipsoide y cualquier valor de parámetro complicado como salidas GeoPackage con las capas específicas
- *Copiar como JSON*: todos los parámetros del comando se copian en formato `JSON`, listos para ser consumidos por qgis_process. Esta es una forma conveniente de ver el formato esperado de los comandos, incluso para parámetros complejos (como los parámetros de interpolación TIN). Puede almacenar estos fácilmente y luego restaurarlos más tarde pegando los valores sobre un diálogo de algoritmo.
- *Crear prueba…* utilizando el algoritmo y los parámetros correspondientes, siguiendo las instrucciones del archivo Processing README.

Además de navegar por las entradas en el registro, también puede volver a ejecutar los procesos simplemente haciendo doble clic en la entrada. A continuación, el cuadro de diálogo del algoritmo se abre con los parámetros ya configurados y puede cambiar cualquiera de ellos para que se ajuste a sus necesidades y volver a ejecutar el algoritmo.

El diálogo *Historia* también proporciona una forma cómoda de contribuir a la consolidación de la infraestructura de pruebas de algoritmos y scripts de Procesamiento de QGIS.

23.4.2 El registro de procesamiento

El diálogo de historial solo contiene las llamadas de ejecución, pero no la información producida por el algoritmo cuando se ejecuta. Esa información se escribe en el registro de QGIS (*Ver -> Paneles -> Panel de mensajes de registro*).

Los algoritmos de terceros generalmente se ejecutan utilizando sus interfaces de línea de comandos, que se comunican con el usuario a través de la consola. Aunque esa consola no se muestra, normalmente se escribe un volcado completo en el registro cada vez que ejecuta uno de esos algoritmos. Para evitar saturar el registro con esa información, puede desactivarlo para cada proveedor en el cuadro de diálogo de configuración.

Algunos algoritmos, incluso si pueden producir un resultado con los datos de entrada dados, devuelven comentarios o información adicional a registrar cuando detectan problemas potenciales con los datos, para advertirle. Asegúrese

de comprobar esos mensajes en el registro si obtiene resultados inesperados.

23.5 El diseñador de modelos

El *diseñador de modelos* permite crear modelos complejos mediante una interfaz sencilla y fácil de usar. Cuando se trabaja con un SIG, la mayoría de las operaciones de análisis no están aisladas, sino que forman parte de una cadena de operaciones. Con el diseñador de modelos, esa cadena de operaciones puede englobarse en un único proceso, lo que facilita su ejecución posterior con un conjunto diferente de entradas. No importa cuántos pasos y algoritmos diferentes implique, un modelo se ejecuta como un único algoritmo, lo que ahorra tiempo y esfuerzo.

El diseñador de modelos puede abrirse desde el menú Procesamiento (**:menuselección:`Procesamiento --> Diseñador de modelos`**).

23.5.1 La interfaz del diseñador de modelos

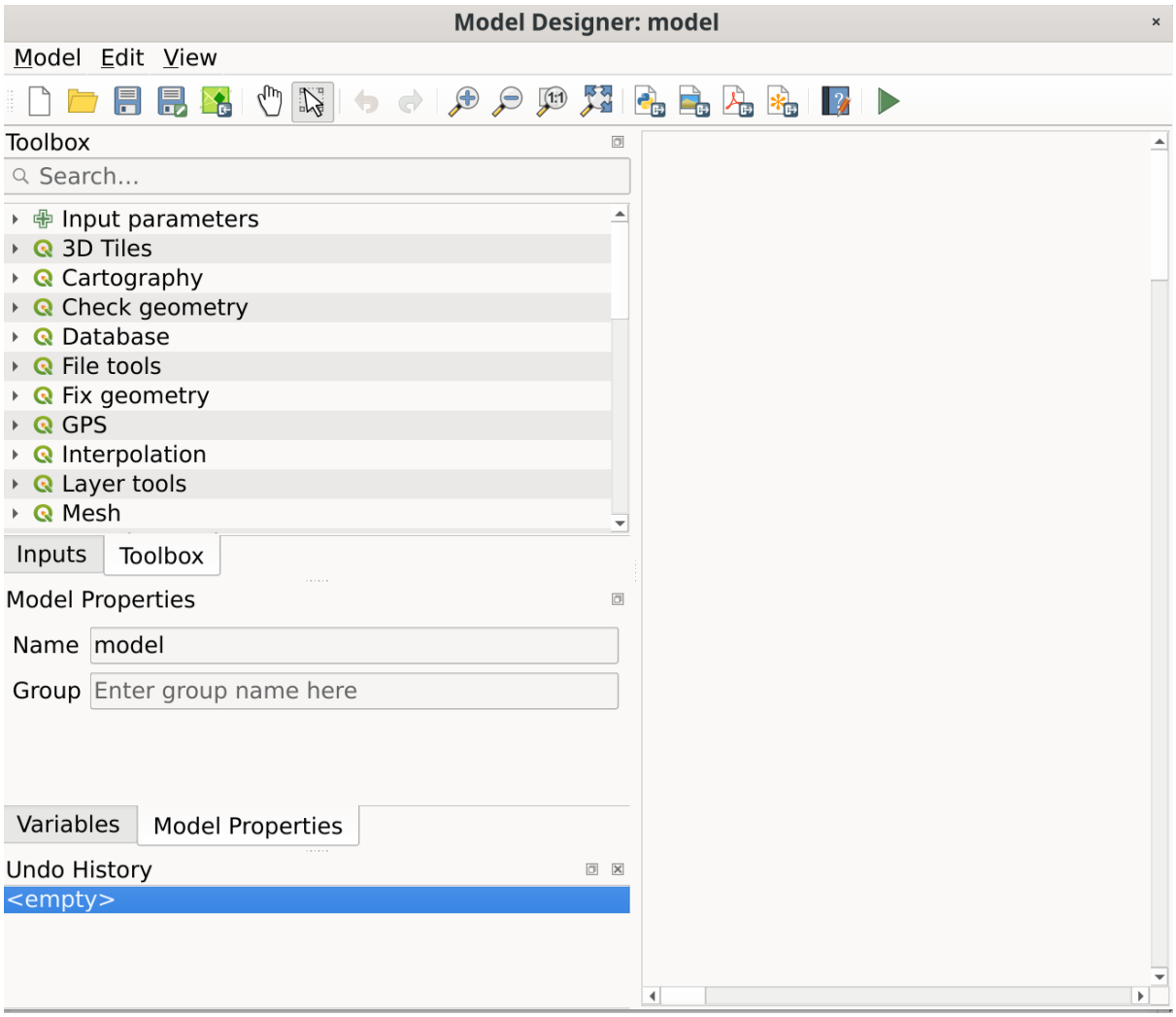


Figura 23.18: Diseñador de modelos

En su parte principal, el modelador tiene un lienzo de trabajo donde se puede construir la estructura del modelo y el flujo de trabajo que representa.

En la parte superior del cuadro de diálogo, diferentes menús y la barra de herramientas *Navigation* dan acceso a una variedad de herramientas.

Menú modelo

Etiqueta	Atajos	Barra de navegación	Descripción
Validar Modelo			Comprueba si existen los algoritmos y las entradas utilizadas en el modelo. Conveniente antes de lanzar un modelo.
Correr Modelo…	`F5`	☑	Ejecuta el modelo
Ejecutar pasos seleccionados…	`Shift+F5`		Ejecuta solo los pasos seleccionados en un modelo, lo que permite ejecutar un subconjunto del modelo.
Reordenar entradas del modelo…			Establece el orden en que se presentan las entradas al usuario en el cuadro de diálogo del algoritmo.
Reordenar capas de salida…			Establece un orden específico que las salidas del modelo deben utilizar al cargar los resultados en un proyecto.
Abrir Modelo…	`Ctrl+O`	☑	Abre un archivo `.model3` para editarlo o ejecutarlo
Guardar Modelo	`Ctrl+S`	☑	Guarda el modelo en el disco como un archivo `.model3`
Guardar Modelo como…	`Ctrl+Shi`	☑	Guarda el modelo en el disco como un nuevo archivo `.model3`
Guardar modelo en proyecto		☑	Incrusta el archivo del modelo en el archivo del proyecto, haciéndolo disponible al compartir el archivo del proyecto.
Ayuda para editar modelo…		☑	Una interfaz para documentar el modelo, los algoritmos, los parámetros y los resultados, así como el autor y el control de versiones.
Exportar ►			
► *Exportar como imagen…*		☑	Guarda el diseño gráfico del modelo en un formato de archivo de imagen (con fines ilustrativos)
► *Exportar como PDF…*			Guarda el diseño gráfico del modelo en un formato de archivo `PDF` (con fines ilustrativos)
► *Exportar como SVG…*			Guarda el diseño gráfico del modelo en un formato de archivo `SVG` (con fines ilustrativos)
► *Exportar como algoritmo de secuencia de comandos…*		☑	Genera un archivo de script de python que incluye las instrucciones del modelo.

Menú Editar

Etiqueta	Atajos	Barra de navegación	Descripción
Seleccionar todo	`Ctrl+A`		Selecciona todos los componentes del modelo en el diseñador.
Ajustar los componentes seleccionados a la cuadrícula			ajusta y alinea los elementos en una cuadrícula
Redo	`Ctrl+Y`	☑	Revertir la última acción cancelada. Véase también el panel *Undo/Redo*.
Undo	`Ctrl+Z`	☑	Cancelar el cambio anterior. Véase también el panel *Undo/Redo*
Cortar	`Ctrl+X`		Corta una selección de componentes del modelo.
Copiar	`Ctrl+C`		Copia una selección de componentes del modelo.
Pegar	`Ctrl+V`		Pega una selección cortada o copiada de componentes de un modelo a otro o dentro del mismo modelo. Los componentes seleccionados mantienen sus propiedades y comentarios originales.
Borrar componentes seleccionados	`Del`		Elimina un componente del modelo.
Añadir cuadro de grupo			Agrega un cuadro en el fondo de los componentes relacionados para agruparlos visualmente. Particularmente útil en modelos grandes para mantener limpio el flujo de trabajo.

Ver menú

Etiqueta	Atajos	Barra de navegación	Descripción
Acercarse ►			Hace zoom a la extensión del cuadro de grupo seleccionado
Acercarse	`Ctrl++`	☑	
Alejarse	`Ctrl+-`	☑	
Acercamiento a 100%	`Ctrl+1`	☑	
Zoom completo	`Ctrl+0`	☑	Muestra todos los componentes en el lienzo actual del diseñador.
Mostrar Comentarios			Muestra los comentarios asociados a cada algoritmo o entrada en el diseñador de modelos.
Habilitar ajuste			
Alternar Visibilidad del Panel	`Ctrl+Tab`		Activa o desactiva los *panels* en el diseñador

Paneles

La parte izquierda de la ventana es una sección con cinco paneles que se pueden usar para agregar nuevos elementos al modelo:

1. *Inputs*: todos los *input parameters* que podrían dar forma a tu modelo
2. *Toolbox*: here you can find available *Processing algorithms*. It also includes input parameters. Use *Search…* box to find algorithms or inputs.
3. *Model Properties*: especifique el nombre (obligatorio) del modelo y el grupo en el que se mostrará en la *Processing Toolbox*
4. *Variables*: Models can contain dedicated *variables* que son únicos y solo están disponibles para ellos. Se puede acceder a estas variables mediante cualquier expresión utilizada dentro del modelo. Son útiles para controlar algoritmos dentro de un modelo y controlar múltiples aspectos del modelo cambiando una sola variable. Las variables se pueden visualizar y modificar en el panel *Variables*.
5. *Undo History*: este panel registrará todo lo que sucede en el modelador, lo que facilita cancelar las cosas que hizo mal.

Acerca de algoritmos disponibles

Some algorithms that can be executed from the *Processing* ► *Toolbox* do not appear in the list of available algorithms when you are designing a model. To be included in a model, an algorithm must have the correct semantic. If an algorithm does not have such a well-defined semantic (for instance, if the number of output layers cannot be known in advance), then it is not possible to use it within a model, and it will not appear in the list of algorithms that you can find in the modeler dialog. On the other hand some algorithms are specific to the modeler. Those algorithms are located within the group “Modeler Tools”.

23.5.2 Creando un modelo

Para crear un modelo hay que seguir dos pasos fundamentales:

1. *Definir los datos de entrada necesarios*. Estos datos de entrada se añadirán a la ventana de parámetros, de tal manera que el usuario puede establecer sus valores cuando se ejecute el modelo. El modelo es, en sí mismo, un algoritmo, por lo que la ventana de parámetros se genera automáticamente para todos los algoritmos disponibles en el entorno de procesamiento.
2. *Definición del flujo de trabajo*. Usando los datos de entrada del modelo, el flujo de trabajo se define agregando algoritmos y seleccionando cómo usan las entradas definidas o las salidas generadas por otros algoritmos en el modelo.

Definir entradas

El primer paso consiste en definir las entradas del modelo. Se encuentran en el panel *Entradas* de la parte izquierda de la ventana del modelador. Si pasa el ratón por encima de las entradas, aparecerá un tooltip con información adicional. Para obtener una lista completa de los parámetros disponibles en el modelador y su correspondencia para scripting, por favor lea *Tipos de entrada y salida para algoritmos de procesamiento*.

Al hacer doble clic sobre un elemento, aparece un diálogo que le permite definir sus características. En función del parámentro, la ventana de diálogo contendrá al menos un elemento (la descripción, que es lo que que el usuario verá al ejecutar el modelo). Por ejemplo, si se añade un valor numérico, como en la figura que se muestra a continuación, además de la descripción de los parámetros, debe establecer un valor por defecto y un rango de valores válidos.

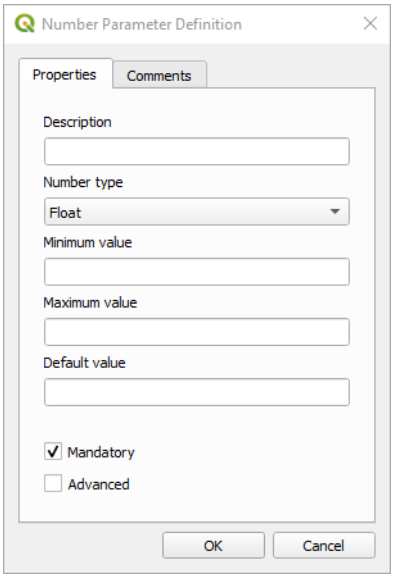


Figura 23.19: Definición de parámetros del Modelo

Puede forzar a que sea obligatorio incluir un valor de entrada marcando la **|casilla|** `Obligatorio`, mientras que marcando la casilla **|desmarcado|** `Avanzado` se establece que la entrada de datos aparezca en la sección `Avanzado`. Esto resulta especialmente útil cuando el modelo cuenta con varios parámetris y algunos de ellos no son triviales, pero usted sigue queriendo elegirlos.

Para cada fuente de datos de entrada añadida, se añade un elemento nuevo al lienzo del modelador.



Figura 23.20: Parámetros del modelo

También puede añadir inputs al modelo, arrastrando el tipo de input al lugar que desee en el lienzo del modelador. Si desea cambiar un parámetro de un input existente, simplemente haga doble clic en él y aparecerá el mismo diálogo.

Al usar un modelo dentro de otro modelo, las entradas y salidas necesarias se mostrarán en el lienzo.

Definicion del flujo de trabajo.

En el siguiente ejemplo, añadimos dos inputs y dos algoritmos. El propósito de este modelo es copiar los valores de elevación tomados de una capa ráster MDE a una capa vectorial de líneas, utilizando el algoritmo `Drape` (cubrir), y después calcular el ascenso total de la capa de líneas utilizando el algoritmo `Climb Along Line` (Ascenso a lo largo de una línea).

En la pestaña *Inputs* , elija ambos inputs como *Capa Vectorial*` para la línea y `Capa Ráster` para el MDE. Y ya estamos listos para añadir los algoritmos para el flujo de trabajo.

Algorithms can be found in the *Toolbox* panel, grouped much in the same way as they are in the Processing toolbox.



Figura 23.21: Entradas del modelo

Para añadir un algoritmo a un modelo, haga doble clic sobre su nombre o arrástrelo, del mismo modo que con los inputs. Al igual que con estos, puede cambiar la descripción del algoritmo y añadir comentarios. En el momento

de añadir un algoritmo, aparecerá un diálogo de ejecución, con contenidos similares a los del panel de ejecución que aparece cuando se ejecuta el algoritmo desde la Caja de Herramientas. En la figura que aparece a continuación se muestran los diálogos de los algoritmos `Cubrir (establecer el valor Z de un ráster)` y `Ascender por una línea`.

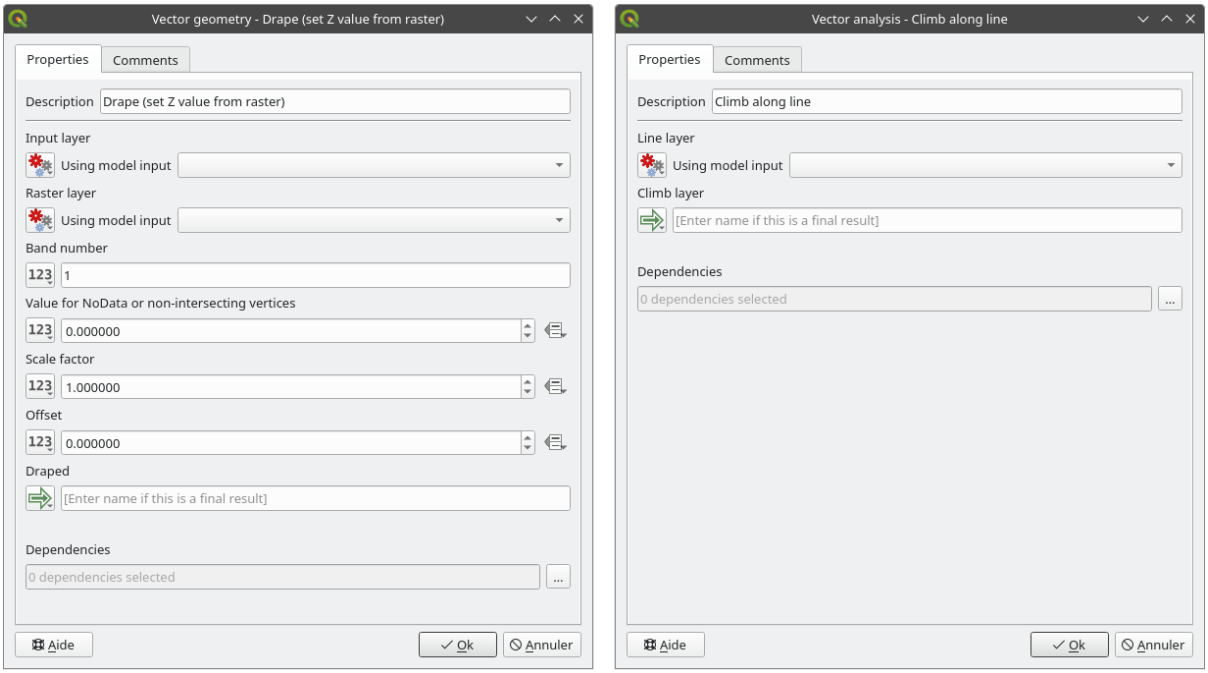


Figura 23.22: Parámetros del Algoritmo del Modelo

Como puede ver, hay sin embargo algunas diferencias. Cada parámetro tiene un menú desplegable al lado que permite controlar cómo se servirá durante el flujo de trabajo:

- Valor: permite asignar un valor estático al parámetro. Dependiendo del tipo de parámetro, el widget le permitirá introducir un número (`5.0`), una cadena (`mytext`), seleccionar capa(s) cargadas en el proyecto QGIS o desde una carpeta, elegir elementos de una lista, …
- Valor precalculado: abre el diálogo *Constructor de expresiones* y permite definir una expresión para rellenar el parámetro. Las entradas del modelo junto con algunas otras estadísticas de la capa están disponibles como **variables** y se listan en la parte superior del diálogo Buscar del Constructor de Expresiones. La expresión se evalúa una vez antes de ejecutar el algoritmo hijo y se utiliza durante la ejecución de dicho algoritmo.
- Entrada del modelo: permite utilizar como parámetro una entrada añadida al modelo. Una vez pulsada, esta opción listará todas las entradas adecuadas para el parámetro.
- Salida del algoritmo: permite utilizar la salida de otro algoritmo como entrada del algoritmo actual. Como en el caso de las entradas de modelo, esta opción listará todas las entradas adecuadas para el parámetro.
- El **parámetro de salida** también tiene las opciones anteriores en su menú desplegable:
  - añadir salidas estáticas para algoritmos hijos, por ejemplo, guardar siempre la salida de un algoritmo hijo en un geopackage predefinido o en una capa postgres.
  - utilizar valores de salida basados en una expresión para los algoritmos hijos, por ejemplo, generar un nombre de archivo automático basado en la fecha de hoy y guardar los resultados en ese archivo
  - utilizar una entrada de modelo, por ejemplo, la entrada de modelo *Archivo/Carpeta* para especificar un archivo o carpeta de salida
  - utilizar la salida de otro algoritmo, por ejemplo, la salida del algoritmo *Crear directorio* (de *Herramientas del modelador*)

– una opción adicional  Salida del modelo hace que la salida del algoritmo esté disponible en el modelo. Si una capa generada por el algoritmo sólo se va a utilizar como entrada para otro algoritmo, no edite ese cuadro de texto.

En la siguiente imagen puede ver las dos entradas de parámetros, definidas como `Entrada del Modelo` y capa temporal de salida:

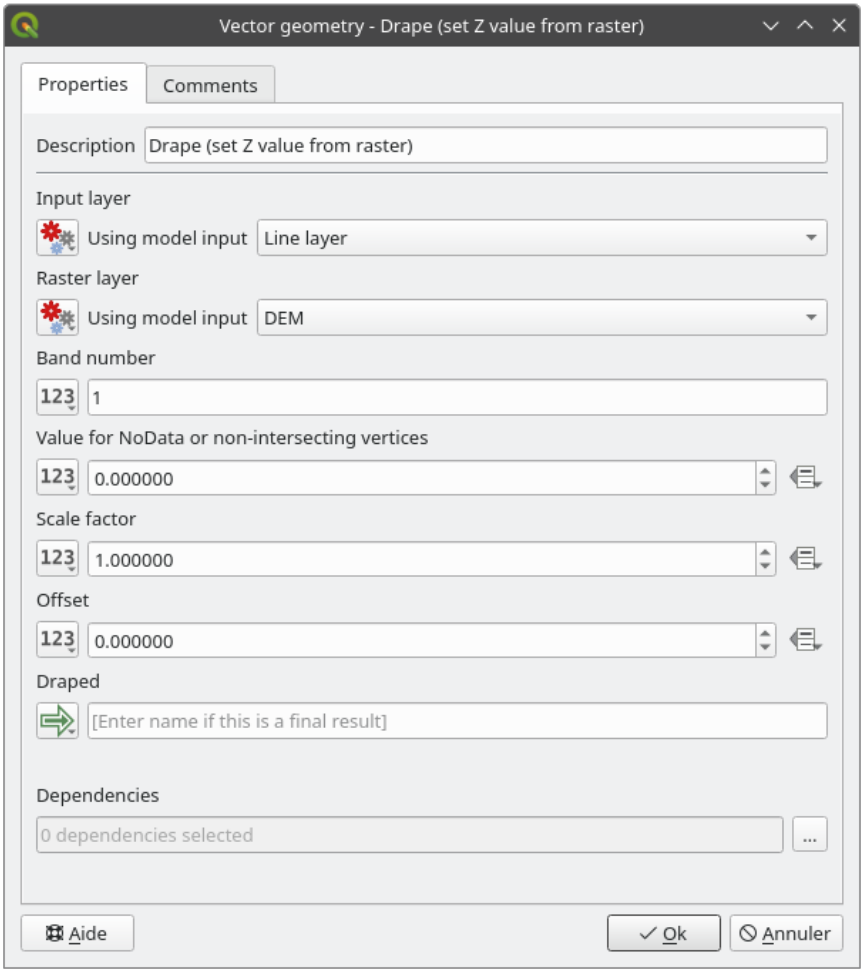


Figura 23.23: Parámetros de entrada y salida del algoritmo

También encontrará un parámetro adicional llamado *Dependencies* que no está disponible al llamar al algoritmo desde la caja de herramientas. Este parámetro le permite definir el orden en que se ejecutan los algoritmos, definiendo explícitamente un algoritmo como *parent* del actual. Esto obligará a que el algoritmo *parent* se ejecute antes que el actual.

When you use the output of a previous algorithm as the input of your algorithm, that implicitly sets the previous algorithm as parent of the current one (and places the corresponding arrow in the modeler canvas). However, in some cases an algorithm might depend on another one even if it does not use any output object from it (for instance, an algorithm that executes a SQL sentence on a PostgreSQL database and another one that imports a layer into that same database). In that case, just select the previous algorithm in the *Dependencies* parameter and they will be executed in the correct order.

Una vez que se hayan asignado valores válidos a todos los parámetros, haga clic en *Aceptar* y el algoritmo se agregará al lienzo. Se vinculará a los elementos del lienzo (algoritmos o entradas) que proporcionan objetos que se utilizan como entradas para el algoritmo.

Elements can be dragged to a different position on the canvas using the  Select/Move/Link Item tool. This is useful to make the structure of the model clearer and more intuitive. You can also resize the elements, grasping their border.

This is particularly useful if the description of the input or algorithm is long. With *View* ► *Enable snapping* option checked, items resizing or displacement can be bound to a virtual grid, for a more visually structured algorithm design.

Los enlaces entre elementos se actualizan automáticamente y puede ver un botón + en la parte superior e inferior de cada algoritmo. Al hacer clic en el botón, se enumerarán todas las entradas y salidas del algoritmo para que pueda tener una descripción general rápida.

You can create or remove connections between model components using an interactive drag-and-drop interface. Each parameter and algorithm item on the model canvas includes small circular sockets, output sockets on the right and input sockets on the left. To connect components, click and drag from an output socket (either from a model input or an algorithm) to an input socket on another algorithm. A visual edge follows the cursor and completes the connection when released. To remove a connection, drag the edge away from the input socket. The algorithm dialogs update automatically to reflect changes made using the drag-and-drop interface.

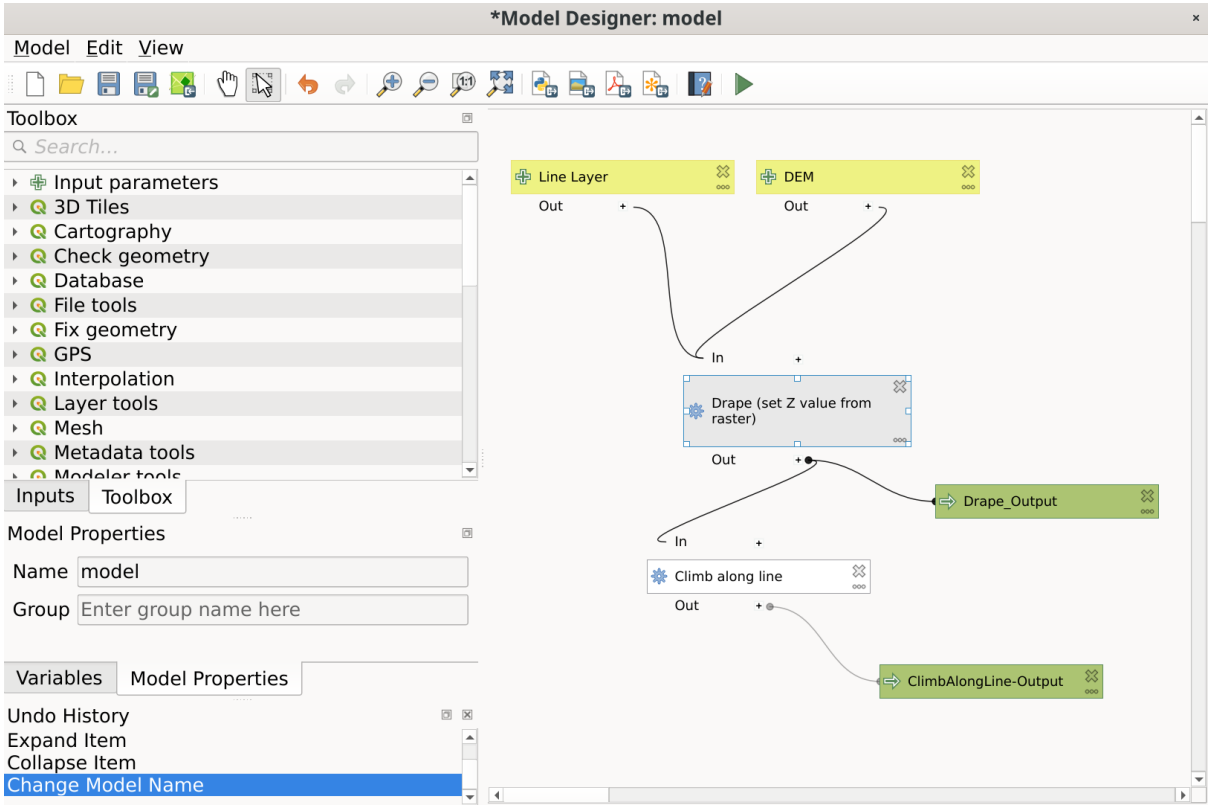


Figura 23.24: Un modelo completo

With the *Edit* ► *Add Group Box* tool, you can add a draggable *box* to the canvas. This feature is very useful in big models to group related elements in the modeler canvas and to keep the workflow clean. For example we might group together all the algorithms of the example:

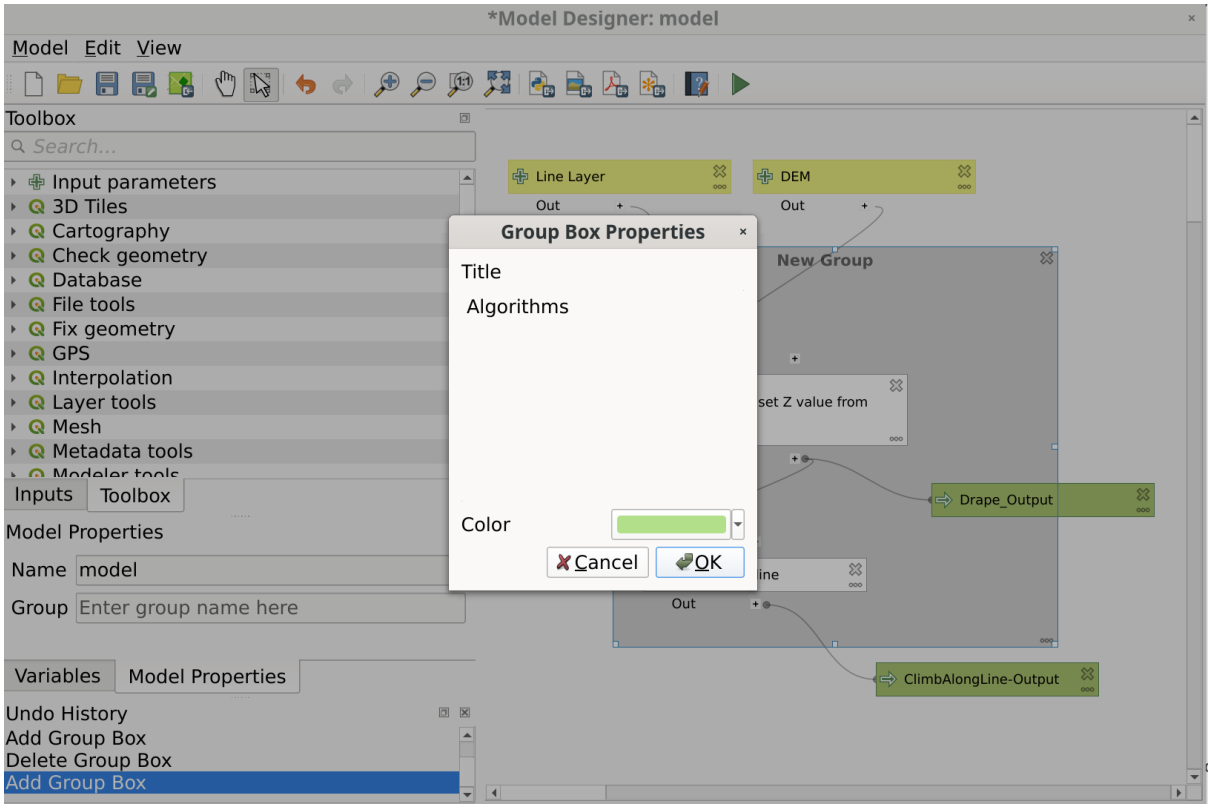


Figura 23.25: Recuadro de grupo de modelo

Puedes cambiar el nombre y el color de las cajas. Los cuadros de grupo son muy útiles cuando se usan junto con la herramienta *View* ► *Zoom To* ►, que le permite hacer zoom en una parte específica del modelo. También puede acercar y alejar usando la rueda del mouse.

Es posible que desee cambiar el orden de las entradas y cómo se enumeran en el cuadro de diálogo del modelo principal. En la parte inferior del panel `Entrada` encontrará el botón `Reordenar entradas del modelo...` y al hacer click en él, aparecerá un nuevo cuadro de diálogo que le permitirá cambiar el orden de las entradas:

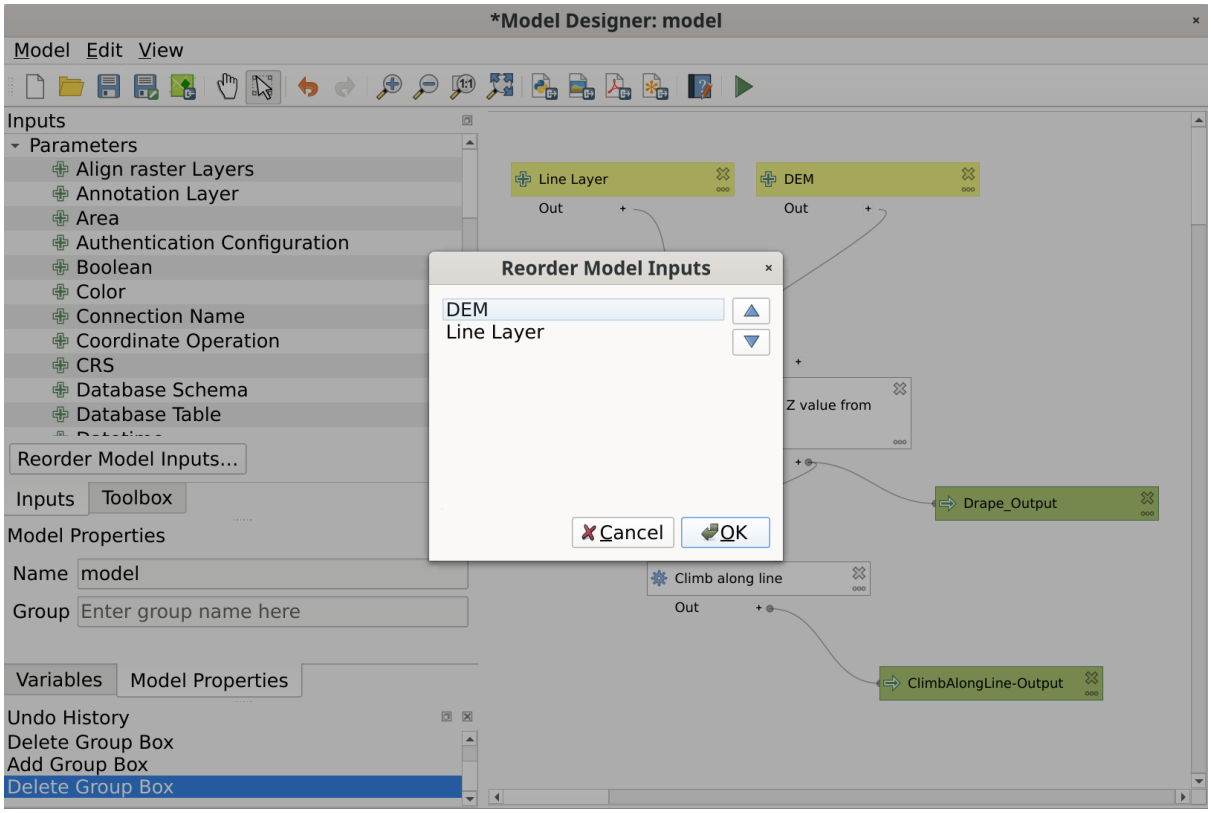


Figura 23.26: Reordenar Entradas del Modelo

También existe la posibilidad de establecer un orden específico que las salidas del modelo deben utilizar al cargar los resultados en un proyecto. Esto proporciona al creador del modelo un medio para asegurar que las capas están ordenadas lógicamente en el lienzo cuando se ejecuta un modelo, como colocar una salida de capa vectorial sobre una salida de capa ráster, o una capa de puntos sobre una capa de polígonos. El creador del modelo también puede establecer un «Nombre de grupo» opcional para las salidas para agruparlas automáticamente dentro del árbol de capas utilizando un nuevo nombre de grupo o añadiéndolas a un grupo existente. En el menú `Modelo` encontrará la entrada `Reordenar capas de salida...` y haciendo clic sobre ella aparecerá un nuevo diálogo que le permitirá cambiar el orden de las capas de salida:

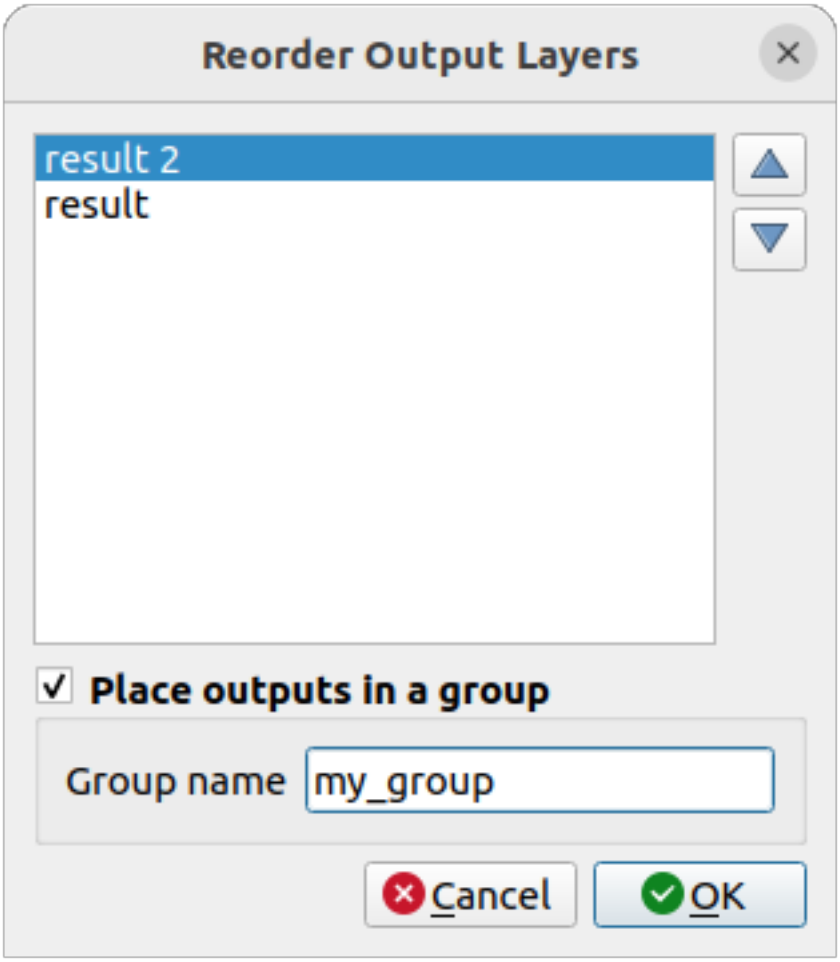


Figura 23.27: Reordenar las capas de salida

También se pueden agregar comentarios a las entradas o algoritmos presentes en el modelador. Esto se puede hacer yendo a la pestaña *Comment* del ítem o haciendo clic derecho. En la misma pestaña, se puede configurar un color de forma manual para comentarios de modelos individuales. Los comentarios solo son visibles en el lienzo del modelador y no en el cuadro de diálogo del algoritmo final; se pueden ocultar desactivando *View* ► *Show Comments*.

Su modelo se puede ejecutar de varias maneras:

- Puede ejecutar todo el modelo haciendo clic en Ejecutar modelo… en la barra de herramientas, haciendo clic con el botón derecho del ratón en el modelo en el *Panel del navegador* o en *Modelo* ► *Ejecutar Modelo…* : cuando se utiliza el editor para ejecutar un modelo, cualquier valor no predeterminado se guardará en las entradas. Esto significa que, al ejecutar el modelo más tarde desde el editor, el cuadro de diálogo aparecerá rellenado con esos valores en cualquier ejecución posterior.

- Puede seleccionar elementos del modelo y ejecutar solo ese subconjunto del modelo: pulse la opción *Ejecutar pasos seleccionados…* del menú *Modelos* ► o del menú contextual de un algoritmo seleccionado. El estado inicial se tomará de cualquier ejecución anterior del modelo a través del editor, por lo que los resultados de los pasos anteriores del modelo estarán disponibles para los pasos seleccionados. Esto permite corregir partes de un modelo grande sin tener que ejecutar constantemente todo el modelo para realizar pruebas. Es especialmente útil cuando los pasos anteriores del modelo requieren mucho tiempo.
- Puede ejecutar un subconjunto del modelo, comenzando por un algoritmo específico: haga clic con el botón derecho del ratón en el algoritmo y seleccione *Run from Here…*. Del mismo modo, los valores se toman de ejecuciones anteriores.

- Para poder utilizar el algoritmo de la caja de herramientas de procesamiento, es necesario guardarlo y cerrar el cuadro de diálogo del modelador, a fin de permitir que la caja de herramientas actualice su contenido.

Documentando tu modelo

Debe documentar su modelo, y esto se puede hacer desde el propio modelador. Haga clic en el botón  Edit model help y aparecerá un cuadro de diálogo como el que se muestra a continuación.

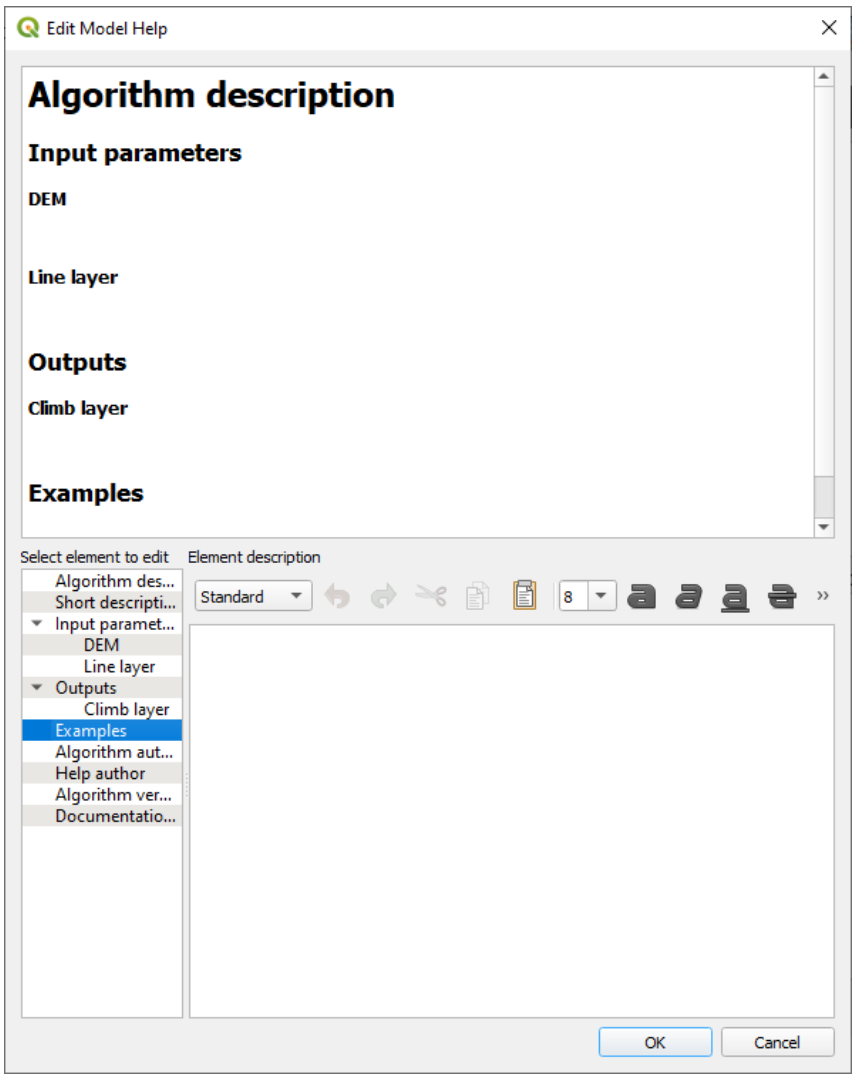


Figura 23.28: Ayuda de Edición

En la parte derecha verá una página HTML sencilla, creada a partir de la descripción de los parámetros de entrada y salida del algoritmo, junto con algunos elementos adicionales como una descripción general del modelo o de su autor. Además, hay una sección de ejemplos donde puede introducir sus propios ejemplos personalizados para ayudar a explicar el uso del modelo. La primera vez que abre el editor de ayuda, todas estas descripciones están vacías, pero puede editarlas utilizando los elementos de la parte izquierda del cuadro de diálogo. Seleccione un elemento en la parte superior y escriba su descripción en el cuadro de texto inferior.

Modelo de ayuda se guarda como parte de un modelo en si.

23.5.3 Guardar y cargar modelos.

Guardando modelos

Utilice el botón  Guardar modelo para guardar el modelo actual y el botón  Abrir modelo para abrir un modelo previamente guardado. Los modelos se guardan con la extensión `.model3`. Si el modelo ya se ha guardado desde la ventana del modelador, no se le pedirá un nombre de archivo. Dado que ya hay un archivo asociado con el modelo, ese archivo se utilizará para guardados posteriores.

Antes de guardar un modelo, debe ingresar un nombre y un grupo para él en los cuadros de texto en la parte superior de la ventana.

Los modelos guardados en la carpeta `models` (la carpeta predeterminada cuando se le solicita un nombre de archivo para guardar el modelo) aparecerán en la caja de herramientas en la rama correspondiente. Cuando se invoca la caja de herramientas, busca en la carpeta `models` archivos con la extensión `.model3` y carga los modelos que contiene. Dado que un modelo es en sí mismo un algoritmo, se puede agregar a la caja de herramientas como cualquier otro algoritmo.

Los modelos también se pueden guardar dentro del archivo del proyecto usando el botón  Guardar modelo en proyecto. Los modelos guardados con este método no se escribirán como archivos `.model3` en el disco, sino que se incrustarán en el archivo del proyecto.

Los modelos de proyecto están disponibles en el menú  *Modelos de proyecto* de la caja de herramientas y en la opción de menú *Proyecto* ► *Modelos*.

La carpeta de los modelos se puede configurar desde el diálogo Procesos, bajo el grupo *Modelador*.

Los modelos cargados desde la carpeta `models` aparecen no solo en la caja de herramientas, sino también en el árbol de algoritmos en la pestaña :guilabel:‘Algoritmos` de la ventana del modelador. Eso significa que puede incorporar un modelo como parte de un modelo más grande, al igual que otros algoritmos.

Los modelos se mostrarán en el panel *Navegador* y pueden ser ejecutados desde allí.

Exportando un modelo como un script Python

Como veremos en un capítulo posterior, los algoritmos de procesamiento se pueden llamar desde la consola QGIS Python, y se pueden crear nuevos algoritmos de procesamiento usando Python. Una forma rápida de crear un script de Python es crear un modelo y luego exportarlo como un archivo de Python.

Para hacerlo, haga click en  Exportar como algoritmo de script … en el lienzo del modelador o haga clicl con el botón derecho en el nombre del modelo en la caja de herramientas de procesamiento y seleccione | saveAsPython |:sup:`Exportar modelo como algoritmo de Python … `.

Exportar un modelo como imagen, PDF o SVG

También se puede exportar un modelo como imagen, SVG o PDF (con fines ilustrativos) haciendo click en  Exportar como imagen,  Exportar como PDF o  Exportar como SVG.

23.5.4 Editar un modelo.

Puede editar el modelo que está creando actualmente, redefiniendo el flujo de trabajo y las relaciones entre los algoritmos y las entradas que definen el modelo.

También puede editar un modelo existente desde el *panel del navegador* seleccionando la opción *Editar modelo…*.

Si hace clic derecho en un algoritmo en el lienzo, verá un menú contextual como el que se muestra a continuación:

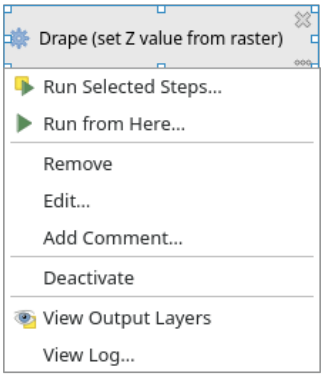


Figura 23.29: Click derecho en el modelador

- Opciones para *ejecutar parte del modelo* han sido expuestos anteriormente.
- Al seleccionar la opción *Eliminar*, se eliminará el algoritmo seleccionado. Un algoritmo solo se puede eliminar si no hay otros algoritmos que dependan de él. Es decir, si ninguna salida del algoritmo se utiliza como entrada en otro diferente.
- Al seleccionar la opción *Editar…*, se mostrará el cuadro de diálogo de parámetros para que pueda cambiar las entradas y los valores de los parámetros. No todos los elementos de entrada disponibles en el modelo aparecerán como entradas disponibles. Las capas o los valores generados en un paso más avanzado del flujo de trabajo definido por el modelo no estarán disponibles si provocan dependencias circulares.

  Seleccione los nuevos valores y haga clic en el botón *Aceptar* como de costumbre. En consecuencia las conexiones entre los elementos del modelo cambiarán en el lienzo del modelador.
- El *Agregar comentario…* le permite agregar un comentario al algoritmo para describir mejor el comportamiento.

> **Nota**
>
> Al hacer clic con el botón derecho del ratón sobre un parámetro de entrada, también podrás *Eliminar*, *Editar…* y *Añadir comentario…*

- Un modelo se puede ejecutar parcialmente desactivando algunos de sus algoritmos. Para hacerlo, seleccione la opción *Desactivar* en el menú contextual que aparece al hacer clic derecho sobre un elemento del algoritmo. El algoritmo seleccionado y todos los del modelo que dependen de él se mostrarán en gris y no se ejecutarán como parte del modelo.

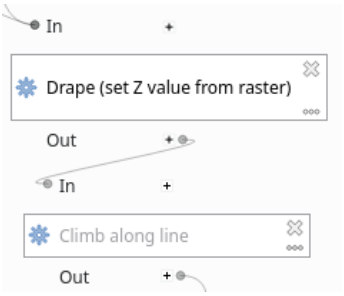


Figura 23.30: Modelo con un algoritmo desactivado

Al hacer clic con el botón derecho del ratón sobre un algoritmo que no está activo, aparecerá la opción de menú *Activar*, que puede utilizar para reactivarlo.

- Al editar un modelo a través del diseñador (y después de haber ejecutado ese modelo), al hacer clic con el botón derecho del ratón en cualquier paso secundario del modelo se puede seleccionar *Ver capas de salida*. Esto añadirá las capas de salida de ese paso como nuevas capas en el proyecto QGIS actual.

  Esta acción está disponible para todos los algoritmos secundarios del modelo, incluso si el modelo no está configurado para utilizar los resultados de esos algoritmos secundarios como resultados del modelo. Está diseñada como una acción útil para la depuración. Si un modelo falla (o da resultados inesperados), puede rastrear el modelo y ver los resultados de los pasos que sospecha que son problemáticos. Evita la necesidad de añadir resultados temporales a un modelo y volver a ejecutarlo para realizar pruebas.

  Además, esta acción siempre está disponible después de ejecutar el modelo, incluso si el modelo en sí ha fallado (por ejemplo, debido a un paso mal configurado más adelante en el modelo).

- La acción *Ver registro* le permite ver el registro de cada algoritmo secundario después de cerrar el cuadro de diálogo del algoritmo.

23.6 La interfaz de procesamiento por lote

23.6.1 Introducción

Todos los algoritmos (incluyendo modelos) se pueden ejecutar como un proceso por lotes. Es decir, que se pueden ejecutar utilizando no sólo un único conjunto de insumos, sino varios de ellos y ejecutar el algoritmo tantas veces sea necesario. Esto es útil al procesar grandes cantidades de datos, ya que no es necesario poner en marcha el algoritmo muchas veces desde la caja de herramientas.

Para ejecutar un algoritmo como un proceso por lotes, haga clic en su nombre en la caja de herramientas y seleccionar la opción *Ejecutar como proceso por lotes* en el menú emergente que aparecerá.

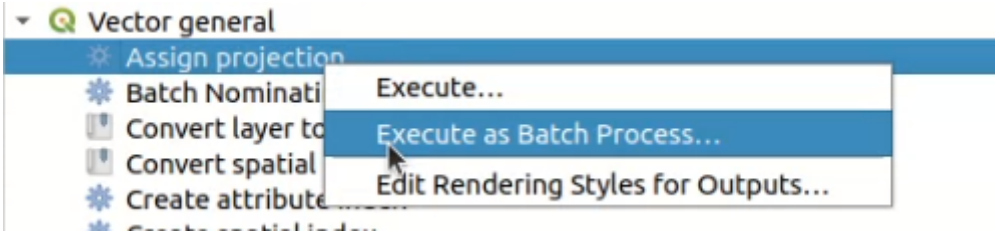


Figura 23.31: Procesamiento por lotes con clic derecho

Si tiene que ejecutar el diálogo del algoritmo abierto, también puede iniciar la interfaz de procesamiento por lotes desde allí, haga clic en el botón *Ejecutar como proceso por lotes…*.

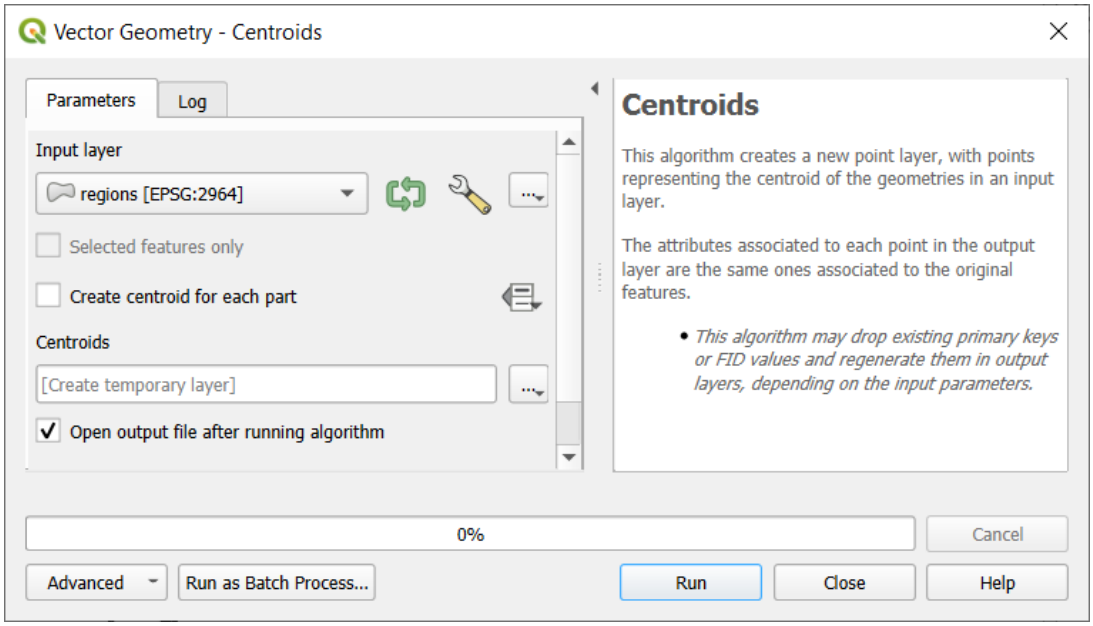


Figura 23.32: diálogo de procesamiento por lotes de algoritmo

23.6.2 La tabla de parámetros

Ejecutar un proceso por lotes es similar a realizar una única ejecución de un algoritmo. Hay que definir los valores de los parámetros, pero en este caso no necesitamos un único valor para cada parámetro, sino un conjunto de ellos, uno por cada vez que haya que ejecutar el algoritmo. Los valores se introducen mediante una tabla como la que se muestra a continuación, donde cada fila es una iteración y las columnas son los parámetros del algoritmo.

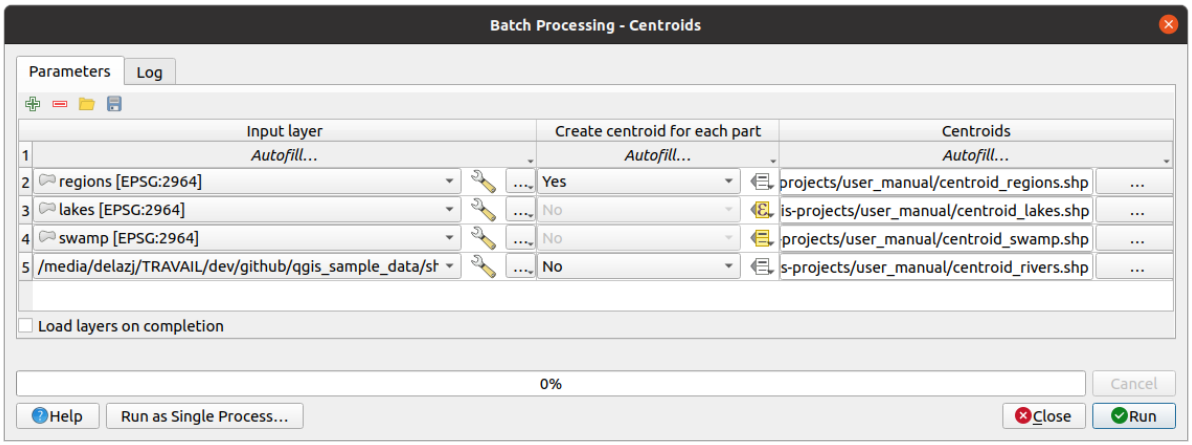


Figura 23.33: Procesamiento por Lotes

Desde la barra de herramientas superior, puede:

- Activar modo avanzado: Disponible sólo cuando el algoritmo tiene parámetros marcados como avanzados, este botón permite mostrar u ocultar dichos parámetros en el diálogo de lotes.
- Añadir fila: añade una nueva entrada de tratamiento para la configuración
- Remover fila(s): elimina las filas seleccionadas de la tabla. La selección de filas se realiza haciendo clic en el número de la izquierda y permite la combinación de teclas . para la selección múltiple.

-  Abrir un archivo de configuración de procesamiento por lotes
-  Guardar la configuración del procesamiento por lotes en un archivo `.JSON` que puede ser ejecutado posteriormente

De forma predeterminada, la tabla contiene sólo dos filas:

- La primera fila muestra en cada celda un menú desplegable *Autofill…* ► con *options* para rellenar rápidamente las celdas de abajo. Las opciones disponibles dependen del tipo de parámetro.
- La segunda fila (así como cada una de las siguientes) representa una única ejecución del algoritmo, y cada celda contiene el valor de uno de los parámetros. Es similar al diálogo de parámetros que se ve al ejecutar un algoritmo desde la caja de herramientas, pero con una disposición diferente.

En la parte inferior de la tabla, puede establecer si  *Cargar capas al completar*.

Una vez que el tamaño de la tabla se ha establecido, este tiene que ser llenado con los valores deseados.

23.6.3 Llenado de la tabla de parámetros

Para la mayoría de los parámetros, establecer el valor es trivial. El widget apropiado, igual que en el diálogo de proceso *single*, se proporciona, permitiendo simplemente escribir el valor, o seleccionarlo de una lista de posibles valores, dependiendo del tipo de parámetro. Esto también incluye el widget de definición de datos, cuando es compatible.

Para automatizar la definición del proceso por lotes y evitar rellenar la tabla celda a celda, puede pulsar el menú *Autorellenar…* de un parámetro y seleccionar cualquiera de las siguientes opciones para sustituir los valores de la columna:

- *Rellenar* tomará la entrada para el primer proceso y la introducirá para todos los demás procesos.
-  *Calcular por expresión…* le permitirá crear una nueva expresión de QGIS que se utilizará para actualizar todos los valores existentes dentro de esa columna. Los valores de los parámetros existentes (incluyendo los de otras columnas) están disponibles para su uso dentro de la expresión a través de *variables*. Por ejemplo, establecer el número de segmentos en función de la distancia de la zona de influencia de cada capa:

```
CASE WHEN @DISTANCE > 20 THEN 12 ELSE 8 END
```

- *Añadir valores por expresión…* añadirá nuevas filas utilizando los valores de una expresión que devuelva una matriz (a diferencia de *Calcular por expresión…*, que sólo funciona con las filas existentes). El caso de uso previsto es permitir rellenar el diálogo de lotes utilizando series numéricas complejas. Por ejemplo, añadir filas para un buffer de lotes utilizando la expresión `generate_series(100, 1000, 50)` para el parámetro de distancia da como resultado nuevas filas con los valores 100, 150, 200, …. 1000.
- Al establecer un parámetro de archivo o de capa, se ofrecen más opciones:
  - *Añadir Archivos por Patrón…*: añade nuevas filas a la tabla para los ficheros que coincidan con un *Patrón de archivo* en una carpeta a *Mirar*. P.ej. `*.shp` añadirá a la lista todos los archivos `SHP` de la carpeta. Confirme **|caja de confirmación|** *Buscar recursivamente* para mirar también las subcarpetas.
  - *Seleccionar Archivos…* individualmente en el disco
  - *Añadir Todos los Archivos de un Directorio…*
  - *Seleccionar desde Capas Abiertas…* en el proyecto activo

El parámetro de datos de salida expone las mismas capacidades que cuando se ejecuta el algoritmo como un proceso único. Dependiendo del algoritmo, la salida puede ser:

- omitido, si la celda se deja vacía
- saved as a temporary layer: fill the cell with your chosen output name, select  *Create Temporary Layer* from the … drop-down, and remember to tick the  *Load layers on completion* checkbox. For temporary layers, the value you provide will be used as the layer name. You can use the *Autofill* options to construct that name.

- saved as a plain file (`.SHP`, `.GPKG`, `.XML`, `.PDF`, `.JPG`,...): choose *Select File/Folder...* from the ... drop-down. For plain files, you can set the path using the *Autofill* options exposed beforehand.

  E.g. use *Calculate by Expression...* to set output file names to complex expressions like:

```
'/home/me/stuff/buffer_' || left(@INPUT, 30) || '_' || @DISTANCE || '.shp'
```

  You can also type the file path directly or use the file chooser dialog that appears when clicking on the accompanying *Select File/Folder...* button. Once you select the file, a new dialog is shown to allow for auto-completion of other cells in the same column (same parameter).

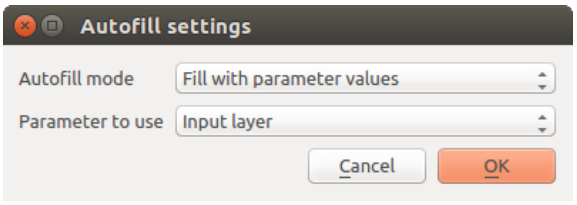


Figura 23.34: Guardar Procesamiento por lotes

  Si se selecciona el valor predeterminado (*No auto rellenar*), sólo pondrá el nombre del archivo seleccionado en la celda seleccionada de la tabla de parámetros. Si se selecciona cualquiera de las otras opciones, todas las celdas **debajo** de la seleccionada se rellenarán automáticamente en base a un criterio definido:

  - *Rellenar con números*: añade incrementalmente un número al nombre del archivo
  - *Rellenar con valores de parámetros*: puede seleccionar un parámetro cuyo valor en la misma fila se anexa al nombre del archivo. Esto es especialmente útil para nombrar los objetos de datos de salida según los de entrada.

- guardado como una capa dentro de un contenedor de base de datos:

```
# Indicate a layer within a GeoPackage file
ogr:dbname='C:/Path/To/Geopackage.gpkg' table="New_Table" (geom)

# Use the "Calculate By Expression" to output to different layers in a␣
↪GeoPackage
'ogr:dbname=\'' || @project_folder || '/Buffers.gpkg\' table="' || @INPUT || '_
↪' || @DISTANCE || '" (geom)'
```

23.6.4 Ejecutar el proceso por lotes

Para ejecutar el proceso por lotes una vez introducidos todos los valores necesarios, basta con hacer clic en *Ejecutar*. El panel *Registro* se activa y muestra los detalles y pasos del proceso de ejecución. El progreso de la tarea global por lotes se mostrará en la barra de progreso de la parte inferior del diálogo.

23.7 Utilizar algoritmos de procesamiento desde la consola

La consola permite a los usuarios avanzados incrementar su productividad y realizar operaciones complejas que no se pueden realizar utilizando cualquiera de los otros elementos de la GUI del marco de procesamiento. Modelos que involucran varios algoritmos se pueden definir mediante la interfaz de línea de comandos y operaciones adicionales tales como bucles y sentencias condicionales que se pueden añadir para crear flujos de trabajo más flexibles y potentes.

No hay una consola de procesamiento en QGIS, pero todos los comandos de procesamiento están disponibles en su lugar desde el constructor de QGIS *consola de Python*. Eso significa que puede incorporar esos comandos en el trabajo

de su consola y conectar algoritmos de procesamiento a todas las demás funciones (incluidos los métodos de la API de QGIS) disponibles desde allí.

El código se puede ejecutar desde la consola de Python, incluso si no especifica ningún método de procesamiento, se puede convertir en un nuevo algoritmo que más tarde puede llamar desde la caja de herramientas, el modelador gráfico o algún otro componente, tal como lo hace con cualquier otro algoritmo. De hecho, algunos de los algoritmos que se pueden encontrar en la caja de herramientas son sencillas secuencias de comandos.

En esta sección, veremos como utilizar algoritmos de procesado desde la consola de Python de QGIS, y también cómo escribir algoritmos utilizando Python.

23.7.1 Invocando algoritmos desde la consola de Python

Lo primero que tiene que hacer es importar las funciones de procesamiento con la siguiente línea:

```
>>> from qgis import processing
```

Ahora, básicamente hay una cosa (interesante) que puedes hacer con eso desde la consola: ejecutar un algoritmo. Eso se hace usando el método `run()`, que toma el nombre del algoritmo a ejecutar como su primer parámetro, y luego un número variable de parámetros adicionales dependiendo de los requisitos del algoritmo. Entonces, lo primero que necesita saber es el nombre del algoritmo a ejecutar. Ese no es el nombre que ve en la caja de herramientas, sino un nombre de línea de comandos único. Para encontrar el nombre correcto para su algoritmo, puede usar `processingRegistry`. Escriba la siguiente línea en su consola:

```
>>> for alg in QgsApplication.processingRegistry().algorithms():
        print(alg.id(), "->", alg.displayName())
```

Verá algo como esto (con algunos guiones adicionales agregados para mejorar la legibilidad).

```
3d:tessellate ---------------> Tessellate
gdal:aspect -----------------> Aspect
gdal:assignprojection -------> Assign projection
gdal:buffervectors ----------> Buffer vectors
gdal:buildvirtualraster -----> Build Virtual Raster
gdal:cliprasterbyextent -----> Clip raster by extent
gdal:cliprasterbymasklayer --> Clip raster by mask layer
gdal:clipvectorbyextent -----> Clip vector by extent
gdal:clipvectorbypolygon ----> Clip vector by mask layer
gdal:colorrelief ------------> Color relief
gdal:contour ----------------> Contour
gdal:convertformat ----------> Convert format
gdal:dissolve ---------------> Dissolve
...
```

Esa es una lista de todos los ID de algoritmos disponibles, ordenados por nombre de proveedor y nombre de algoritmo, junto con sus nombres correspondientes.

Una vez que conozca el nombre de la línea de comandos del algoritmo, lo siguiente que debe hacer es determinar la sintaxis correcta para ejecutarlo. Eso significa saber qué parámetros se necesitan al llamar al método `run()`.

Existe un método para describir un algoritmo en detalle, que se puede utilizar para obtener una lista de los parámetros que requiere un algoritmo y las salidas que generará. Para obtener esta información, puede utilizar el método `algorithmHelp(id_of_the_algorithm)`. Utilice el ID del algoritmo, no el nombre descriptivo completo.

Llamando al método con `native:buffer` como parámetro (`qgis:buffer` es un alias para `native:buffer` y también funcionará), obtienes la siguiente descripción:

```
>>> processing.algorithmHelp("native:buffer")
Buffer (native:buffer)

This algorithm computes a buffer area for all the features in an
```

(continúe en la próxima página)

(proviene de la página anterior)

```
input layer, using a fixed or dynamic distance.

The segments parameter controls the number of line segments to
use to approximate a quarter circle when creating rounded
offsets.

The end cap style parameter controls how line endings are handled
in the buffer.

The join style parameter specifies whether round, miter or
beveled joins should be used when offsetting corners in a line.

The miter limit parameter is only applicable for miter join
styles, and controls the maximum distance from the offset curve
to use when creating a mitered join.


----------------
Input parameters
----------------

INPUT: Input layer

   Parameter type: QgsProcessingParameterFeatureSource

   Accepted data types:
           - str: layer ID
           - str: layer name
           - str: layer source
           - QgsProcessingFeatureSourceDefinition
           - QgsProperty
           - QgsVectorLayer

DISTANCE: Distance

   Parameter type: QgsProcessingParameterDistance

   Accepted data types:
           - int
           - float
           - QgsProperty

SEGMENTS: Segments

   Parameter type: QgsProcessingParameterNumber

   Accepted data types:
           - int
           - float
           - QgsProperty

END_CAP_STYLE: End cap style

   Parameter type: QgsProcessingParameterEnum

   Available values:
           - 0: Round
           - 1: Flat
           - 2: Square

   Accepted data types:
```

(continúe en la próxima página)

(proviene de la página anterior)

```
           - int
           - str: as string representation of int, e.g. '1'
           - QgsProperty

JOIN_STYLE: Join style

   Parameter type: QgsProcessingParameterEnum

   Available values:
           - 0: Round
           - 1: Miter
           - 2: Bevel

   Accepted data types:
           - int
           - str: as string representation of int, e.g. '1'
           - QgsProperty

MITER_LIMIT: Miter limit

   Parameter type: QgsProcessingParameterNumber

   Accepted data types:
           - int
           - float
           - QgsProperty

DISSOLVE: Dissolve result

   Parameter type: QgsProcessingParameterBoolean

   Accepted data types:
           - bool
           - int
           - str
           - QgsProperty

OUTPUT: Buffered

   Parameter type: QgsProcessingParameterFeatureSink

   Accepted data types:
           - str: destination vector file, e.g. 'd:/test.shp'
           - str: 'memory:' to store result in temporary memory layer
           - str: using vector provider ID prefix and destination URI,
                  e.g. 'postgres:...' to store result in PostgreSQL table
           - QgsProcessingOutputLayerDefinition
           - QgsProperty

----------------
Outputs
----------------

OUTPUT:  <QgsProcessingOutputVectorLayer>
   Buffered
```

Ahora tienes todo lo que necesitas para ejecutar cualquier algoritmo. Como ya hemos mencionado, los algoritmos se pueden ejecutar usando: `run()`. Su sintaxis es la siguiente:

```
>>> processing.run(name_of_the_algorithm, parameters)
```

Donde parámetros es un diccionario de parámetros que dependen del algoritmo que desea ejecutar, y es exactamente

la lista que le proporciona el método `algorithmHelp()`.

```
>>> processing.run("native:buffer", {'INPUT': '/data/lines.shp',
                'DISTANCE': 100.0,
                'SEGMENTS': 10,
                'DISSOLVE': True,
                'END_CAP_STYLE': 0,
                'JOIN_STYLE': 0,
                'MITER_LIMIT': 10,
                'OUTPUT': '/data/buffers.shp'})
```

Si un parámetro es opcional y no desea utilizarlo, no lo incluya en el diccionario.

Si no se especifica un parámetro, se utilizará el valor predeterminado.

Dependiendo del tipo de parámetro, los valores se introducen de manera diferente. La siguiente lista da una rápida revisión de cómo introducir los valores para cada tipo de parámetro de entrada.

- Capa ráster, capa vectorial o tabla. Simplemente use una cadena con el nombre que identifica el objeto de datos a usar (el nombre que tiene en la Tabla de contenido de QGIS) o un nombre de archivo (si la capa correspondiente no está abierta, se abrirá pero no se agregará al lienzo del mapa) . Si tiene una instancia de un objeto QGIS que representa la capa, también puede pasarla como parámetro.
- Enumeración. Si un algoritmo tiene un parámetro de enumeración, el valor de ese parámetro debe ingresarse usando un valor entero. Para conocer las opciones disponibles, puede usar el comando `algorithmHelp()`, como arriba. Por ejemplo, el algoritmo `native:buffer` tiene una enumeración llamada JOIN_STYLE:

  ```
  JOIN_STYLE: Join style

     Parameter type: QgsProcessingParameterEnum

     Available values:
             - 0: Round
             - 1: Miter
             - 2: Bevel

     Accepted data types:
             - int
             - str: as string representation of int, e.g. '1'
             - QgsProperty
  ```

  En este caso, el parámetro tiene tres opciones. Tenga en cuenta que el orden parte de cero.
- Boolean. Use `True` o `False`.
- La entrada múltiple. El valor es una cadena con descriptores de entrada separadas por punto y coma (`;`). Como en el caso de capas individuales o tablas, cada descriptor de entrada se puede el nombre del objeto de datos, o su ruta de archivo.
- El campo de la tabla de XXX. Utilice una cadena con el nombre del campo a usar. Este parámetro es sensible a mayúsculas y minúsculas.
- Tabla fija. Escribir la lista de todas las tablas de valores separadas por comas (`,`) y cerrar entre comillas (`"`). Los valores que empiezan en la fila superior y van de izquierda a derecha. También se puede utilizar un arreglo 2-D de valores que representen la tabla.
- SRC. Introduzca el número del código EPSG del SRC deseado.
- Extensión. Se debe utilizar una cadena con valores de `xmin`, `xmax`, `ymin` y `ymax` separados por comas (`,`).

Los parámetros boolean, archivo, cadena y numéricos no necesitan alguna explicación adicional.

Los parámetros de entrada como cadenas, booleanos o valores numéricos tienen valores predeterminados. El valor predeterminado se utiliza si falta la entrada de parámetro correspondiente.

Para los objetos de datos de salida, escriba la ruta del archivo que se utilizará para guardarlos, tal como se hace en la caja de herramientas. Si no se especifica el objeto de salida, el resultado se guarda en un archivo temporal (o se omite si es una salida opcional). La extensión del archivo determina el formato del archivo. Si ingresa una extensión de archivo no admitida por el algoritmo, se utilizará el formato de archivo predeterminado para ese tipo de salida y su extensión correspondiente se agregará a la ruta de archivo dada.

A diferencia de cuando se ejecuta un algoritmo desde la caja de herramientas, las salidas no se añaden al lienzo del mapa si ejecutas ese mismo algoritmo desde la consola de Python usando el método `run()`. Ese método devuelve un diccionario con uno o más nombres de salida (los que se muestran en la descripción del algoritmo) como claves y las rutas de los archivos de esas salidas como valores:

```
>>> myresult = processing.run("native:buffer", {'INPUT': '/data/lines.shp',
                'DISTANCE': 100.0,
                'SEGMENTS': 10,
                'DISSOLVE': True,
                'END_CAP_STYLE': 0,
                'JOIN_STYLE': 0,
                'MITER_LIMIT': 10,
                'OUTPUT': '/data/buffers.shp'})
>>> myresult['OUTPUT']
/data/buffers.shp
```

A continuación, puede cargar la salida en el proyecto como cualquier capa común:

```
>>> buffered_layer = myresult['OUTPUT']
>>> QgsProject.instance().addMapLayer(buffered_layer)
```

Para cargar inmediatamente los resultados del procesamiento en el proyecto, puede utilizar el método `runAndLoadResults()` en lugar de `run()`.

```
>>> processing.runAndLoadResults("native:buffer", {parameters:values})
```

Si desea abrir un diálogo de algoritmo desde la consola, puede usar el método `createAlgorithmDialog`. El único parámetro obligatorio es el nombre del algoritmo, pero también puedes definir el diccionario de parámetros para que el diálogo se llene automáticamente:

```
>>> my_dialog = processing.createAlgorithmDialog("native:buffer", {
                'INPUT': '/data/lines.shp',
                'DISTANCE': 100.0,
                'SEGMENTS': 10,
                'DISSOLVE': True,
                'END_CAP_STYLE': 0,
                'JOIN_STYLE': 0,
                'MITER_LIMIT': 10,
                'OUTPUT': '/data/buffers.shp'})
>>> my_dialog.show()
```

El método `execAlgorithmDialog` abre el diálogo inmediatamente:

```
>>> processing.execAlgorithmDialog("native:buffer", {
                'INPUT': '/data/lines.shp',
                'DISTANCE': 100.0,
                'SEGMENTS': 10,
                'DISSOLVE': True,
                'END_CAP_STYLE': 0,
                'JOIN_STYLE': 0,
                'MITER_LIMIT': 10,
                'OUTPUT': '/data/buffers.shp'})
```

23.7.2 Crear scripts y ejecurarlos desde le Caja de Herramientas

Puede crear sus propios algoritmos escribiendo código Python. Los scripts de procesamiento extienden `QgsProcessingAlgorithm`, por lo que necesita agregar algunas líneas adicionales de código para implementar funciones obligatorias. Puede encontrar *Create new script* (hoja limpia) y *Create New Script from Template* (plantilla que incluye código para funciones obligatorias de `QgsProcessingAlgorithm`) en el menú desplegable *Scripts* desplegable en la parte superior de la caja de herramientas de Procesamiento. Se abrirá Processing Script Editor, y ahí es donde debe escribir su código. Guardar el script desde allí en la carpeta `scripts` (la carpeta predeterminada cuando abre el cuadro de diálogo para guardar el archivo) con una extensión `.py` debería crear el algoritmo correspondiente.

El nombre del algoritmo (el que verá en la caja de herramientas) se define dentro del código.

Echemos un vistazo al siguiente código, que define un algoritmo de procesamiento que realiza una operación de búfer con una distancia de búfer definida por el usuario en una capa vectorial especificada por el usuario, después de suavizar primero la capa.

```
from qgis.core import (QgsProcessingAlgorithm,
       QgsProcessingParameterNumber,
       QgsProcessingParameterFeatureSource,
       QgsProcessingParameterFeatureSink)

from qgis import processing

class algTest(QgsProcessingAlgorithm):
    INPUT_BUFFERDIST = 'BUFFERDIST'
    OUTPUT_BUFFER = 'OUTPUT_BUFFER'
    INPUT_VECTOR = 'INPUT_VECTOR'

    def __init__(self):
        super().__init__()

    def name(self):
        return "algTest"

    def displayName(self):
        return "algTest script"

    def createInstance(self):
        return type(self)()

    def initAlgorithm(self, config=None):
        self.addParameter(QgsProcessingParameterFeatureSource(
            self.INPUT_VECTOR, "Input vector"))
        self.addParameter(QgsProcessingParameterNumber(
            self.INPUT_BUFFERDIST, "Buffer distance",
            QgsProcessingParameterNumber.Double,
            100.0))
        self.addParameter(QgsProcessingParameterFeatureSink(
            self.OUTPUT_BUFFER, "Output buffer"))

    def processAlgorithm(self, parameters, context, feedback):
        #DO SOMETHING
        algresult = processing.run("native:smoothgeometry",
            {'INPUT': parameters[self.INPUT_VECTOR],
             'ITERATIONS':2,
             'OFFSET':0.25,
             'MAX_ANGLE':180,
             'OUTPUT': 'memory:'},
            context=context, feedback=feedback, is_child_algorithm=True)
        smoothed = algresult['OUTPUT']
        algresult = processing.run('native:buffer',
```

(continúe en la próxima página)

(proviene de la página anterior)

```
            {'INPUT': smoothed,
            'DISTANCE': parameters[self.INPUT_BUFFERDIST],
            'SEGMENTS': 5,
            'END_CAP_STYLE': 0,
            'JOIN_STYLE': 0,
            'MITER_LIMIT': 10,
            'DISSOLVE': True,
            'OUTPUT': parameters[self.OUTPUT_BUFFER]},
            context=context, feedback=feedback, is_child_algorithm=True)
        buffered = algresult['OUTPUT']
        return {self.OUTPUT_BUFFER: buffered}
```

Después de hacer las importaciones necesarias, las siguientes funciones `QgsProcessingAlgorithm` son especificadas:

- `name()`: La id del algoritmo (minúsculas).
- `displayName()`: Un nombre de algoritmo legible por humanos
- `createInstance()`: Crear una nueva instancia de la clase del algoritmo.
- `initAlgorithm()`: Comfigurar las definiciones de Parámetro y Definiciones de Salida.

  Aquí describe los parámetros y la salida del algoritmo. En este caso, una fuente de objetos para la entrada, una salida de objetos para el resultado y un número para la distancia del búfer.

- `processAlgorithm()`: Hacer el trabajo.

  Aquí primero ejecutamos el algoritmo `smoothgeometry` para suavizar la geometría, y luego ejecutamos el algoritmo `buffer` en la salida suavizada. Para poder ejecutar algoritmos desde otro algoritmo, debemos establecer el argumento `is_child_algorithm` en `True`. Puede ver cómo los parámetros de entrada y salida se utilizan como parámetros para los algoritmos `smoothgeometry` y `buffer`.

Existen varios tipos de parámetros de entrada y salida. Puede consultar la lista completa en *Tipos de entrada y salida para algoritmos de procesamiento*.

El primer parámetro para los constructores es el nombre del parámetro y el segundo es la descripción del parámetro (para la interfaz de usuario). El resto de los parámetros del constructor son específicos del tipo de parámetro.

La entrada se puede convertir en clases QGIS usando las funciones `parameterAs` de `QgsProcessingAlgorithm`. Por ejemplo, para obtener el número proporcionado para la distancia del búfer como un doble:

```
self.parameterAsDouble(parameters, self.INPUT_BUFFERDIST, context)).
```

La función `processAlgorithm` debe devolver un diccionario que contenga valores para cada salida definida por el algoritmo. Esto permite el acceso a estas salidas de otros algoritmos, incluidos otros algoritmos contenidos dentro del mismo modelo.

Los algoritmos que se comportan bien deben definir y devolver tantas salidas como tenga sentido. Las salidas que no son objetos, como números y cadenas, son muy útiles cuando se ejecuta su algoritmo como parte de un modelo más grande, ya que estos valores se pueden usar como parámetros de entrada para algoritmos posteriores dentro del modelo. Considere agregar salidas numéricas para cosas como la cantidad de objetos procesados, la cantidad de objetos no válidos encontrados, la cantidad de objetos de salida, etc. ¡Cuantas más salidas devuelva, más útil se volverá su algoritmo!

Revisión

El objeto `feedback` pasado a `processAlgorithm()` debe usarse para la retroalimentación / interacción del usuario. Puede usar la función `setProgress()` del objeto `feedback` para actualizar la barra de progreso (0 a 100) para informar el usuario sobre el progreso del algoritmo. Esto es muy útil si su algoritmo tarda mucho en completarse.

El objeto `feedback` proporciona un método `isCanceled()` que debe ser monitoreado para permitir la cancelación del algoritmo por parte del usuario. El `pushInfo()` método de `feedback` se puede utilizar para enviar información al usuario, y :meth:`reportError( ) <qgis.core.QgsProcessingFeedback.reportError> `es útil para enviar errores no fatales a los usuarios.

Los algoritmos deben evitar el uso de otras formas de proporcionar retroalimentación a los usuarios, como declaraciones de impresión o registro en `QgsMessageLog`, y siempre deben usar el objeto de retroalimentación en su lugar. Esto permite un registro detallado para el algoritmo y también es seguro para subprocesos (lo cual es importante, dado que los algoritmos generalmente se ejecutan en un subproceso en segundo plano).

Manejo de errores

Si su algoritmo encuentra un error que le impide ejecutarse, como valores de entrada no válidos o alguna otra condición de la que no puede o no debe recuperarse, entonces debe generar un `QgsProcessingException`. P.Ejemplo

```
if feature['value'] < 20:
  raise QgsProcessingException('Invalid input value {}, must be >= 20'.
↪format(feature['value']))
```

Trate de evitar generar `QgsProcessingException` para errores no fatales (por ejemplo, cuando un objeto tiene una geometría nula) y, en su lugar, solo informe estos errores a través de `feedback.reportError()` y omita la función. Esto ayuda a que su algoritmo sea «compatible con el modelo», ya que evita detener la ejecución de un algoritmo completo cuando se encuentra un error no fatal.

Documentación de las secuencias de comandos

Como en el caso de los modelos, puede crear documentación adicional para sus scripts, para explicar qué hacen y cómo usarlos.

`QgsProcessingAlgorithm` proporciona las funciones `helpString()`, `shortHelpString()` y `helpUrl()` para ese propósito. Especificar / suplantar estos para proporcionar mas ayuda al usuario.

`shortDescription()` se utiliza en la información sobre herramientas al pasar el cursor sobre el algoritmo en la caja de herramientas.

23.7.3 Pre y post-ejecución de la secuencia de comandos hooks

Los scripts también se pueden utilizar como ganchos previos y posteriores a la ejecución que se ejecutan antes y después de la ejecución de un algoritmo, respectivamente. Esto se puede utilizar para automatizar tareas que deben realizarse cada vez que se ejecuta un algoritmo.

La sintaxis es idéntica a la que se ha explicado anteriormente, pero una variable global adicional llamada `alg` está disponible, lo que representa el algoritmo que acaba de ser (o está a punto de ser) ejecutado.

En el grupo *General* del cuadro de diálogo de opciones de procesamiento, encontrará dos entradas denominadas *Pre-execution script* y *Post-execution script* donde se pueden introducir los nombres de archivo de los scripts a ejecutar en cada caso.

23.8 Usar procesamiento desde la línea de comando

QGIS viene con una herramienta llamada `QGIS Processing Executor` que le permite ejecutar modelos y algoritmos de procesamiento (integrados o proporcionados por complementos) directamente desde la línea de comandos sin iniciar QGIS Escritorio.

Desde una herramienta de línea de comandos, ejecutar `qgis_process` y se debería obtener:

```
QGIS Processing Executor - 3.35.0-Master 'Master' (3.35.0-Master)
Usage: C:\OSGeo4W\apps\qgis-dev\bin\qgis_process.exe [--help] [--version] [--json]␣
↪[--verbose] [--no-python] [--skip-loading-plugins] [command] [algorithm id, path␣
↪to model file, or path to Python script] [parameters]

Options:

  --help or -h            Output the help
  --version or -v         Output all versions related to QGIS Process
  --json                  Output results as JSON objects
  --verbose               Output verbose logs
  --no-python             Disable Python support (results in faster startup)
  --skip-loading-plugins  Avoid loading enabled plugins (results in faster startup)

Available commands:

  plugins             list available and active plugins
  plugins enable      enables an installed plugin. The plugin name must be␣
↪specified, e.g. "plugins enable cartography_tools"
  plugins disable     disables an installed plugin. The plugin name must be␣
↪specified, e.g. "plugins disable cartography_tools"
  list                list all available processing algorithms
  help                show help for an algorithm. The algorithm id or a path to a␣
↪model file must be specified.
  run                 runs an algorithm. The algorithm id or a path to a model file␣
↪and parameter values must be specified.
                      Parameter values are specified after -- with PARAMETER=VALUE␣
↪syntax.
                      Ordered list values for a parameter can be created by␣
↪specifying the parameter multiple times,
                      e.g. --LAYERS=layer1.shp --LAYERS=layer2.shp
                      Alternatively, a '-' character in place of the parameters␣
↪argument indicates that the parameters should be read from STDIN as a JSON␣
↪object.
                      The JSON should be structured as a map containing at least␣
↪the "inputs" key specifying a map of input parameter values.
                      This implies the --json option for output as a JSON object.
                      If required, the ellipsoid to use for distance and area␣
↪calculations can be specified via the "--ELLIPSOID=name" argument.
                      If required, an existing QGIS project to use during the␣
↪algorithm execution can be specified via the "--PROJECT_PATH=path" argument.
                      When passing parameters as a JSON object from STDIN, these␣
↪extra arguments can be provided as an "ellipsoid" and a "project_path" key␣
↪respectively.
```

> **Nota**
>
> Sólo los complementos instalados que anuncian `hasProcessingProvider=yes` en su archivo `metadata.txt` son reconocidos y pueden ser activados o cargados por la herramienta qgis_process.

Consejo

Antes de llamar a qgis_process en un sistema sin gestor de ventanas (por ejemplo, un servidor headless), debe establecer:

```
export QT_QPA_PLATFORM=offscreen
```

El comando `list` puede ser usado para obtener una lista de proveedores y algoritmos disponibles.

```
qgis_process list
```

El comando `help` puede ser usado para obtener mas información sobre comandos o algoritmos.

```
qgis_process help qgis:regularpoints
```

El comando `run` puede ser usado para ejecutar un algoritmo o modelo. Especificar el nombre del algoritmo o una ruta a un modelo como primer parámetro.

```
qgis_process run native:buffer -- INPUT=source.shp DISTANCE=2 OUTPUT=buffered.shp
```

Cuando un parámetro acepte una lista de valores, establezca la misma variable múltiples veces.

```
qgis_process run native:mergevectorlayers -- LAYERS=input1.shp LAYERS=input2.shp␣
↪OUTPUT=merged.shp
```

Mientras se ejecuta un algoritmo se muestra una barra de retroalimentación basada en texto, y la operación puede cancelarse mediante `CTRL+C`.

El comando `run` también admite otros parámetros.

- `--json` formateará la salida estándar de una manera estructurada JSON.
- `--ellipsoid` establecerá el elipsoide en el especificado.
- `--distance_units` usará las unidades especificadas de distancia..
- `--area_units` usará las unidades de área especificadas.
- `--project_path` cargará el proeycto especificado para ejecutar el algoritmo.

qgis_process admite parámetros de entrada complejos, es decir, tipos de parámetros que se especifican a su vez como un objeto de tipo diccionario para algoritmos. Para indicar que los parámetros se especificarán a través de stdin, el comando qgis_process debe seguir el formato (con un − al final en lugar de la lista de argumentos habitual).

```
qgis_process run algorithmId -
```

El objeto JSON debe contener una clave «inputs», que es un mapa de los valores de los parámetros de entrada. Por ejemplo

```
echo "{'inputs': {'INPUT': 'my_shape.shp', 'DISTANCE': 5}}" | qgis_process run␣
↪native:buffer -
```

Además, en este objeto JSON se pueden incluir ajustes adicionales como las unidades de distancia, las unidades de área, el elipsoide y la ruta del proyecto:

```
{
 'ellipsoid': 'EPSG:7019',
 'distance_units': 'feet',
 'area_units': 'ha',
 'project_path': 'C:/temp/my_project.qgs'
 'inputs': {'DISTANCE': 5, 'SEGMENTS': 8 ... }
}
```

La especificación de parámetros de entrada a través de stdin implica automáticamente el formato de salida `JSON` para los resultados.

23.9 Escribir nuevos algoritmos de procesamiento como scripts de Python

Hay dos opciones para escribir algoritmos de procesamiento usando Python.

- *Extending* `QgsProcessingAlgorithm`
- *Using the @alg decorator*

Dentro de QGIS, puede usar *Crear nuevo script* en el menú :guilabel:` Scripts` en la parte superior de *Barra de Herramientas Procesos* para abrir *Editor de Scripst de Procesos* donde puede escribir tu codigo. Para simplificar la tarea, puede comenzar con una plantilla de script usando *Crear nuevo script desde plantilla* del mismo menú. Esto abre una plantilla que se extiende `QgsProcessingAlgorithm`.

Si guarda el script en la carpeta: file: *scripts* (la ubicación predeterminada) con la extensión `.py`, el algoritmo estará disponible en *Caja de Herramientas de Procesos*.

23.9.1 Extendiendo QgsProcessingAlgorithm

El siguiente código

1. toma una capa vectorial como entrada
2. cuenta el número de objetos
3. hace una operación de buffer
4. crea una capa ráster a partir del resultado de la operación buffer
5. devuelve la capa buffer, la capa ráster y el número de objetos

```
from qgis.PyQt.QtCore import QCoreApplication
from qgis.core import (QgsProcessing,
                       QgsProcessingAlgorithm,
                       QgsProcessingException,
                       QgsProcessingOutputNumber,
                       QgsProcessingParameterDistance,
                       QgsProcessingParameterFeatureSource,
                       QgsProcessingParameterVectorDestination,
                       QgsProcessingParameterRasterDestination)
from qgis import processing


class ExampleProcessingAlgorithm(QgsProcessingAlgorithm):
    """
    This is an example algorithm that takes a vector layer,
    creates some new layers and returns some results.
    """

    def tr(self, string):
        """
        Returns a translatable string with the self.tr() function.
        """
        return QCoreApplication.translate('Processing', string)

    def createInstance(self):
```

(continúe en la próxima página)

(proviene de la página anterior)

```
        # Must return a new copy of your algorithm.
        return ExampleProcessingAlgorithm()

    def name(self):
        """
        Returns the unique algorithm name.
        """
        return 'bufferrasterextend'

    def displayName(self):
        """
        Returns the translated algorithm name.
        """
        return self.tr('Buffer and export to raster (extend)')

    def group(self):
        """
        Returns the name of the group this algorithm belongs to.
        """
        return self.tr('Example scripts')

    def groupId(self):
        """
        Returns the unique ID of the group this algorithm belongs
        to.
        """
        return 'examplescripts'

    def shortHelpString(self):
        """
        Returns a localised short help string for the algorithm.
        """
        return self.tr('Example algorithm short description')

    def initAlgorithm(self, config=None):
        """
        Here we define the inputs and outputs of the algorithm.
        """
        # 'INPUT' is the recommended name for the main input
        # parameter.
        self.addParameter(
            QgsProcessingParameterFeatureSource(
                'INPUT',
                self.tr('Input vector layer'),
                types=[QgsProcessing.TypeVectorAnyGeometry]
            )
        )
        self.addParameter(
            QgsProcessingParameterVectorDestination(
                'BUFFER_OUTPUT',
                self.tr('Buffer output'),
            )
        )
        # 'OUTPUT' is the recommended name for the main output
        # parameter.
        self.addParameter(
            QgsProcessingParameterRasterDestination(
                'OUTPUT',
                self.tr('Raster output')
            )
        )
```

(continúe en la próxima página)

(proviene de la página anterior)

```
        self.addParameter(
            QgsProcessingParameterDistance(
                'BUFFERDIST',
                self.tr('BUFFERDIST'),
                defaultValue = 1.0,
                # Make distance units match the INPUT layer units:
                parentParameterName='INPUT'
            )
        )
        self.addParameter(
            QgsProcessingParameterDistance(
                'CELLSIZE',
                self.tr('CELLSIZE'),
                defaultValue = 10.0,
                parentParameterName='INPUT'
            )
        )
        self.addOutput(
            QgsProcessingOutputNumber(
                'NUMBEROFFEATURES',
                self.tr('Number of features processed')
            )
        )

    def processAlgorithm(self, parameters, context, feedback):
        """
        Here is where the processing itself takes place.
        """
        # First, we get the count of features from the INPUT layer.
        # This layer is defined as a QgsProcessingParameterFeatureSource
        # parameter, so it is retrieved by calling
        # self.parameterAsSource.
        input_featuresource = self.parameterAsSource(parameters,
                                                     'INPUT',
                                                     context)
        numfeatures = input_featuresource.featureCount()

        # Retrieve the buffer distance and raster cell size numeric
        # values. Since these are numeric values, they are retrieved
        # using self.parameterAsDouble.
        bufferdist = self.parameterAsDouble(parameters, 'BUFFERDIST',
                                            context)
        rastercellsize = self.parameterAsDouble(parameters, 'CELLSIZE',
                                                context)
        if feedback.isCanceled():
            return {}
        buffer_result = processing.run(
            'native:buffer',
            {
                # Here we pass on the original parameter values of INPUT
                # and BUFFER_OUTPUT to the buffer algorithm.
                'INPUT': parameters['INPUT'],
                'OUTPUT': parameters['BUFFER_OUTPUT'],
                'DISTANCE': bufferdist,
                'SEGMENTS': 10,
                'DISSOLVE': True,
                'END_CAP_STYLE': 0,
                'JOIN_STYLE': 0,
                'MITER_LIMIT': 10
            },
            # Because the buffer algorithm is being run as a step in
```

(continúe en la próxima página)

(proviene de la página anterior)

```
            # another larger algorithm, the is_child_algorithm option
            # should be set to True
            is_child_algorithm=True,
            #
            # It's important to pass on the context and feedback objects to
            # child algorithms, so that they can properly give feedback to
            # users and handle cancelation requests.
            context=context,
            feedback=feedback)

        # Check for cancelation
        if feedback.isCanceled():
            return {}

        # Run the separate rasterization algorithm using the buffer result
        # as an input.
        rasterized_result = processing.run(
            'qgis:rasterize',
            {
                # Here we pass the 'OUTPUT' value from the buffer's result
                # dictionary off to the rasterize child algorithm.
                'LAYER': buffer_result['OUTPUT'],
                'EXTENT': buffer_result['OUTPUT'],
                'MAP_UNITS_PER_PIXEL': rastercellsize,
                # Use the original parameter value.
                'OUTPUT': parameters['OUTPUT']
            },
            is_child_algorithm=True,
            context=context,
            feedback=feedback)

        if feedback.isCanceled():
            return {}

        # Return the results
        return {'OUTPUT': rasterized_result['OUTPUT'],
                'BUFFER_OUTPUT': buffer_result['OUTPUT'],
                'NUMBEROFFEATURES': numfeatures}
```

Funciones estándar del algoritmo de Porcesamiento:

- **createInstance (obligatorio)**
  Debe devolver una nueva copia de su algoritmo. Si cambia el nombre de la clase, asegúrese de actualizar también el valor devuelto aquí para que coincida.
- **nombre (obligatorio)**
  Devuelve el nombre exclusivo del algoritmo, que se utiliza para identificar el algoritmo.
- **displayName (obligatorio)**
  Devuelve el nombre del algoritmo traducido.
- **grupo**
  Devuelve el nombre del grupo al que pertenece este algoritmo.
- **groupId**
  Devuelve el ID único del grupo al que pertenece este algoritmo.
- **shortHelpString**
  Devuelve una cadena de ayuda corta localizada para el algoritmo.
- **initAlgorithm (obligatorio)**
  Aquí definimos las entradas y salidas del algoritmo.

  `INPUT` y `OUTPUT` son nombres recomendados para los parámetros de entrada y salida principal,

respectivamente.

Si un parámetro depende de otro parámetro, se usa `parentParameterName` para especificar esta relación (podría ser el campo / banda de una capa o las unidades de distancia de una capa).

- **processAlgorithm (obligatorio)**
  Aquí es donde tiene lugar el procesamiento.

  Los parámetros se recuperan mediante funciones especiales, por ejemplo, `parameterAsSource` y `parameterAsDouble`.

  `processing.run` se puede utilizar para ejecutar otros algoritmos de procesamiento desde un algoritmo de procesamiento. El primer parámetro es el nombre del algoritmo, el segundo es un diccionario de los parámetros del algoritmo. `Is_child_algorithm` normalmente se establece en `True` cuando se ejecuta un algoritmo desde dentro de otro algoritmo. El `contexto` y la `retroalimentación` informan al algoritmo sobre el entorno en el que se ejecutará y el canal para comunicarse con el usuario (capturando la solicitud de cancelación, reportando el progreso, proporcionando retroalimentación textual). Cuando se utilizan los parámetros del algoritmo (principal) como parámetros de los algoritmos «secundarios», se deben utilizar los valores de los parámetros originales (por ejemplo, `parámetros ['OUTPUT']`).

  ¡Es una buena práctica verificar el objeto de retroalimentación para la cancelación tanto como sea posible! Hacerlo permite una cancelación receptiva, en lugar de obligar a los usuarios a esperar a que ocurra un procesamiento no deseado.

  El algoritmo debe devolver valores para todos los parámetros de salida que ha definido como diccionario. En este caso, eso es el búfer y las capas de salida rasterizadas, y el recuento de entidades procesadas. Las claves del diccionario deben coincidir con los nombres de salida / parámetros originales.

23.9.2 El decorador @alg

Con el decorador @alg, puede crear sus propios algoritmos escribiendo el código Python y agregando algunas líneas adicionales para proporcionar la información adicional necesaria para convertirlo en un algoritmo de procesamiento adecuado. Esto simplifica la creación de algoritmos y la especificación de entradas y salidas.

Una limitación importante con el enfoque del decorador es que los algoritmos creados de esta manera siempre se agregarán al proveedor de Scripts de procesamiento de un usuario; no es posible agregar estos algoritmos a un proveedor personalizado, p. Ej. para su uso en complementos.

EL siguiente ejemplo usa el decorador @alg a

1. usa una capa vectorial como entrada
2. cuenta el número de objetos
3. hace una operación buffer
4. crea una capa ráster a partir del resultado de la operación buffer
5. devuelve la capa buffer, la capa ráster y el número de objetos

```
from qgis import processing
from qgis.processing import alg
from qgis.core import QgsProject

@alg(name='bufferrasteralg', label='Buffer and export to raster (alg)',
     group='examplescripts', group_label='Example scripts')
# 'INPUT' is the recommended name for the main input parameter
@alg.input(type=alg.SOURCE, name='INPUT', label='Input vector layer')
# 'OUTPUT' is the recommended name for the main output parameter
@alg.input(type=alg.RASTER_LAYER_DEST, name='OUTPUT',
           label='Raster output')
@alg.input(type=alg.VECTOR_LAYER_DEST, name='BUFFER_OUTPUT',
           label='Buffer output')
```

(continúe en la próxima página)

(proviene de la página anterior)

```
@alg.input(type=alg.DISTANCE, name='BUFFERDIST', label='BUFFER DISTANCE',
           default=1.0)
@alg.input(type=alg.DISTANCE, name='CELLSIZE', label='RASTER CELL SIZE',
           default=10.0)
@alg.output(type=alg.NUMBER, name='NUMBEROFFEATURES',
            label='Number of features processed')

def bufferrasteralg(instance, parameters, context, feedback, inputs):
    """
    Description of the algorithm.
    (If there is no comment here, you will get an error)
    """
    input_featuresource = instance.parameterAsSource(parameters,
                                                     'INPUT', context)
    numfeatures = input_featuresource.featureCount()
    bufferdist = instance.parameterAsDouble(parameters, 'BUFFERDIST',
                                            context)
    rastercellsize = instance.parameterAsDouble(parameters, 'CELLSIZE',
                                                context)
    if feedback.isCanceled():
        return {}
    buffer_result = processing.run('native:buffer',
                                   {'INPUT': parameters['INPUT'],
                                    'OUTPUT': parameters['BUFFER_OUTPUT'],
                                    'DISTANCE': bufferdist,
                                    'SEGMENTS': 10,
                                    'DISSOLVE': True,
                                    'END_CAP_STYLE': 0,
                                    'JOIN_STYLE': 0,
                                    'MITER_LIMIT': 10
                                    },
                                   is_child_algorithm=True,
                                   context=context,
                                   feedback=feedback)
    if feedback.isCanceled():
        return {}
    rasterized_result = processing.run('qgis:rasterize',
                                   {'LAYER': buffer_result['OUTPUT'],
                                    'EXTENT': buffer_result['OUTPUT'],
                                    'MAP_UNITS_PER_PIXEL': rastercellsize,
                                    'OUTPUT': parameters['OUTPUT']
                                   },
                                   is_child_algorithm=True, context=context,
                                   feedback=feedback)
    if feedback.isCanceled():
        return {}
    return {'OUTPUT': rasterized_result['OUTPUT'],
            'BUFFER_OUTPUT': buffer_result['OUTPUT'],
            'NUMBEROFFEATURES': numfeatures}
```

Como puede ver, involucra dos algoritmos (“native:buffer” y “qgis:rasterize”). El último (“qgis:rasterize”) crea una capa ráster a partir de la capa de búfer que fue generada por el primero (“native:buffer”).

La parte del código en la que tiene lugar este procesamiento no es difícil de entender si se ha leído el capítulo anterior. Las primeras líneas, sin embargo, necesitan alguna explicación adicional. Proporcionan la información que se necesita para convertir tu código en un algoritmo que pueda ejecutarse desde cualquiera de los componentes de la IGU, como la caja de herramientas o el diseñador de modelos.

Todas estas líneas son llamadas a las funciones del decorador `@alg` que ayudan a simplificar la codificación del algoritmo.

- El decorador @alg se utiliza para definir el nombre y la ubicación del algoritmo en la Caja de herramientas.

- El decorador @alg.input se utiliza para definir las entradas del algoritmo.
- El decorador @alg.output se utiliza para definir las salidas del algoritmo.

Para conocer los parámetros existentes y su correspondencia, lea *Tipos de entrada y salida para algoritmos de procesamiento*.

23.9.3 Manejo de salida de algoritmo

Cuando declara una salida que representa una capa (ráster o vector), el algoritmo intentará agregarla a QGIS una vez que haya terminado.

- Capa ráster saliente: QgsProcessingParameterRasterDestination / alg.RASTER_LAYER_DEST.
- Capa vectorial saliente: QgsProcessingParameterVectorDestination / alg.VECTOR_LAYER_DEST.

Entonces, incluso si el método `processing.run()` no agrega las capas que crea al proyecto actual del usuario, se cargarán las dos capas de salida (búfer y búfer ráster), ya que se guardan en los destinos ingresados por el usuario (o a destinos temporales si el usuario no especifica destinos).

Si se crea una capa como salida de un algoritmo, debe declararse como tal. De lo contrario, no podrá utilizar correctamente el algoritmo en el modelador, ya que lo que se declara no coincidirá con lo que realmente crea el algoritmo.

Puede devolver cadenas, números y más especificándolos en el diccionario de resultados (como se demuestra para «NUMBEROFFEATURES»), pero siempre deben definirse explícitamente como salidas de su algoritmo. Alentamos a los algoritmos a generar tantos valores útiles como sea posible, ya que estos pueden ser valiosos para su uso en algoritmos posteriores cuando su algoritmo se usa como parte de un modelo.

23.9.4 Comunicarse con el usuario

Si su algoritmo tarda mucho en procesarse, es una buena idea informar al usuario sobre el progreso. Puedes usar `feedback` (`QgsProcessingFeedback`) para esto.

El texto de progreso y la barra de progreso se pueden actualizar usando dos métodos: `setProgressText(text)` y `setProgress(percent)`.

Puede proporcionar más información utilizando `pushCommandInfo(text)`, `pushDebugInfo(text)`, `pushInfo(text)` and `reportError(text)`.

Si su guión tiene un problema, la forma correcta de manejarlo es generar una `QgsProcessingException`. Puede pasar un mensaje como argumento al constructor de la excepción. Procesos se encargará de manejarlo y comunicarse con el usuario, dependiendo de dónde se esté ejecutando el algoritmo (caja de herramientas, modelador, consola Python, …)

23.9.5 Documentación de las secuencias de comandos

Puede documentar sus scripts sobrecargando el `helpString()` and `helpUrl()` methods of `QgsProcessingAlgorithm`.

23.9.6 Banderas

Puede anular el método `flags()` de `QgsProcessingAlgorithm` para decirle a QGIS más sobre su algoritmo. Por ejemplo, puede decirle a QGIS que el script se ocultará al modelador, que se puede cancelar, que no es seguro para subprocesos y más.

Truco

De forma predeterminada, Procesos ejecuta algoritmos en un hilo separado para mantener la respuesta de QGIS mientras se ejecuta la tarea de procesamiento. Si su algoritmo falla regularmente, probablemente esté utilizando llamadas a la API que no son seguras de hacer en un hilo en segundo plano. Intente devolver el indicador QgsProcessingAlgorithm.FlagNoThreading del método flags() de su algoritmo para forzar a Processing a ejecutar su algoritmo en el hilo principal.

23.9.7 Las mejores practivas al escribir scripts de algoritmos

Aquí hay un resumen rápido de ideas a considerar al crear sus algoritmos de script y, especialmente, si desea compartirlos con otros usuarios de QGIS. Seguir estas sencillas reglas garantizará la coherencia entre los diferentes elementos de procesamiento, como la caja de herramientas, el modelador o la interfaz de procesamiento por lotes.

- No cargar la capa resultante. Deje al Procesamiento manejar sus resultados y cargue sus capas si es necesario.
- Siempre declare las salidas que crea su algoritmo.
- No muestre cuadros de mensaje ni use ningún elemento GUI del script. Si desea comunicarse con el usuario, utilice los métodos del objeto de comentarios (`QgsProcessingFeedback`) o lance una `QgsProcessingException`.

Ya hay muchos algoritmos de procesamiento disponibles en QGIS. Puedes encontrar código en el repositorio QGIS.

23.10 Tipos de entrada y salida para algoritmos de procesamiento

Aquí está la lista de tipos de entrada y salida que son soportados en Procesos con sus correspondientes constantes del decorador alg, su nombre en el diseñador del modelador si está expuesto.

23.10.1 Tipos de entrada

Nombre de entrada	Clase	Alg constante	Descripción
Capa de anotación	`QgsProcessingParame`	`alg.ANNOTATION_LAYE`	Una capa de anotación
Area	`QgsProcessingParame`		Parámetro numérico que representa una medida de superficie
Configuración de Autenticación	`QgsProcessingParame`	`alg.AUTH_CFG`	Permite a los usuarios seleccionar entre las configuraciones de autenticación disponibles o crear nuevas configuraciones de autenticación
Banda ráster	`QgsProcessingParame`	`alg.BAND`	Una banda de una capa ráster
Boolean	`QgsProcessingParame`	`alg.BOOL`	Un valor booleano

continúe en la próxima página

Tabla 23.1 – proviene de la página anterior

Nombre de entrada	Clase	Alg constante	Descripción
Nombre de la conexión	QgsProcessingParame		Una selección de las conexiones registradas disponibles para un proveedor de datos concreto
Color	QgsProcessingParame	alg.COLOR	Un color
Operación de coordenada	QgsProcessingParame	alg.COORDINATE_OPER	Una operación de coordenada (para transformaciones SRC)
CRS	QgsProcessingParame	alg.CRS	Un Sistema de Coordenadas de Referencia
Esquema de la base de datos	QgsProcessingParame	alg.DATABASE_SCHEMA	Un esquema de Base de Datos
Tabla de base de datos	QgsProcessingParame	alg.DATABASE_TABLE	Una tabla de Base de Datos
Fecha	ProcessingDateTimeP Date	alg.DATE	Una fecha pura
Fechahora	ProcessingDateTimeP DateTime	alg.DATETIME	Una fecha y hora
Distancia	QgsProcessingParame	alg.DISTANCE	Un parámetro numérico doble para valores de distancia
Capas DXF	QgsProcessingParame		Una lista de capas vectoriales para exportar como archivo DXF
Enum	QgsProcessingParame	alg.ENUM	Una enumeración, permitiendo la selección de un conjunto de valores predefinidos
Expresión	QgsProcessingParame	alg.EXPRESSION	Una expresión
Extensión	QgsProcessingParame	alg.EXTENT	Una extensión espacial definida por xmin, xmax, ymin, ymax
	QgsProcessingParame	alg.FIELD	Un campo en la tabla de atributos de una capa vectorial
Agregados de campo	QgsProcessingParame		Un parámetro para las configuraciones «agregadas», que consisten en una definición de los campos de salida deseados, los tipos y el agregado utilizado para rellenarlos.
Mapeador de campos	QgsProcessingParame		Un parámetro para las configuraciones de «mapeo de campos», que consisten en una definición de los campos de salida deseados, los tipos y las expresiones utilizadas para rellenarlos.
Archivo/Carpeta	QgsProcessingParame	alg.FILE	Un nombre de archivo de un archivo existente
	QgsProcessingParame	alg.FILE_DEST	Un nombre de archivo para un archivo de salida recién creado
	QgsProcessingParame	alg.FOLDER_DEST	Una carpeta (carpeta de destino)
Geometría	QgsProcessingParame	alg.GEOMETRY	Una geometría
	ProcessingNumberPar Integer	alg.INT	Un entero
Diseño de impresión	QgsProcessingParame	alg.LAYOUT	Un diseño
Elemento de diseño de impresión	QgsProcessingParame	alg.LAYOUT_ITEM	Un elemento de diseño
Capa de mapa	QgsProcessingParame	alg.MAPLAYER	Una capa de mapa

continúe en la próxima página

Tabla 23.1 – proviene de la página anterior

Nombre de entrada	Clase	Alg constante	Descripción
Tema del mapa	QgsProcessingParame	alg.MAP_THEME	Un tema de mapa de proyecto
Matriz	QgsProcessingParame	alg.MATRIX	Una matriz
Grupos de conjuntos de datos de malla	QgsProcessingParame		Grupos de datos de la capa de malla
Malla Conjunto de datos Tiempo	QgsProcessingParame		Índice del conjunto de datos de una capa de malla por ajuste temporal
Capa de malla	QgsProcessingParame	alg. MESH_LAYER	Una capa de malla
Entrada múltiple	QgsProcessingParame	alg. MULTILAYER	Un conjunto de capas
NUmero	ProcessingNumberPar Double	alg.NUMBER	Un valor numérico
Punto	QgsProcessingParame	alg.POINT	Un punto
Atributo de la nube de puntos	QgsProcessingParame		Un atributo de una capa de nube de puntos
	QgsProcessingParame	alg. POINTCLOUD_LAYE	Un parámetro de destino de la capa de nube de puntos, para especificar la ruta de destino de una capa de nube de puntos creada por el algoritmo.
Capa de nubes de puntos	QgsProcessingParame	alg. POINTCLOUD_LAYE	Una capa de nube de puntos
	QgsProcessingParame	alg. PROVIDER_CONNEC	Una conexión disponible para un proveedor de base de datos
Rango	QgsProcessingParame	alg.RANGE	Un rango de número
Capa Ráster	QgsProcessingParame	alg. RASTER_LAYER	Una capa ráster.
	QgsProcessingParame	alg. RASTER_LAYER_DE	Un parámetro de destino de la capa ráster, para especificar la ruta de destino de una capa ráster creada por el algoritmo.
Escala	QgsProcessingParame	alg.SCALE	Una escala de mapa
	QgsProcessingParame	alg.SINK	Un destino de objetos
Cadena	QgsProcessingParame	alg.STRING	Una cadena de texto
Capas de creación de TIN	QgsProcessingParame		Selección de varias capas para crear un TIN con vértices y/o líneas de rotura
Tiempo	ProcessingDateTimeP Time	alg.TIME	Un tiempo puro
Objetos espaciales vectoriales	QgsProcessingParame	alg.SOURCE	Una fuente de objetos
Capa vectorial	QgsProcessingParame	alg. VECTOR_LAYER	Una capa vectorial.
	QgsProcessingParame	alg. VECTOR_LAYER_DE	Un parámetro de destino de la capa vectorial, para especificar la ruta de destino de una capa vectorial creada por el algoritmo.
Capas del escritor de teselas vectoriales	QgsProcessingParame		Un parámetro de destino de la capa de mosaicos vectoriales, para especificar la ruta de destino para una capa de mosaicos vectoriales creada por el algoritmo.

continúe en la próxima página

Tabla 23.1 – proviene de la página anterior

Nombre de entrada	Clase	Alg constante	Descripción
Volumen	QgsProcessingParame		Parámetro numérico que representa una medida de volumen

23.10.2 Tipos de salida

Clase	Alg constante	Descripción
QgsProcessingOutputBoolean	alg.BOOL	Un valor booleano
QgsProcessingOutputNumber	alg.DISTANCE	Un parámetro numérico doble para valores de distancia
QgsProcessingOutputFile	alg.FILE	Un nombre de archivo de un archivo existente
QgsProcessingOutputFolder	alg.FOLDER	Una carpeta
QgsProcessingOutputHtml	alg.HTML	HTML
QgsProcessingOutputNumber	alg.INT	Un Entero
QgsProcessingOutputLayerDefinition	alg.LAYERDEF	Una definición de capa
QgsProcessingOutputMapLayer	alg.MAPLAYER	Una capa de mapa
QgsProcessingOutputMultipleLayers	alg.MULTILAYER	Un conjunto de capas
QgsProcessingOutputNumber	alg.NUMBER	Un valor numérico
QgsProcessingOutputPointCloudLayer	alg.POINTCLOUD_LAYER	Una capa de nube de puntos
QgsProcessingOutputRasterLayer	alg.RASTER_LAYER	Una capa ráster.
QgsProcessingOutputString	alg.STRING	Una cadena de texto
QgsProcessingOutputVectorLayer	alg.VECTOR_LAYER	Una capa vectorial.
QgsProcessingOutputVectorTileLayer		Una capa de teselas vectoriales

23.11 Configurar aplicaciones externas

The Processing framework can be extended using additional applications. Algorithms that rely on external applications are managed by their own algorithm providers. Additional providers can be found as separate plugins, and installed using the QGIS Plugin Manager.

Esta sección le mostrará cómo configurar el marco de trabajo de Procesos para incluir estas aplicaciones adicionales, y le explicará algunas características particulares de los algoritmos basados en ellas. Una vez que haya configurado correctamente el sistema, podrá ejecutar algoritmos externos desde cualquier componente como la caja de herramientas o el diseñador de modelos, igual que hace con cualquier otro algoritmo.

De forma predeterminada, los algoritmos que dependen de una aplicación externa que no se envía con QGIS no están habilitados. Puede habilitarlos en el cuadro de diálogo Configuración de procesamiento si están instalados en su sistema.

23.11.1 Aclaración para los usuarios de Windows

If you are not an advanced user and you are running QGIS on Windows, you might not be interested in reading the rest of this chapter. Make sure you install QGIS in your system using the standalone installer. That will automatically install GRASS in your system and configure it so it can be run from QGIS. All the algorithms from this provider will be ready to be run without needing any further configuration. If installing with the OSGeo4W application, make sure that you also select GRASS for installation.

23.11.2 Aclaración respecto a los formatos de archivos

Cuando se usa un software externo, abrir un archivo en QGIS no significa que se pueda abrir y procesar en ese otro software. En la mayoría de los casos, otro software puede leer lo que ha abierto en QGIS, pero en algunos casos, puede que eso no sea cierto. Cuando se utilizan bases de datos o formatos de archivo poco comunes, ya sea para capas ráster o vectoriales, pueden surgir problemas. Si eso sucede, intente usar formatos de archivo conocidos que esté seguro de que ambos programas entienden y verifique la salida de la consola (en el panel de registro) para averiguar qué está fallando.

Por ejemplo, es posible que tenga problemas y no pueda completar su trabajo si llama a un algoritmo externo con capas ráster de GRASS como entrada. Por esta razón, dichas capas no aparecerán como disponibles para los algoritmos.

Sin embargo, no debería tener problemas con las capas vectoriales, ya que QGIS convierte automáticamente del formato de archivo original a uno aceptado por la aplicación externa antes de pasarle la capa. Esto agrega tiempo de procesamiento adicional, que puede ser significativo para capas grandes, por lo que no se sorprenda si se necesita más tiempo para procesar una capa de una conexión de base de datos que una capa de un dataset en formato Shapefile de tamaño similar.

Los proveedores que no utilizan aplicaciones externas pueden procesar cualquier capa que pueda abrir en QGIS, ya que la abren para su análisis a través de QGIS.

Todos los formatos de salida ráster y vectoriales producidos por QGIS pueden usarse como capas de entrada. Algunos proveedores no admiten ciertos formatos, pero todos pueden exportar a formatos comunes que luego pueden ser transformados automáticamente por QGIS. En cuanto a las capas de entrada, si necesitan una conversión, eso podría aumentar el tiempo de procesamiento.

23.11.3 Nota referente a las seleccion de capas vectoriales

Las aplicaciones externas pueden también ser conscientes de las selecciones que existen en capas vectoriales dentro |qgi|. Sin embargo, eso requiere reescribir todas las capas vectoriales de entrada, al igual que si originalmente estuvieran en un formato no compatible con la aplicación externa. Sólo cuando no existe ninguna selección o la opción *Utilizar sólo objetos espaciales seleccionados * no está activada en la configuración general de procesamiento, puede una capa ser directamente pasada a una aplicación externa.

En otros casos, es necesario exportar solo las características seleccionadas, lo que provoca tiempos de ejecución más largos.

23.11.4 Uso de proveedores externos

Scripts y bibliotecas de R

Para habilitar R en Procesamiento, necesita instalar el complemento **Processing R Provider** y configurar R para QGIS.

La configuración se realiza en *Proveedor* ► *R* en la pestaña *Procesos* de *Configuración* ► *Opciones*.

Dependiendo de su sistema operativo, es posible que deba usar *Carpeta R* para especificar dónde se encuentran sus binarios R.

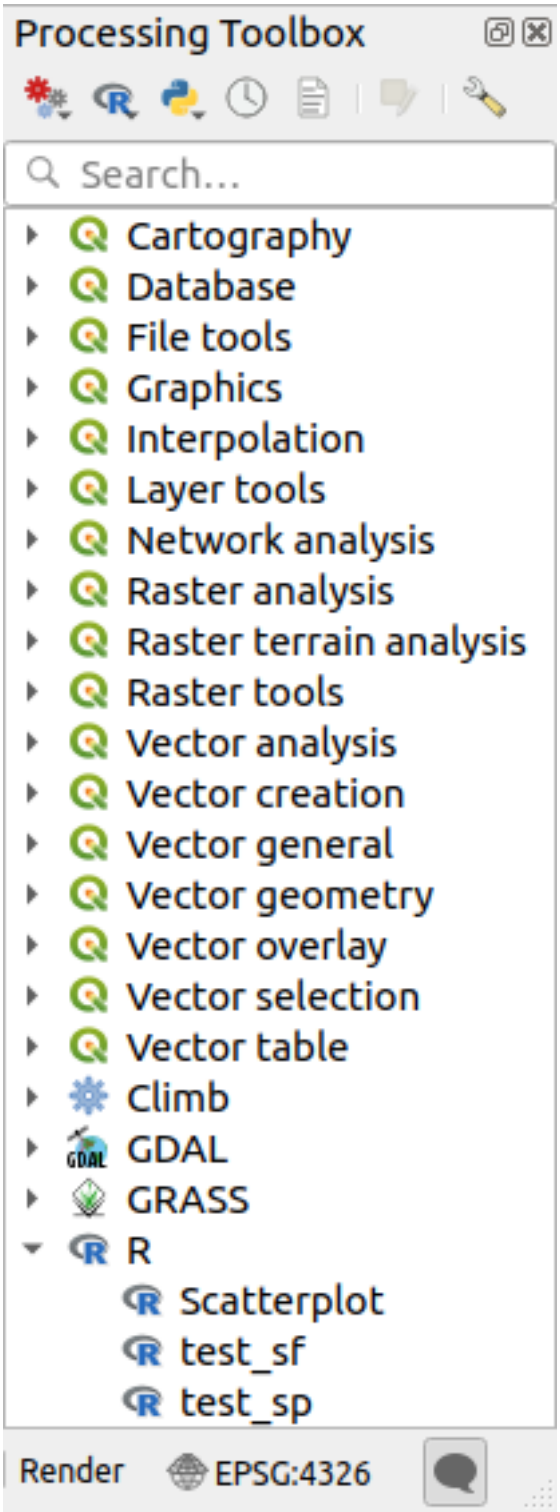
Nota

En **Windows** el archivo R ejecutable está normalmente en una carpeta (`R-<version>`) en `C:\Program Files\R\`. Especifique la carpeta y ** NO ** el binario!

En **Linux**, solo debe asegurarse de que la carpeta R esté en la variable de entorno PATH. Si `R` en una ventana de terminal inicia R, entonces está listo para comenzar.

Después de instalar el complemento **Processing R Provider**, encontrará algunos scripts de ejemplo en *Caja de Herramientas de Procesos*:

- *Scatterplot* ejecuta una función R que produce un gráfico de dispersión a partir de dos campos numéricos de la capa vectorial proporcionada.
- *test_sf* hace algunas operaciones que dependen del paquete `sf` y se puede usar para comprobar que el paquete `sf` de R está instalado. Si el paquete no está instalado, R intentará instalarlo (y todos los paquetes que dependan) por usted, utilizando el *Repositorio de Paquetes* especificado en *Proveedor* ► *R* en la opciones de Procesado. De forma predeterminada es https://cran.r-project.org/. La instalación puede llevar un tiempo…
- *test_sp* se puede utilizar para comprobar si el paquete de R `sp` está instalado. Si el paquete no está instalado, R intentará instalarlo por usted.



Si tiene R configurado correctamente para QGIS, debería poder ejecutar estos scripts.

Agregar scripts de R de la colección QGIS

R integration in QGIS is different from that of some other third party providers like SAGA in that there is not a predefined set of algorithms you can run (except for some example script that come with the *Processing R Provider* plugin).

Un conjunto de scripts de R de ejemplo está disponible en el repositorio de QGIS. Realice los siguientes pasos para cargarlos y habilitarlos usando el complemento *QGIS Resource Sharing*.

1. Agregue el complemento *QGIS Resource Sharing* (es posible que deba habilitar *Mostrar también complementos experimentales* en el Administrador de complementos *Configuración*)
2. Ábralo (*Complementos* ► *Compartir recursos* ► *Compartir recursos*)
3. Escoge la pestaña *Configuración*
4. Click *Recargar repositorios*
5. Elige la pestaña *Todos*
6. Seleccione *QGIS R script collection* en la lista y haga clic en el botón *Instalar*
7. La colección debería aparecer ahora en la pestaña *instalado*
8. Cierre el complemento
9. Abra *Caja de Herramientas de Procesamiento*, y si todo está bien, los scripts de ejemplo estarán presentes en R, en varios grupos (solo algunos de los grupos se expanden en la captura de pantalla a continuación).

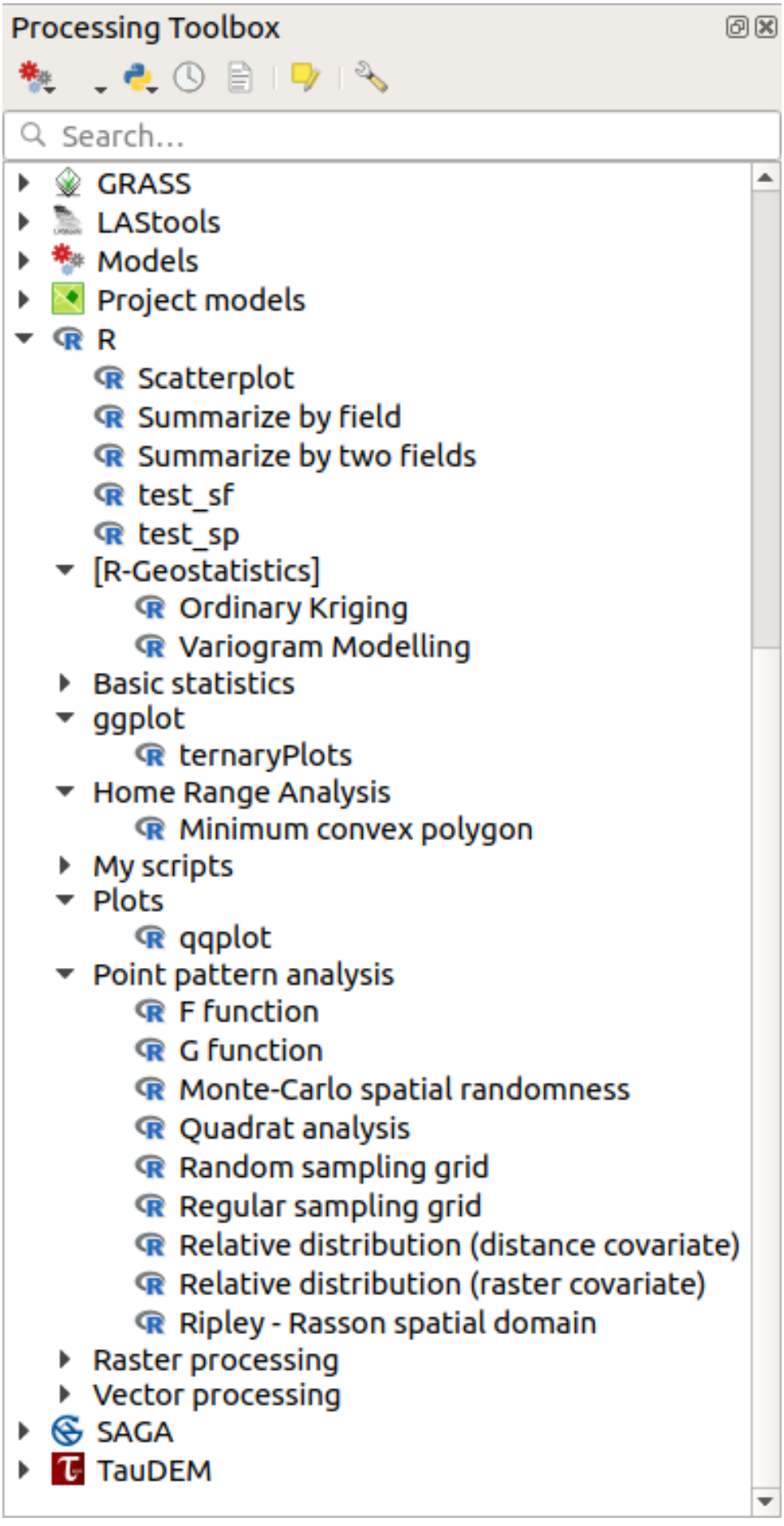
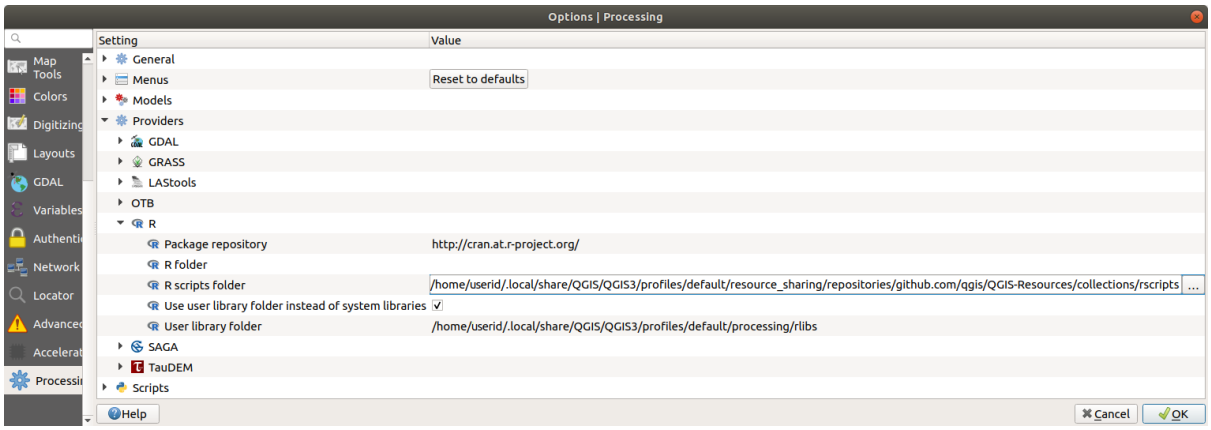


Figura 23.35: La *Caja de Herramientas de Procesos* con algunos scripts R mostrados

Los scripts en la parte superior son los scripts de ejemplo del complemento *Processing R Provider*.

10. Si, por alguna razón, los scripts no están disponibles en *Caja de Herramientas de Procesos*, puede intentar:
    1. Abra la configuración de Procesos (pestaña *Configuración* ► *Opciones* ► *Procesis*)
    2. Ve a *Providers* ► *R* ► *R scripts folder*
       - En Ubuntu, establezca la ruta en (o, mejor, incluya en la ruta):

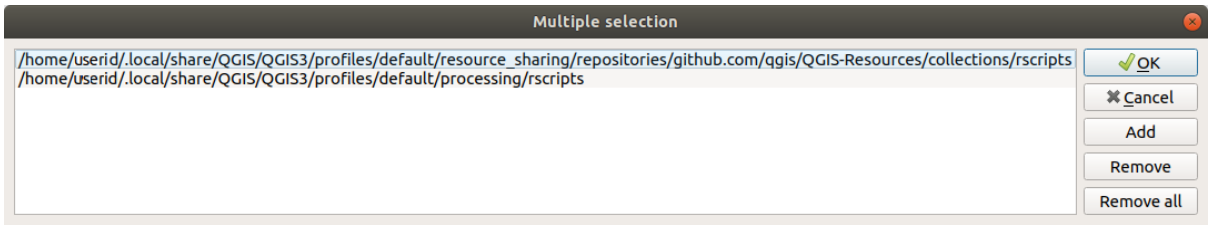
         /home/<user>/.local/share/QGIS/QGIS3/profiles/default/resource_sharing/repositories/github.com/qgis/QGIS-Resources/collections/rscripts



       - En Windows, establezca la ruta en (o, mejor, incluya en la ruta):

         C:\Users\<user>\AppData\Roaming\QGIS\QGIS3\profiles\default\resource_sharing\repositories\github.com\qgis\QGIS-Resources\collections\rscripts

       Para editar, haga doble clic. Luego puede elegir simplemente pegar/escribir la ruta, o puede navegar al directorio usando el botón … y presionar el botón :guilabel:‘Añadir` en el diálogo que se abre. Es posible proporcionar varios directorios aquí. Estarán separados por un punto y coma («;»).



Si desea obtener todos los scripts R de la colección en línea de QGIS 2, puede seleccionar *Colección de scripts QGIS R (de QGIS 2)* en lugar de *Colección de scripts QGIS R*. Probablemente encontrará que los scripts que dependen de la entrada o salida de datos vectoriales no funcionarán.

Creando scripts R

Puede escribir scripts y llamar comandos R, como lo haría desde R. Esta sección le muestra la sintaxis para usar comandos R en QGIS y cómo usar objetos QGIS (capas, tablas) en ellos.

Para agregar un algoritmo que llame a una función R (o un script R más complejo que haya desarrollado y le gustaría tener disponible en QGIS), debe crear un archivo de script que ejecute los comandos R.

Los archivos de script R tienen la extensión `.rsx`, y crearlos es bastante fácil si solo tiene un conocimiento básico de la sintaxis R y el scripting R. Deben almacenarse en la carpeta de scripts de R. Puede especificar la carpeta (*Carpeta de scripts R*) en el grupo de configuración *R* en el cuadro de diálogo Configuración de procesamiento).

Echemos un vistazo a un archivo de script muy simple, que llama al método R `spsample` para crear una cuadrícula aleatoria dentro del límite de los polígonos en una capa de polígono determinada. Este método pertenece al paquete `maptools`. Dado que casi todos los algoritmos que le gustaría incorporar en QGIS usarán o generarán datos espaciales, el conocimiento de paquetes espaciales como `maptools` y `sp`/`sf` es muy útil.

```
##Random points within layer extent=name
##Point pattern analysis=group
##Vector_layer=vector
##Number_of_points=number 10
##Output=output vector
library(sp)
spatpoly = as(Vector_layer, "Spatial")
pts=spsample(spatpoly,Number_of_points,type="random")
spdf=SpatialPointsDataFrame(pts, as.data.frame(pts))
Output=st_as_sf(spdf)
```

Las primeras líneas, que comienzan con un doble signo de comentario de Python (`##`), definen el nombre para mostrar y el grupo del script, e informan a QGIS sobre sus entradas y salidas.

Nota

Para obtener más información sobre cómo escribir sus propios scripts R, eche un vistazo a la sección R Intro en el manual de capacitación y consulte la sección *QGIS R Syntax*.

Cuando declara un parámetro de entrada, QGIS usa esa información para dos cosas: crear la interfaz de usuario para pedirle al usuario el valor de ese parámetro y crear una variable R correspondiente que se puede usar como entrada de función R.

En el ejemplo anterior, hemos declarado una entrada de tipo `vector`, denominada `Vector_layer`. Al ejecutar el algoritmo, QGIS abrirá la capa seleccionada por el usuario y la almacenará en una variable llamada `Vector_layer`. Entonces, el nombre de un parámetro es el nombre de la variable que usa en R para acceder al valor de ese parámetro (por lo tanto, debe evitar usar palabras R reservadas como nombres de parámetros).

Los parámetros espaciales, como capas vectoriales y ráster, se leen utilizando los comandos `st_read()` (o `readOGR`) y `brick()` (o `readGDAL`) (no es necesario preocúpese por agregar esos comandos a su archivo de descripción, QGIS lo hará), y se almacenan como objetos `sf` (o `Spatial * DataFrame`).

Los campos de la tabla se almacenan como cadenas que contienen el nombre del campo seleccionado.

Los archivos vectoriales se pueden leer usando el comando `readOGR()` en lugar de `st_read()` especificando `##load_vector_using_rgdal`. Esto producirá un objeto `Spatial*DataFrame` en lugar de un objeto `sf`.

Los archivos ráster se pueden leer usando el comando `readGDAL()` en lugar de `brick()` especificando `##load_raster_using_rgdal`.

Si es un usuario avanzado y no desea que QGIS cree el objeto para la capa, puede usar `##pass_filenames` para indicar que prefiere una cadena con el nombre del archivo. En este caso, depende de usted abrir el archivo antes de realizar cualquier operación en los datos que contiene.

Con la información anterior, es posible comprender las primeras líneas del script R (la primera línea no comienza con un carácter de comentario de Python).

```
library(sp)
spatpoly = as(Vector_layer, "Spatial")
pts=spsample(polyg,numpoints,type="random")
```

La función `spsample` es proporcionada por la biblioteca *sp*, por lo que lo primero que hacemos es cargar esa biblioteca. La variable `Vector_layer` contiene un objeto `sf`. Ya que vamos a usar una función (`spsample`) de la biblioteca *sp*, debemos convertir el objeto `sf` en un objeto `SpatialPolygonsDataFrame` usando la función `as`.

Luego llamamos a la función `spsample` con este objeto y el parámetro de entrada `numpoints` (que especifica el número de puntos a generar).

Ya que hemos declarado una salida de vector llamada `Salida`, tenemos que crear una variable llamada `Salida` que contenga un objeto `sf`.

Hacemos esto en dos pasos. Primero creamos un objeto `SpatialPolygonsDataFrame` a partir del resultado de la función, usando la función *SpatialPointsDataFrame*, y luego convertimos ese objeto en un objeto `sf` usando la función `st_as_sf` (de la biblioteca *sf*).

Puede utilizar los nombres que desee para sus variables intermedias. Solo asegúrese de que la variable que almacena el resultado final tenga el nombre definido (en este caso, `Salida`) y que contenga un valor adecuado (un objeto `sf` para la salida de la capa vectorial).

En este caso, el resultado obtenido del método `spsample` tuvo que convertirse explícitamente en un objeto `sf` a través de un objeto `SpatialPointsDataFrame`, ya que él mismo es un objeto de la clase `ppp` , que no se puede devolver a QGIS.

Si su algoritmo genera capas ráster, la forma en que se guarden dependerá de si ha utilizado o no la opción `##dontuserasterpackage`. Si lo ha utilizado, las capas se guardan utilizando el método `writeGDAL()`. De lo contrario, se utilizará el método `writeRaster()` del paquete `raster`.

Si ha utilizado la opción `##pass_filenames`, las salidas son generadas usando el paquete `ráster` (con `writeRaster()`).

Si su algoritmo no genera una capa, sino un resultado de texto en la consola, debe indicar que desea que la consola se muestre una vez finalizada la ejecución. Para hacerlo, simplemente inicie las líneas de comando que producen los resultados que desea imprimir con el signo `> `` ('mayor que'). Solo se muestran los resultados de las líneas con el prefijo ``>`. Por ejemplo, aquí está el archivo de descripción de un algoritmo que realiza una prueba de normalidad en un campo (columna) dado de los atributos de una capa vectorial:

```
##layer=vector
##field=field layer
##nortest=group
library(nortest)
>lillie.test(layer[[field]])
```

La salida de la última línea se imprime, pero la salida de la primera no (y tampoco están las salida de otras líneas de comando agregadas automáticamente por QGIS).

Si su algoritmo crea algún tipo de gráfico (utilizando el método `plot()`), añada la siguiente línea (`output_plots_to_html` solía ser `showplots`):

```
##output_plots_to_html
```

Esto provocará que QGIS redireccione todas las salidas de gráficas R a un archivo temporal, que se abrirá una vez que terminé la ejecución de R.

Tanto los gráficos como los resultados de la consola estarán disponibles a través del administrador de resultados de procesamiento.

Para más información, por favor consulte los scripts R en la colección oficial de QGIS (se descargan e instalan usando el complemento *QGIS Resource Sharing*, como se explica en *Agregar scripts de R de la colección QGIS*). La mayoría de ellos son bastante sencillos y le ayudarán enormemente a entender cómo crear sus propios scripts.

Nota

Las bibliotecas `sf`, `rgdal` y `raster` se cargan por defecto, por lo que no es necesario agregar los comandos correspondientes de `biblioteca()`. Sin embargo, otras bibliotecas que pueda necesitar deben cargarse explícitamente escribiendo: `library(ggplot2) '' (para cargar la biblioteca ``ggplot2)`. Si el paquete aún no está instalado en su máquina, Procesos intentará descargarlo e instalarlo.

De esta forma, el paquete también estará disponible en R Standalone. **Tenga en cuenta** que si el paquete debe descargarse, el script puede tardar mucho en ejecutarse la primera vez.

Bibliotecas de R instaladas al ejecutar sf_test

El script R `sp_test` intenta cargar los paquetes R `sp` y `raster`.

El script R *sf_test* intenta cargar `sf` y `raster`. Si estos dos paquetes no están instalados, R puede intentar cargarlos e instalarlos (y todas las bibliotecas de las que dependen).

Las siguientes bibliotecas de R terminan en `~/.local/share/QGIS/QGIS3/profiles/default/processing/rscripts` después de que se haya ejecutado `sf_test` desde la caja de herramientas de Procesos en Ubuntu con la versión 2.0 del *Plugin Processing R Provider* y una nueva instalación de *R* 3.4.4 (solo paquete *apt* `r-base-core`):

```
abind, askpass, assertthat, backports, base64enc, BH, bit, bit64, blob, brew,␣
↪callr, classInt, cli, colorspace, covr, crayon, crosstalk, curl, DBI, deldir,
desc, dichromat, digest, dplyr, e1071, ellipsis, evaluate, fansi, farver, fastmap,␣
↪gdtools, ggplot2, glue, goftest, gridExtra, gtable, highr, hms,
htmltools, htmlwidgets, httpuv, httr, jsonlite, knitr, labeling, later, lazyeval,␣
↪leafem, leaflet, leaflet.providers, leafpop, leafsync, lifecycle, lwgeom,
magrittr, maps, mapview, markdown, memoise, microbenchmark, mime, munsell, odbc,␣
↪openssl, pillar, pkgbuild, pkgconfig, pkgload, plogr, plyr, png, polyclip,
praise, prettyunits, processx, promises, ps, purrr, R6, raster, RColorBrewer, Rcpp,
↪ reshape2, rex, rgeos, rlang, rmarkdown, RPostgres, RPostgreSQL,
rprojroot, RSQLite, rstudioapi, satellite, scales, sf, shiny, sourcetools, sp,␣
↪spatstat, spatstat.data, spatstat.utils, stars, stringi, stringr, svglite,
sys, systemfonts, tensor, testthat, tibble, tidyselect, tinytex, units, utf8, uuid,
↪ vctrs, viridis, viridisLite, webshot, withr, xfun, XML, xtable
```

GRASS

Configuring GRASS is very easy. First, the path to the GRASS folder has to be defined, but only if you are running Windows.

De forma predeterminada, el marco de procesamiento intenta configurar su conector GRASS para usar la distribución GRASS que se envía junto con QGIS. Esto debería funcionar sin problemas para la mayoría de los sistemas, pero si tiene problemas, es posible que deba configurar el conector GRASS manualmente. Además, si desea utilizar una instalación de GRASS diferente, puede cambiar la configuración para que apunte a la carpeta donde está instalada la otra versión. GRASS 7 es necesario para que los algoritmos funcionen correctamente.

Si está ejecutando Linux, solo debe asegurarse de que GRASS esté instalado correctamente y de que se pueda ejecutar sin problemas desde una ventana de terminal.

GRASS algorithms use a region for calculations. This region can be defined manually or automatically, taking the minimum extent that covers all the input layers used to execute the algorithm each time.

LAStools

Para utilizar LAStools en QGIS, debe descargar e instalar LAStools en su ordenador e instalar el complemento LAStools (disponible en el repositorio oficial) en QGIS.

En plataformas Linux, necesitarás Wine para poder ejecutar algunas de las herramientas.

LAStools está activado y configurado en las opciones de Procesamiento (*Configuración* ► *Opciones*, pestaña *Procesos*, *Proveedores* ► *LAStools*), donde puede especificar la ubicación de LAStools (*LAStools folder*) y Wine (*Wine folder*). En Ubuntu, la carpeta Wine predeterminada es `/usr/bin`.

Aplicaciones OTB

Las aplicaciones OTB son totalmente soportadas dentro del espacio de trabajo de Procesamiento de QGIS.

OTB (Orfeo ToolBox) es una biblioteca de procesamiento de imágenes para datos de teledetección. También proporciona aplicaciones que ofrecen funciones de procesamiento de imágenes. La lista de aplicaciones y su documentación están disponibles en OTB CookBook

> **Nota**
>
> Tenga en cuenta que OTB no se distribuye con QGIS y debe instalarse por separado. Los paquetes binarios para OTB se pueden encontrar en la página de descarga.

Para configurar el procesamiento de QGIS para encontrar la biblioteca OTB:

1. Abra la configuración de procesamiento: *Configuración* ► *Opciones* ► *Procesos*.
2. Puede ver `OTB` en el menú *Proveedores*:
    1. Ampliar la entrada *OTB*
    2. Configure el *OTB folder*. Esta es la ubicación de su instalación OTB.
    3. Configure la *OTB application folder*. Esta es la ubicación de sus aplicaciones OTB (`<PATH_TO_OTB_INSTALLATION>/lib/otb/applications`)
    4. Haga clic en *Aceptar* para guardar la configuración y cerrar el cuadro de diálogo.

Si la configuración es correcta, los algoritmos OTB estarán disponibles en *Caja de Herramientas de Procesos*.

Documentación de la configuración de OTB disponible en Procesos QGIS

- **OTB folder**: Este es el directorio donde está disponible OTB.
- **OTB application folder**: Esta es la ubicación(es) de las aplicaciones OTB.

  Son permitidas las rutas múltiples.
- **Logger level** (opcional): Nivel de registrador que utilizarán las aplicaciones OTB.

  El nivel de registro controla la cantidad de detalles impresos durante la ejecución del algoritmo. Los valores posibles para el nivel del registrador son `INFO`, `WARNING`, `CRITICAL`, `DEBUG`. Este valor es `INFO` por defecto. Esta es una configuración de usuario avanzada.
- **Maximum RAM to use** (opcional): de forma predeterminada, las aplicaciones OTB utilizan toda la RAM disponible del sistema.

  Sin embargo, puede indicarle a OTB que use una cantidad específica de RAM (en MB) usando esta opción. El proveedor de procesamiento OTB ignora un valor de 256. Esta es una configuración de usuario avanzada.

- **Geoid file** (opcional): Ruta al archivo geoide.

  Esta opción establece el valor de los parámetros elev.dem.geoid y elev.geoid en aplicaciones OTB. Establecer este valor globalmente permite a los usuarios compartirlo a través de múltiples algoritmos de procesamiento. Vacío por defecto.

- **SRTM tiles folder** (opcional): directorio donde están disponibles los mosaicos SRTM.

  Los datos SRTM se pueden almacenar localmente para evitar la descarga de archivos durante el procesamiento. Esta opción establece el valor de los parámetros elev.dem.path y elev.dem en aplicaciones OTB. Establecer este valor globalmente permite a los usuarios compartirlo a través de múltiples algoritmos de procesamiento. Vacío por defecto.

Compatibilidad y resolución de problemas

A partir de OTB 6.6.1, las nuevas versiones de OTB se hacen compatibles con al menos la última versión de QGIS disponible en ese momento.

Si tiene problemas con las aplicaciones OTB en el procesamiento de QGIS, abra un problema en el rastreador de errores de OTB <https://gitlab.orfeo-toolbox.org/orfeotoolbox/otb/-/issues>`_, usando la etiqueta `qgis`.

Puede encontrar información adicional sobre OTB y QGIS en OTB Cookbook.

CAPÍTULO 24

Proveedor de procesos y algoritmos

Los algoritmos de procesamiento y sus parámetros (como se presentan en la interfaz de usuario) se documentan aquí.

24.1 Proveedor de algoritmos QGIS

El proveedor de algoritmos de QGIS implementa varios análisis y operaciones de geoprocesamiento utilizando en su mayoria QGIS API. Por lo que casi todos los algoritmos de este proveedor funcionará «fuera de caja» sin ninguna configuración adicional.

El proveedor incorpora algunos algoritmos desde plugins y también añade sus propios algoritmos.

24.1.1 Teselas 3D

Convertir B3DM a GLTF

Convierte archivos del formato heredado `.B3DM` a `.GLTF` o `.GLB`.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Entrada B3DM	`INPUT`	[Tesela 3D]	Archivo de entrada a convertir.
Archivo de salida	`OUTPUT`	[Tesela 3D] Predeterinado: `[Save to temporary file]`	Especifique el archivo de teselas 3D de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo de salida	`OUTPUT`	[Tesela 3D]	Capa de salida en formato `.GLTF`.

Código Python

Identificación del algoritmo: `native:b3dmtogltf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Convertir GLTF en objetos espaciales vector

Convierte el contenido de los archivos `.GLTF` o `.GLB` a formatos estándar de capas vectoriales.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
GLTF de entrada	`INPUT`	[Tesela 3D]	Archivo de entrada a convertir.
Polígonos de salida Opcional	`OUTPUT_POLYGON`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida para almacenar las características poligonales. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Líneas de salida Opcional	`OUTPUT_LINES`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida para almacenar las características de línea. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Polígonos de salida	`OUTPUT_POLYGON`	[vectorial: poligonal]	Una capa vectorial que almacena los objetos espaciales polígonos en la capa de entrada.
Líneas de salida	`OUTPUT_LINES`	[vectorial: lineal]	Una capa vectorial que almacena los objetos espaciales línea en la capa de entrada.

Código Python

ID Algoritmo: `native:gltftovector`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.2 Cartografía

Alinear puntos a entidades

Calcula la rotación requerida para alinear entidades puntuales con su entidad más cercana de otra capa de referencia. Se agrega un nuevo campo a la capa de salida que se rellena con el ángulo (en grados, sentido horario) hasta la entidad de referencia más cercana.

Opcionalmente, la simbología de la capa de salida se puede configurar para usar automáticamente el campo de rotación calculado para rotar los símbolos de marcador. Si desea, se puede establecer una distancia máxima a usar cuando se alinean puntos, para evitar alinear puntos aislados con entidades distantes.

Consejo

Este algoritmo está diseñado para casos de uso como la alineación de símbolos de punto de construcción para seguir la dirección de la carretera más cercana.

Permite la modificación in situ de *características* de objetos espaciales punto

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	Entidad de tipo punto para calcular la rotación de
Capa de referencia	`REFERENCE_LAYE`	[vector: geometría]	Capa para encontrar la entidad más cercana para el cálculo de rotación
Máxima distancia a considerar Opcional	`MAX_DISTANCE`	[numérico: doble] Predeterminado: No establecido	Si no se encuentra ninguna entidad de referencia dentro de esta distancia, ninguna rotación es asignada a la entidad de punto.

continúe en la próxima página

Tabla 24.5 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Nombre del campo angular	`FIELD_NAME`	[cadena de texto] Predeterminado: "rotation"	Campo en el cual almacenar el valor de rotación.
Aplicar automáticamente la simbología	`APPLY_SYMBOLOG`	[booleano] Predeterminado: Verdadero	Rotar el marcador de símbolo de las entidades usando el valor del campo de ángulo
Capa alineada	`OUTPUT`	[vector: punto] Predeterinado: `[Save to temporary file]`	Especifique la capa de vectores de salida rotada. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa alineada	`OUTPUT`	[vector: punto]	La capa de tipo punto cuenta con un campo de rotación. Si se carga en QGIS, se aplica por defecto la simbología de la capa de entrada, con una rotación definida por datos de su símbolo de marcador.

Código Python

Identificación del algoritmo: `native:angletonearest`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Combinar bases de datos de estilos

Combina múltiples bases de datos de estilo QGIS en una única base de datos de estilo. Si existen elementos del mismo tipo con el mismo nombre en diferentes bases de datos de origen, se les cambiará el nombre para tener nombres únicos en la base de datos combinada de salida.

Ver también

Crear una base de datos de estilo a partir del proyecto

Parametros

Etiqueta	Nombre	Tipo	Descripción
bases de datos de entrada	`INPUT`	[file] [list]	Archivos que contienen elementos de estilo QGIS
Objetos a combinar	`OBJECTS`	[enumeración] [lista]	Tipos de elementos de estilo en las bases de datos de entrada que le gustaría incluir en la nueva base de datos. Estos pueden ser: • 0 — *Símbolos* • 1 — *Rampas de color* • 2 — *Formatos de texto* • 3 — *Configuración de etiqueta*
**Base de datos de Estilo Saliente	`OUTPUT`	[archivo] Predeterinado: `[Save to temporary file]`	Salida: archivo `.XML` que combina los elementos de estilo seleccionados. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Recuento de rampas de color	`COLORRAMPS`	[numérico: entero]	Número de rampas de color
Recuento de configuraciones de etiqueta	`LABELSETTINGS`	[numérico: entero]	Número de configuraciones de etiqueta
**Base de datos de Estilo Saliente	`OUTPUT`	[archivo]	Salida `.XML` archivo que combina los elementos de estilo seleccionados
Recuento de símbolos	`SYMBOLS`	[numérico: entero]	Número de símbolos
Recuento de formatos de texto	`TEXTFORMATS`	[numérico: entero]	Número de formatos de texto

Código Python

Identificación del algoritmo: `native:combinestyles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear representación categorizada a partir de estilos

Establece el renderizador de una capa vectorial en un renderizador categorizado utilizando símbolos coincidentes de una base de datos de estilos. Si no se especifica ningún archivo de estilo, en su lugar se utilizan los símbolos de la actual *biblioteca de símbolos*.

Se utiliza una expresión o campo específico para crear categorías para el renderizador. Cada categoría se empareja individualmente con los símbolos que existen dentro de la base de datos de estilo XML QGIS especificada. Siempre que se encuentre un nombre de símbolo coincidente, el símbolo de la categoría se establecerá en este símbolo coincidente.

Si se desea, los resultados también pueden ser tablas que contengan listas de las categorías que no pudieron coincidir con los símbolos, y símbolos que no coincidieron con las categorías.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial para aplicar un estilo categorizado
** Categorizar usando expresión **	`FIELD`	[expresión]	Campo o expresión para categorizar los objetos
Base de datos de estilos (déjela en blanco para usar símbolos guardados)	`STYLE`	[archivo]	Archivo (`.XML`) que contiene los símbolos que se aplicarán a las categorías de la capa de entrada. El archivo se puede obtener en la Herramienta del Administrador de estilos Compartir símbolos. Si no se especifica ningún archivo, se utiliza la biblioteca de símbolos locales de QGIS.
Usar coincidencias sensibles a mayúsculas y minúsculas para nombres de símbolos	`CASE_SENSITIVE`	[booleano] Predeterminado: Falso	Si es Verdadero(marcado), aplica una comparación sensible a mayúsculas y minúsculas entre las categorías y los nombres de los símbolos
Ignorar los caracteres no alfanuméricos mientras coincidan	`TOLERANT`	[booleano] Predeterminado: Falso	Si es Verdadero (marcado), los caracteres no alfanuméricos en las categorías y los nombres de los símbolos se ignorarán, lo que permitirá una mayor tolerancia durante el partido.
Categorías que no coinciden Opcional	`NON_MATCHING_C`	[vector: tabla] Preestablecido: `[Omitir salida]`	Tabla de salida para categorías que no coinciden con ningún símbolo de la base de datos. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

continúe en la próxima página

Tabla 24.9 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Nombres de símbolos que no coinciden Opcional	`NON_MATCHING_S`	[vector: tabla] Preestablecido: `[Omitir salida]`	Tabla de salida para símbolos de la base de datos de estilos proporcionada que no coinciden con ninguna categoría. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Categorías que no coinciden	`NON_MATCHING_C`	[vector: tabla]	Muestra las categorías que no pueden coincidir con ningún símbolo en la base de datos de estilos proporcionada
Nombres de símbolos que no coinciden	`NON_MATCHING_S`	[vector: tabla]	Muestra los símbolos de la base de datos de estilo proporcionada que no pueden coincidir con ninguna categoría
Capa categorizada	`OUTPUT`	[la misma que la entrada]	La capa de vector de entrada con el estilo categorizado aplicado. No se emite ninguna capa nueva.

Código Python

Identificación del algoritmo: `native:categorizeusingstyle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear una base de datos de estilo a partir del proyecto

Extrae todos los objetos de estilo (símbolos, rampas de color, formatos de texto y configuraciones de etiquetas) de un proyecto QGIS.

Los símbolos extraídos se guardan en una base de datos de estilo QGIS (formato `XML`), que se puede administrar e importar a través del cuadro de diálogo *Administrador de Estilos*.

Ver también

Combinar bases de datos de estilos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Proyecto de entrada (déjelo en blanco para usar el actual) Opcional	`INPUT`	[archivo]	Un archivo de proyecto QGIS para extraer los elementos de estilo de
Objectos a extraer	`OBJECTS`	[enumeración] [lista]	Tipos de elementos de estilo en el proyecto de entrada que le gustaría poner en la nueva base de datos. Estos pueden ser: • 0 — *Símbolos* • 1 — *Rampas de color* • 2 — *Formatos de texto* • 3 — *Configuración de etiqueta*
**Base de datos de Estilo Saliente	`OUTPUT`	[archivo] Predeterinado: `[Save to temporary file]`	Especifique el archivo de salida `.XML` para los elementos de estilo seleccionados. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Recuento de rampas de color	`COLORRAMPS`	[numérico: entero]	Número de rampas de color
Recuento de configuraciones de etiqueta	`LABELSETTINGS`	[numérico: entero]	Número de configuraciones de etiqueta
**Base de datos de Estilo Saliente	`OUTPUT`	[archivo]	Archivo de salida `.XML` para los elementos de estilo seleccionados
Recuento de símbolos	`SYMBOLS`	[numérico: entero]	Número de símbolos
Recuento de formatos de texto	`TEXTFORMATS`	[numérico: entero]	Número de formatos de texto

Código Python

Identificación del algoritmo: `native:stylefromproject`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar diseño de atlas como imagen

Exporta el diseño de atlas de impresión como archivos de imagen (por ejemplo, imágenes PNG o JPEG).

Si se establece una capa de cobertura, se sobrescribirán los ajustes del diseño de atlas seleccionado expuestos en este algoritmo. En este caso, un filtro vacío o una expresión de ordenación por desactivará esos ajustes.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Diseño de atlas	`LAYOUT`	[diseño]	Capa a exportar
Capa de cobertura Opcional	`COVERAGE_LAYER`	[vector: geometría]	Capa a usar para generar el atlas
Expresión de filtrado	`FILTER_EXPRESS`	[expresión]	Expresión para filtrar las características del atlas
Expresión de ordenar Opcional	`SORTBY_EXPRESS`	[expresión]	Expresión para ordenar las características del atlas
Ordenación inversa	`SORTBY_REVERSE`	[booleano] Predeterminado: Falso	Determina si la ordenación debe ser invertida. Se utiliza cuando se proporciona una expresión de ordenación.
Expresión del nombre del archivo de salida	`FILENAME_EXPRE`	[expresión] Predeterminado: “output_”‖@atlas_fea	Expresión para generar los nombres de los archivos
Carpeta de salida	`FOLDER`	[carpeta]	Directorio de destino donde las imágenes se han generado

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Capas de mapa para asignar a elemento(s) de mapa desbloqueados Opcional	`LAYERS`	[enumeration] [layer]	Capas a mostrar en los elemento(s) del mapa cuyo contenido no está bloqueado
Formato de imagen	`EXTENSION`	[enumeración] Predeterminado: png	Formato de archivo de la(s) salida(s) generada(s). La lista de formatos disponibles varía según el sistema operativo y los controladores instalados.
PPP Opcional	`DPI` Predeterminado: No establecido	[numérico: doble]	DPI de los archivos de salida. Si no se establece, se utilizará el valor de la configuración del diseño de impresión.
Generar un archivo mundial	`GEOREFERENCE`	[booleano] Predeterminado: Verdadero	Determina si se debe generar un archivo de mundo
Exportar metadatos RDF	`INCLUDE_METADA`	[booleano] Predeterminado: Verdadero	Determina si se deben generar metadatos RDF (título, autor, …)

continúe en la próxima página

Tabla 24.14 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Habilitar antialiasing	`ANTIALIAS`	[booleano] Predeterminado: Verdadero	Determina si se debe activar el antialiasing

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo de imagen	`OUTPUT`	[archivo]	Archivos de imagen generados por el diseño del atlas

Código Python

Identificación del algoritmo: `native:atlaslayouttoimage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar el diseño del atlas como PDF (varios archivos)

Exporta el atlas de un diseño de impresión a varios archivos PDF.

Si se establece una capa de cobertura, se sobrescribirán los ajustes del diseño de atlas seleccionado expuestos en este algoritmo. En este caso, un filtro vacío o una expresión de ordenación por desactivará esos ajustes.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Diseño de atlas	`LAYOUT`	[diseño]	Capa a exportar
Capa de cobertura Opcional	`COVERAGE_LAYER`	[vector: geometría]	Capa a usar para generar el atlas
Expresión de filtrado	`FILTER_EXPRESS`	[expresión]	Expresión para filtrar las características del atlas
Expresión de ordenar Opcional	`SORTBY_EXPRESS`	[expresión]	Expresión para ordenar las características del atlas
Ordenación inversa	`SORTBY_REVERSE`	[booleano] Predeterminado: Falso	Determina si la ordenación debe ser invertida. Se utiliza cuando se proporciona una expresión de ordenación.

continúe en la próxima página

Tabla 24.16 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
**Nombre del archivo de salida Opcional	OUTPUT_FILENAM	[expresión]	Patrón de nombre de los archivos PDF de salida.
Carpeta de salida	OUTPUT_FOLDER	[carpeta]	Carpeta de destino para los archivos PDF de salida.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Capas de mapa para asignar a elemento(s) de mapa desbloqueados Opcional	LAYERS	[enumeration] [layer]	Capas a mostrar en los elemento(s) del mapa cuyo contenido no está bloqueado
PPP Opcional	DPI Predeterminado: No establecido	[numérico: doble]	DPI de los archivos de salida. Si no se establece, se utilizará el valor de la configuración del diseño de impresión.
Exportar siempre como vectoriales	FORCE_VECTOR	[booleano] Predeterminado: Falso	Determina si los datos vectoriales deben dejarse como vectores
Exportar siempre como ráster	FORCE_RASTER	[booleano] Predeterminado: Falso	Obliga a rasterizar todos los elementos del mapa. Este parámetro tiene prioridad sobre el parámetro FORCE_VECTOR.
Añadir información de georreferencia	GEOREFERENCE	[booleano] Predeterminado: Verdadero	Determina si se debe generar un archivo de mundo
Exportar metadatos RDF	INCLUDE_METADA	[booleano] Predeterminado: Verdadero	Determina si se deben generar metadatos RDF (título, autor, …)
Deshabilitar exportaciones de capa ráster teselada	DISABLE_TILED	[booleano] Predeterminado: Falso	Determina si el ráster debe ser mosaico
Simplificar geometrías para reducir el tamaño del archivo de salida	SIMPLIFY	[booleano] Predeterminado: Verdadero	Determina si las geometrías deben simplificarse para reducir el tamaño del archivo de salida
Exportar texto	TEXT_FORMAT	[enumeración] Predeterminado: 0	Determina si el texto debe exportarse como ruta u objeto de texto. *Opciones posibles* son: • 0 - Exportar siempre el texto como rutas (recomendado) • 1 - Exportar siempre los textos como objetos de texto • 2 - Preferir la exportación de textos como objetos de texto

continúe en la próxima página

Tabla 24.17 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Compresión de imágenes	`IMAGE_COMPRESS`	[enumeración] Predeterminado: 0	Determina el nivel de compresión de la imagen y lo adecuado que podría ser el archivo para salidas de impresión o postproducción en aplicaciones externas. Las opciones posibles son: • 0 - Con pérdida (JPEG) • 1 - Sin pérdida

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo PDF	`OUTPUT`	[archivo]	Archivo PDF correspondiente al diseño del atlas exportado

Código Python

ID del algoritmo: `native:atlaslayouttomultiplepdf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar la composición del atlas como PDF (archivo único)

Exporta el atlas de una composición de impresión como un único archivo PDF.

Si se establece una capa de cobertura, se sobrescribirán los ajustes del diseño de atlas seleccionado expuestos en este algoritmo. En este caso, un filtro vacío o una expresión de ordenación por desactivará esos ajustes.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Diseño de atlas	`LAYOUT`	[diseño]	Capa a exportar
Capa de cobertura Opcional	`COVERAGE_LAYER`	[vector: geometría]	Capa a usar para generar el atlas
Expresión de filtrado	`FILTER_EXPRESS`	[expresión]	Expresión para filtrar las características del atlas
Expresión de ordenar Opcional	`SORTBY_EXPRESS`	[expresión]	Expresión para ordenar las características del atlas

continúe en la próxima página

Tabla 24.19 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Ordenación inversa	SORTBY_REVERSE	[booleano] Predeterminado: Falso	Determina si la ordenación debe ser invertida. Se utiliza cuando se proporciona una expresión de ordenación.
Archivo PDF	OUTPUT	[archivo] Predeterminado: [Guardar en archivo temporal]	Nombre (incluida la ruta) del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Capas de mapa para asignar a elemento(s) de mapa desbloqueados Opcional	LAYERS	[enumeration] [layer]	Capas a mostrar en los elemento(s) del mapa cuyo contenido no está bloqueado
PPP Opcional	DPI Predeterminado: No establecido	[numérico: doble]	DPI de los archivos de salida. Si no se establece, se utilizará el valor de la configuración del diseño de impresión.
Exportar siempre como vectoriales	FORCE_VECTOR	[booleano] Predeterminado: Falso	Determina si los datos vectoriales deben dejarse como vectores
Exportar siempre como ráster	FORCE_RASTER	[booleano] Predeterminado: Falso	Obliga a rasterizar todos los elementos del mapa. Este parámetro tiene prioridad sobre el parámetro FORCE_VECTOR.
Añadir información de georreferencia	GEOREFERENCE	[booleano] Predeterminado: Verdadero	Determina si se debe generar un archivo de mundo
Exportar metadatos RDF	INCLUDE_METADA	[booleano] Predeterminado: Verdadero	Determina si se deben generar metadatos RDF (título, autor, …)
Deshabilitar exportaciones de capa ráster teselada	DISABLE_TILED	[booleano] Predeterminado: Falso	Determina si el ráster debe ser mosaico
Simplificar geometrías para reducir el tamaño del archivo de salida	SIMPLIFY	[booleano] Predeterminado: Verdadero	Determina si las geometrías deben simplificarse para reducir el tamaño del archivo de salida
Exportar texto	TEXT_FORMAT	[enumeración] Predeterminado: 0	Determina si el texto debe exportarse como ruta u objeto de texto. *Opciones posibles* son: • 0 - Exportar siempre el texto como rutas (recomendado) • 1 - Exportar siempre los textos como objetos de texto • 2 - Preferir la exportación de textos como objetos de texto

continúe en la próxima página

Tabla 24.20 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Compresión de imágenes	`IMAGE_COMPRESS`	[enumeración] Predeterminado: 0	Determina el nivel de compresión de la imagen y lo adecuado que podría ser el archivo para salidas de impresión o postproducción en aplicaciones externas. Las opciones posibles son: • 0 - Con pérdida (JPEG) • 1 - Sin pérdida

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo PDF	`OUTPUT`	[archivo]	Archivo PDF correspondiente al diseño del atlas exportado

Código Python

Identificación del algoritmo: `native:atlaslayouttopdf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar diseño de impresión como imagen

Exporta un diseño de impresión como un archivo de imagen (por ejemplo, imágenes PNG o JPEG)

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Diseño de Impresión	`LAYOUT`	[diseño]	Capa a exportar
Archivo de imagen	`OUTPUT`	[archivo] Predeterminado: [Guardar en archivo temporal]	Nombre (incluida la ruta) del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Capas de mapa para asignar a elemento(s) de mapa desbloqueados Opcional	`LAYERS`	[enumeration] [layer]	Capas a mostrar en los elemento(s) del mapa cuyo contenido no está bloqueado
PPP Opcional	`DPI` Predeterminado: No establecido	[numérico: doble]	DPI de los archivos de salida. Si no se establece, se utilizará el valor de la configuración del diseño de impresión.
Generar un archivo mundial	`GEOREFERENCE`	[booleano] Predeterminado: Verdadero	Determina si se debe generar un archivo de mundo
Exportar metadatos RDF	`INCLUDE_METADA`	[booleano] Predeterminado: Verdadero	Determina si se deben generar metadatos RDF (título, autor, …)
Habilitar antialiasing	`ANTIALIAS`	[booleano] Predeterminado: Verdadero	Determina si se debe activar el antialiasing

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo de imagen	`OUTPUT`	[archivo]	Archivo de imagen correspondiente al diseño de impresión exportado

Código Python

Identificación del algoritmo: `native:printlayouttoimage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar el diseño de impresión como PDF

Exporta un diseño de impresión como PDF.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Diseño de Impresión	`LAYOUT`	[diseño]	Capa a exportar
Archivo PDF	`OUTPUT`	[archivo] Predeterminado: [Guardar en archivo temporal]	Nombre (incluida la ruta) del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Capas de mapa para asignar a elemento(s) de mapa desbloqueados Opcional	`LAYERS`	[enumeration] [layer]	Capas a mostrar en los elemento(s) del mapa cuyo contenido no está bloqueado
PPP Opcional	`DPI` Predeterminado: No establecido	[numérico: doble]	DPI de los archivos de salida. Si no se establece, se utilizará el valor de la configuración del diseño de impresión.
Exportar siempre como vectoriales	`FORCE_VECTOR`	[booleano] Predeterminado: Falso	Determina si los datos vectoriales deben dejarse como vectores
Exportar siempre como ráster	`FORCE_RASTER`	[booleano] Predeterminado: Falso	Obliga a rasterizar todos los elementos del mapa. Este parámetro tiene prioridad sobre el parámetro `FORCE_VECTOR`.
Añadir información de georreferencia	`GEOREFERENCE`	[booleano] Predeterminado: Verdadero	Determina si se debe generar un archivo de mundo
Exportar metadatos RDF	`INCLUDE_METADA`	[booleano] Predeterminado: Verdadero	Determina si se deben generar metadatos RDF (título, autor, …)
Deshabilitar exportaciones de capa ráster teselada	`DISABLE_TILED`	[booleano] Predeterminado: Falso	Determina si el ráster debe ser mosaico
Simplificar geometrías para reducir el tamaño del archivo de salida	`SIMPLIFY`	[booleano] Predeterminado: Verdadero	Determina si las geometrías deben simplificarse para reducir el tamaño del archivo de salida

continúe en la próxima página

Tabla 24.26 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Exportar texto	`TEXT_FORMAT`	[enumeración] Predeterminado: 0	Determina si el texto debe exportarse como ruta u objeto de texto. *Opciones posibles* son: • 0 - Exportar siempre el texto como rutas (recomendado) • 1 - Exportar siempre los textos como objetos de texto • 2 - Preferir la exportación de textos como objetos de texto
Compresión de imágenes	`IMAGE_COMPRESS`	[enumeración] Predeterminado: 0	Determina el nivel de compresión de la imagen y lo adecuado que podría ser el archivo para salidas de impresión o postproducción en aplicaciones externas. Las opciones posibles son: • 0 - Con pérdida (JPEG) • 1 - Sin pérdida
Exportar capas como archivos PDF separados	`SEPARATE_LAYER`	[booleano] Predeterminado: Falso	Si es Verdadero, se creará un archivo PDF independiente por capa y por elemento de mapa en la maqueta. Además, se pueden crear archivos PDF independientes para otros elementos de diseño complejos, lo que da lugar a un conjunto de archivos PDF que contienen componentes atómicos lógicos del diseño.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo PDF	`OUTPUT`	[archivo]	Archivo(s) PDF correspondiente(s) al diseño de impresión exportado

Código Python

Identificación del algoritmo: `native:printlayouttopdf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extraer etiquetas

Extrae información de etiquetas de un mapa renderizado a una extensión y escala determinadas.

Si se proporciona un tema de mapa, el mapa renderizado coincidirá con la visibilidad y simbología de ese tema. Si se deja en blanco, se utilizarán todas las capas visibles del proyecto. La información extraída de la etiqueta incluye: posición (servida como geometrías de puntos), el nombre de la capa asociada y el ID de la característica, el texto de la etiqueta, la rotación (en grados, en el sentido de las agujas del reloj), la alineación multilínea y los detalles de la fuente.

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión del mapa	`EXTENT`	[extensión]	Extensión del mapa de la que extraer las etiquetas Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Escala de Mapa	`SCALE`	[escala]	Las etiquetas extraídas se renderizarán utilizando sus propiedades establecidas a esta escala.
Tema del mapa Opcional	`MAP_THEME`	[maptheme]	Un tema de mapa que muestra las capas de las que extraer las etiquetas. Si no se establece, se extraen las etiquetas de las capas visibles en ese momento.
Incluir etiquetas sin colocar	`INCLUDE_UNPLAC`	[booleano] Predeterminado: Verdadero	Especifique si deben extraerse todas las etiquetas solapadas, incluidas las conflictivas (por tanto, no colocadas).

continúe en la próxima página

Tabla 24.28 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Etiquetas extraídas	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida para la extensión o extensiones. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Resolución del mapa (en PPP) Opcional	`DPI` Predeterminado: 96.0	[numérico: doble]	

Salidas

Etiqueta	Nombre	Tipo	Descripción
Etiquetas extraídas	`OUTPUT`	[vector: punto]	Capa vectorial de puntos que representa las etiquetas obtenidas. Cada característica tiene atributos que identifican su origen (capa, ID de característica) y las propiedades de etiquetado asignadas (texto, fuente, tamaño, rotación, …). También se aplica a la capa un estilo por defecto con etiquetado y símbolo nulo. **Advertencia** Debido a que algunos de los campos generados tienen nombres con más de 10 caracteres, el uso del formato ESRI shapefile (`.SHP`) para almacenar la salida puede conducir a una representación inesperada al cargar la capa en QGIS.

Código Python

ID del algoritmo: `native:extractlabels`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Imprimir extensión de mapa de diseño a capa

Crea una capa de polígono que contiene la extensión de un elemento (o elementos) del mapa de diseño de impresión, con atributos que especifican el tamaño del mapa (en unidades de diseño, es decir, unidades del *mapa de referencia*), escala y rotación.

Si se especifica el parámetro del elemento del mapa, solo se exportará la extensión del mapa coincidente. Si no se especifica, se exportarán todas las extensiones de mapa del diseño.

Opcionalmente, se puede especificar un SRC de salida específico. Si no se especifica, se utilizará el elemento de mapa original SRC.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Diseño de Impresión	`LAYOUT`	[enumeración]	Un diseño de impresión en el proyecto actual
Elemento mapa Opcional	`MAP`	[enumeración] Predeterminado: *Todos los elementos de mapa*	El elemento(s) del mapa cuya información desea extraer. Si no se proporciona ninguno, se procesan todos los elementos del mapa.
Extensión	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida para la extensión o extensiones. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Suplantar SRC Opcional	`CRS`	[src] Por defecto: *El SRC del diseño*	Seleccione el SRC para la capa en la que se presentará la información.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Altura de mapa	`HEIGHT`	[numérico: doble]	
Extensión	`OUTPUT`	[vectorial: poligonal]	Capa vectorial poligonal de salida que contiene extensiones de todo el elemento(s) de la composición de mapa
Rotación de mapa	`ROTATION`	[numérico: doble]	
Escala de Mapa	`SCALE`	[numérico: doble]	
Anchura de Mapa	`WIDTH`	[numérico: doble]	

Código Python

Algorithm ID: `native:printlayoutmapextenttolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Establecer el estilo de capa

Aplica un estilo proporcionado a una capa. El estilo debe definirse en un archivo `QML`.

No se crean nuevas salidas: el estilo es inmediatamente asignado a la capa.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[layer]	Capa entrante a la que quiere aplicar el estilo
archivo Estilo	`STYLE`	[archivo]	Ruta al archivo de estilo `.qml`

Salidas

Etiqueta	Nombre	Tipo	Descripción
	`OUTPUT`	[la misma que la entrada]	La capa entrante con el nuevo estilo asignado. No se crea una nueva capa.

Código Python

Identificación del algoritmo: `native:setlayerstyle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Coloreado topológico

Asigna un índice de color a las entidades poligonales de tal manera que ningún polígono adyacente comparta el mismo índice de color, minimizando al mismo tiempo la cantidad de colores requeridos.

El algoritmo permite elegir el método a utilizar al asignar colores.

Se puede especificar un número mínimo de colores si se desea. El índice de color se guarda en un nuevo atributo llamado **color_id**.

El siguiente ejemplo muestra el algoritmo con cuatro colores diferentes elegidos; como puede ver, cada clase de color tiene la misma cantidad de entidades.

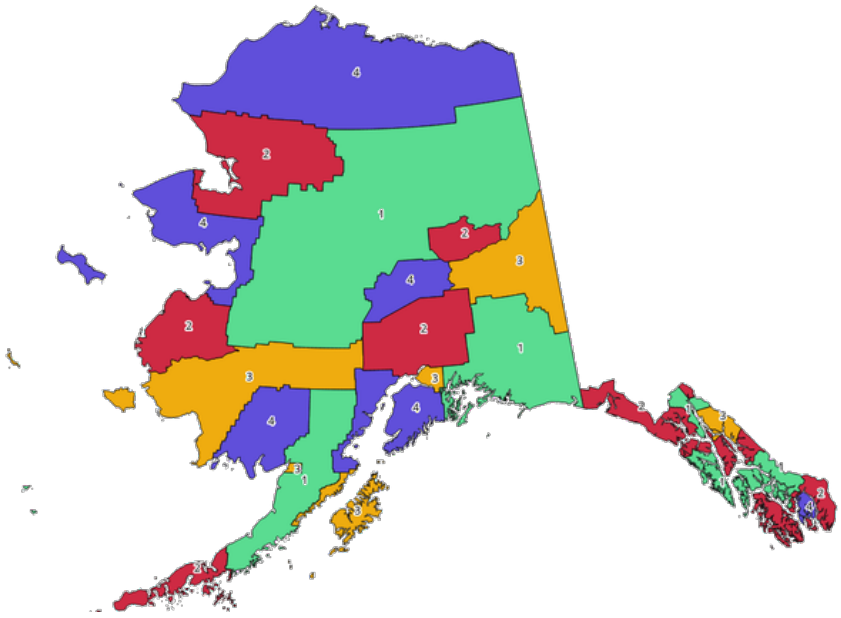


Figura 24.1: Ejemplo de colores Topológicos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	La capa poligonal entrante
Número mínimo de colores	`MIN_COLORS`	[numérico: entero] Predeterminado: 4	El número mínimo de colores para asignar. Mínimo 1, máximo 1000.
Distancia mínima entre entidades	`MIN_DISTANCE`	[numérico: doble] Predeterminado: 0.0	Evite que a las entidades cercanas (pero que no se toquen) se les asignen colores iguales. Mínimo 0.0.
Asignación de color de equilibrio	`BALANCE`	[enumeración] Predeterminado: 0	Las opciones son: • 0 — Por número de objetos Intenta asignar colores de tal manera que el número de objetos asignados a cada índice de color sea equilibrado. • 1 — Por área asignada Asigna colores para que el área total de objetos asignados a cada color esté equilibrado. Este modo puede ser útil para ayudar a evitar entidades grandes que den como resultado que uno de los colores aparezca más dominante en un mapa coloreado. • 2 — Por distancia entre colores Asigna colores para maximizar la distancia entre entidades del mismo color. Este modo ayuda a crear una distribución de colores más uniforme en un mapa.
Coloreado	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifica la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Coloreado	`OUTPUT`	[vectorial: poligonal]	Capa vectorial de polígono con una columna `color_id` añadida

Código Python

Algorithm ID: `qgis:topologicalcoloring`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Transferir anotaciones desde la capa principal

Transfiere todas las *anotaciones* de la capa principal de anotaciones de un proyecto a una nueva capa de anotaciones. La colocación de los elementos puede ajustarse entonces dentro de la pila de capas.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Nuevo nombre de la capa	`LAYER_NAME`	[cadena de texto] Por defecto: «Anotaciones	Nombre de la capa de anotaciones a crear

Salidas

Etiqueta	Nombre	Tipo	Descripción
Nuevo nombre de la capa	`OUTPUT`	[layer]	Una capa con elementos de la capa principal de anotaciones

Código Python

ID Algorithmo: `native:transferannotationsfrommain`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.3 Check Geometry

Dangle-end lines

Detects dangle-end lines in line geometries and reports them as errors. A dangle-end line is a line feature that terminates at a vertex connected to only one segment, resulting in an endpoint without a proper connection.

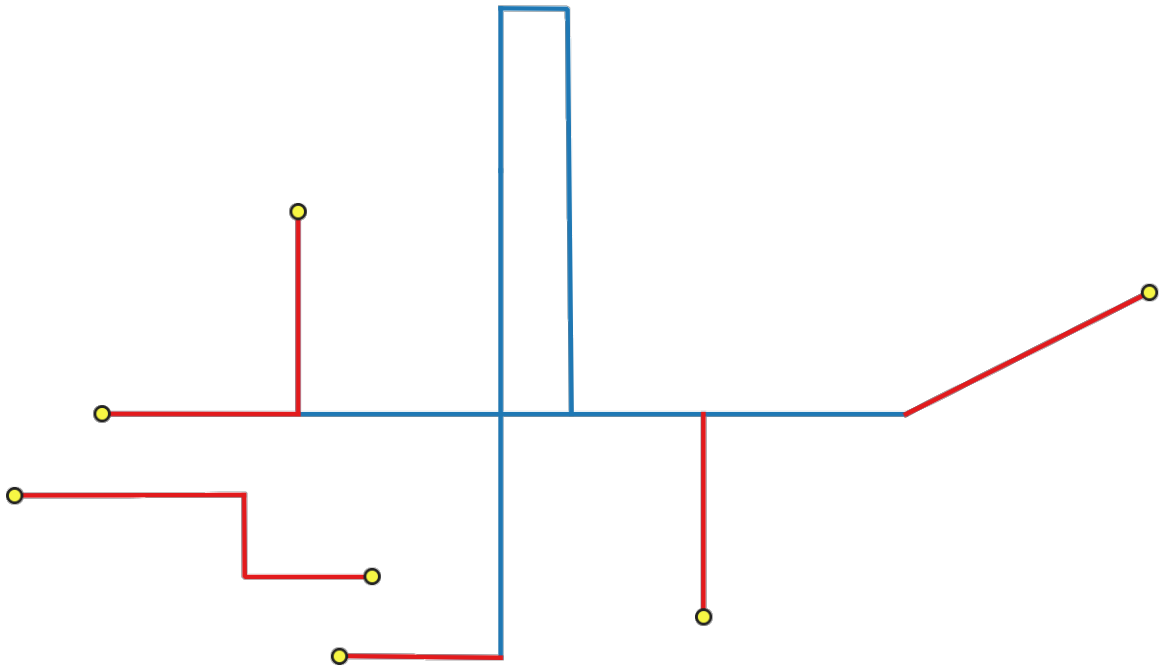


Figura 24.2: Identifying dangle-end features (in red) and vertices (yellow).

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Dangle-end errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

continúe en la próxima página

Tabla 24.40 – proviene de la página anterior

Label	Name	Type	Description
Dangle-end features Optional	`OUTPUT`	[vector: line] Default: `[Skip output]`	Line layer containing the identified dangle-end line features. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Dangle-end errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part containing the dangle-end. • `gc_ringidx`: the index of the feature's geometry ring containing the dangle-end. • `gc_vertidx`: the vertex index of the dangle-end in the feature's geometry ring. • `gc_errorx`: the x coordinate of the dangle-end. • `gc_errory`: the y coordinate of the dangle-end. • `gc_error` • `UNIQUE_ID` field: the unique ID of the input feature with the dangle-end.

continúe en la próxima página

Tabla 24.42 – proviene de la página anterior

Label	Name	Type	Description
Dangle-end features	`OUTPUT`	[vector: line]	Output line layer containing the input dangle-end features. If no dangle-end features are found, the output layer will be empty. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometrydangle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Degenerate polygons

Checks the polygons with less than 3 points, which are degenerate polygons. Degenerate polygons are errors.

Nota

This algorithm detects invalid polygon geometries (degenerate polygons). To ensure these invalid features are not filtered out before processing, open Advanced options next to the *Input layer* and set *Invalid feature filtering* to `Do not Filter (Better Performance)`.

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Degenerate polygons errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the centroid of the degenerate polygons. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

continúe en la próxima página

Tabla 24.43 – proviene de la página anterior

Label	Name	Type	Description
Degenerate polygons features Optional	`OUTPUT`	[vector: polygon] Default: `[Skip output]`	Specification of the output layer containing the degenerate polygons. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Degenerate polygons errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the geometry part of the degenerate polygon. • `gc_ringidx`: the index of the ring of the degenerate polygon. • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the degenerate polygon • `gc_errory`: the y coordinate of the centroid of the degenerate polygon • `gc_error` • `UNIQUE_ID` field: the unique ID of the input feature that is degenerate.
Degenerate polygons features	`OUTPUT`	[vector: polygon]	Output polygon layer with features containing the degenerate polygons. If no degenerate polygon features are found, the output layer will be empty. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometrydegeneratepolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Duplicated geometries

Checks for duplicated geometries in a vector layer, and reports them as errors.

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: geometry]	Layer with the geometries to check
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification
Duplicate geometries errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Duplicate geometries Optional	`OUTPUT`	[vector: same as input] Default: `[Skip output]`	Specification of the output layer containing the geometries that were found to be duplicated. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Duplicate geometries errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the duplicate geometry. • `gc_errory`: the y coordinate of the centroid of the duplicate geometry. • `gc_error`: the indices of the duplicate geometry (all except the highest index). • `UNIQUE_ID` field: the unique ID of the input feature that is a duplicate.
Duplicate geometries	`OUTPUT`	[vector: same as input]	Output layer containing the geometries that were found to be duplicated. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometryduplicate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Duplicated vertices

Checks for duplicated vertices in line or polygon geometries, and reports them as errors.

Ver también

Delete duplicated vertices, *Eliminar vértices duplicados*

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line, polygon]	Layer with the geometries to check
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification
Duplicated vertices errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Duplicated vertices features Optional	`OUTPUT`	[vector: same as input] Default: `[Skip output]`	Specification of the output layer containing the features with duplicated vertices. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Duplicated vertices errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part containing the duplicate vertex. • `gc_ringidx`: the index of the feature's geometry ring containing the duplicate vertex. • `gc_vertidx`: the index of the duplicate vertex in the feature's geometry ring. • `gc_errorx`: the x coordinate of the duplicate vertex. • `gc_errory`: the y coordinate of the duplicate vertex. • `gc_error` • `UNIQUE_ID` field: the unique ID of the input feature that has duplicate vertices.
Duplicated vertices features	`OUTPUT`	[vector: same as input]	Output layer containing the features with duplicated vertices. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometryduplicatenodes`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Features inside polygon

Checks the input geometries contained in the polygons from the polygon layers list. A polygon layer can be checked against itself. Any contained features are reported as errors.

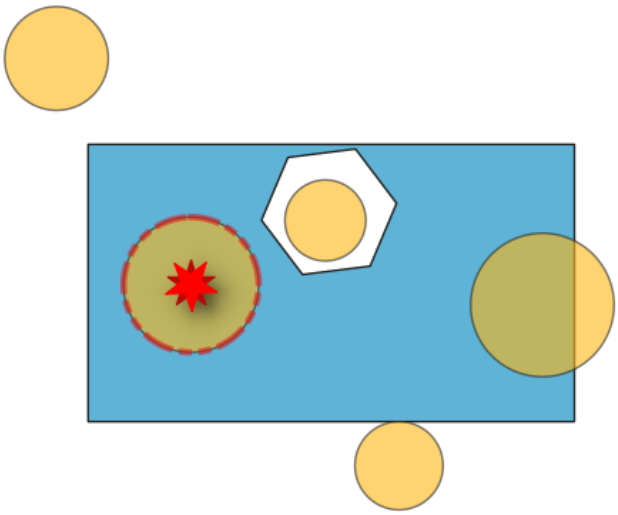


Figura 24.3: Reporting errors on features inside a polygon.

Ver también

Delete features

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: geometry]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Polygon layers	`POLYGONS`	[vector: polygon][list]	List of polygon layers to check against.
Errors from contained features	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Point layer representing the contained features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

continúe en la próxima página

Tabla 24.52 – proviene de la página anterior

Label	Name	Type	Description
Contained features Optional	`OUTPUT`	[vector: same as input] Default: `[Skip output]`	Specification of the output layer containing features within the checked polygons. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Errors from contained features	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the contained feature. • `gc_errory`: the y coordinate of the centroid of the contained feature. • `gc_error`: the layer name and feature ID of a polygon that contains the feature. • `UNIQUE_ID` field: the unique ID of the input feature that is contained within another.
Contained features	`OUTPUT`	[vector: same as input]	Output layer containing features within the checked polygons. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometrycontained`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Holes

Detects holes in polygon geometries and reports them as errors.

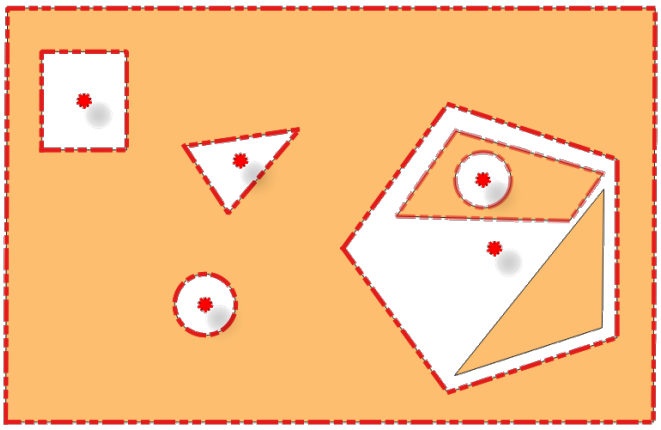


Figura 24.4: Reporting errors for holes on a polygon feature.

Ver también

Fill holes

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to check
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification

continúe en la próxima página

Tabla 24.55 – proviene de la página anterior

Label	Name	Type	Description
Holes errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Polygons with holes Optional	`OUTPUT`	[vector: polygon] Default: `[Skip output]`	Polygon layer with the features containing holes. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Holes errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part containing the hole. • `gc_ringidx`: the index of the feature's geometry ring containing the hole. • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the hole. • `gc_errory`: the y coordinate of the centroid of the hole. • `gc_error` • `UNIQUE_ID` field: the unique ID of the input feature that has a hole.
Polygon with holes	`OUTPUT`	[vector: polygon]	Output polygon layer with features containing holes. If no holes are found, the output layer will be empty. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometryhole`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Lines intersecting each other

Checks intersections between line geometries within a layer. Intersections between two different lines are errors.

Ver también

Intersecciones de línea, *Intersección*, *Self-intersections*

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Intersection errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the intersections location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Intersecting feature Optional	`OUTPUT`	[vector: line] Default: `[Skip output]`	Specification of the output layer for features intersecting each other. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Intersection errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part where the intersection occurs. • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the intersection point. • `gc_errory`: the y coordinate of the intersection point. • `UNIQUE_ID` field: the unique ID of an intersecting input feature. • `gc_error`: the layer name and index of the other intersecting feature.
Intersecting feature	`OUTPUT`	[vector: line]	Output layer containing, for each identified intersection, the feature (with same `UNIQUE_ID` value) it belongs to. If no intersections are found, the output layer will be empty. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometrylineintersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Lines intersecting other layer

Checks if the input line layer features intersect with the check layer features. An input feature that intersects with a check layer feature is an error.

Ver también

Intersecciones de línea, *Intersección*, *Self-intersections*

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Check layer	`CHECK_LAYER`	[vector: line, polygon]	Layer with the geometries to check against.
Line intersecting other layer errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output point layer containing the locations of intersection errors. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Line intersecting other layer features Optional	`OUTPUT`	[line] Default: `[Skip output]`	Line layer with the input features that intersect features in the check layer. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Line intersecting other layer errors	`ERRORS`	[vector: point]	Output point layer representing the error location and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part where the intersection occurs. • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the intersection point. • `gc_errory`: the y coordinate of the intersection point. • `UNIQUE_ID` field: the unique ID of the intersecting input feature. • `gc_error`: the index of the check layer where the intersection occurs.
Line intersecting other layer features	`OUTPUT`	[vector: line]	Output layer containing, for each identified intersection, the input feature it belongs to. If no intersections are found, the output layer will be empty. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometrylinelayerintersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Missing vertices along borders

Checks for missing vertices along polygon borders. To be topologically correct, a vertex at the junction of two polygons must be present on both polygons. Missing vertices are errors.

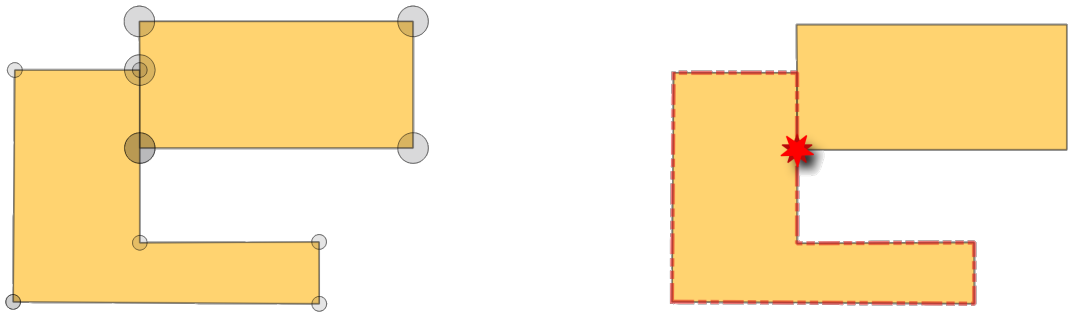


Figura 24.5: Reporting errors for missing vertices on polygon features.

Ver también

Add missing vertices along borders

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Missing vertices errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Missing vertices features Optional	`OUTPUT`	[vector: polygon] Default: `[Skip output]`	Polygon layer with the features whose vertices are missing. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Missing vertices errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the missing vertex. • `gc_errory`: the y coordinate of the missing vertex. • `gc_error` • `UNIQUE_ID` field: the unique ID of the input feature that has a missing vertex.
Missing vertices features	`OUTPUT`	[vector: polygon]	Output polygon layer with features whose vertices are missing. There will be as many (duplicate) features as there are missing vertices in each geometry. If no vertices are missing, the output layer will be empty. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometrymissingvertex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Overlaps

Calculates overlapping areas in polygon geometries, and reports areas smaller than a given minimum as errors.

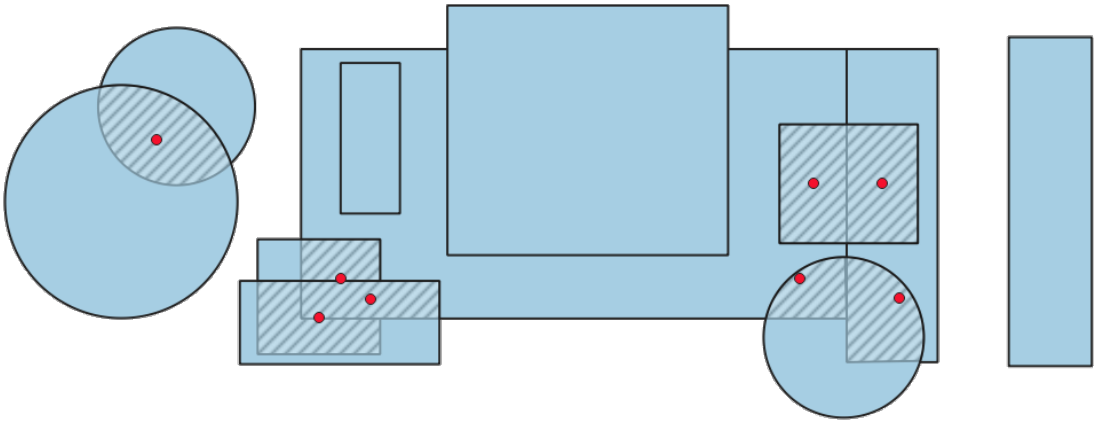


Figura 24.6: Hashed polygons indicate overlapping areas smaller than the specified minimum.

Ver también

Delete overlaps, *Validar la cobertura*

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Minimum overlap area	`MIN_OVERLAP_AR`	[numeric: double]	Minimum area of the overlap to be reported as an error. If set to 0, all overlaps are reported.
Overlap errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing centroid points of the overlapping areas. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

continúe en la próxima página

Tabla 24.67 – proviene de la página anterior

Label	Name	Type	Description
Overlap features Optional	`OUTPUT`	[vector: polygon] Default: `[Skip output]`	Specification of the output layer for features containing overlaps. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Overlap errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_errorx`: the x coordinate of the centroid of the overlapping area. • `gc_errory`: the y coordinate of the centroid of the overlapping area. • `gc_error`: the area of the overlapping geometry. • `UNIQUE_ID` field: the unique ID of an overlapped input feature. • `gc_overlap_feature_{unique_id}`: the `UNIQUE_ID` field value for the other overlapping feature.
Overlap features	`OUTPUT`	[vector: polygon]	Output layer containing the overlapping areas. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometryoverlap`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Points outside lines

Checks if the points in the input layer are covered by a line in the selected line layers. A point not covered by a line is an error.

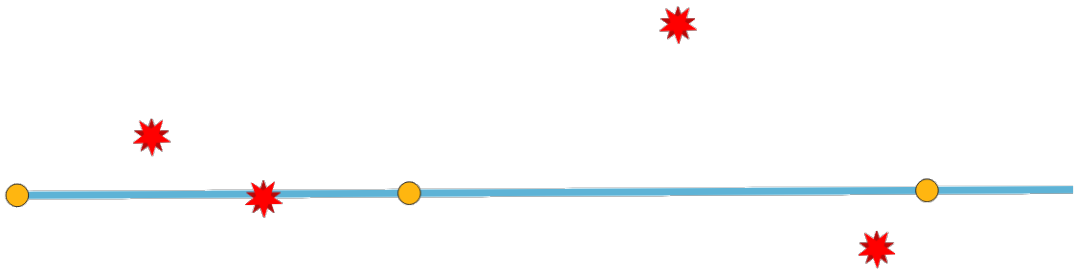


Figura 24.7: Reporting errors on points not covered by a line.

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: point]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Line layers	`LINES`	[vector: line][list]	Layer(s) with the lines to check against.
Points not covered by a line	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing points not covered by a line. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Points not covered by a line	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of point outside the lines. • `gc_errory`: the y coordinate of point outside the lines. • `gc_error` • `UNIQUE_ID` field: the unique ID of the input feature that is outside the lines.

Python code

Algorithm ID: `native:checkgeometrypointcoveredbyline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Points outside polygons

Checks if points from the input layer are in polygons from the selected polygon layers. Points that are not fully inside polygons are errors.

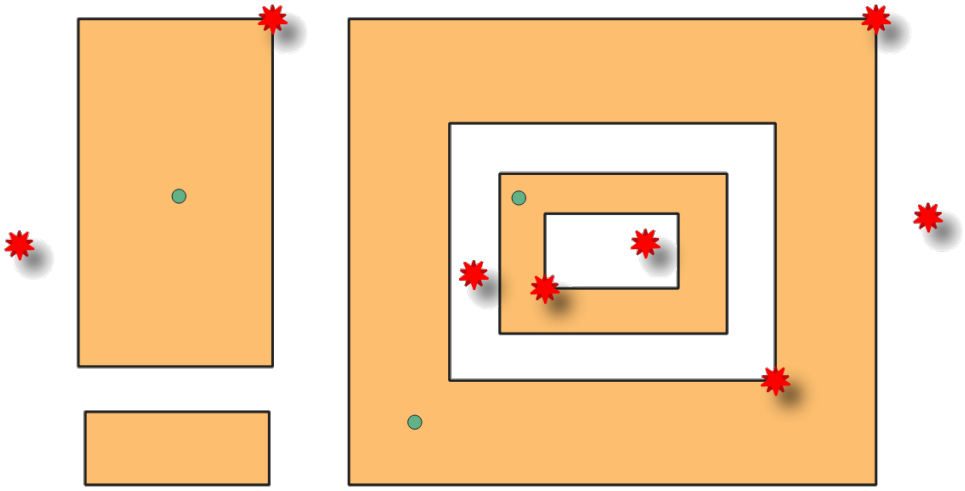


Figura 24.8: Reporting errors on points outside polygons.

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: point]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Polygon layers	`POLYGONS`	[vector: polygon][list]	Layer(s) with the polygons to check against.
Points outside polygons errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing points outside the polygons. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Points outside polygons errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of point outside the polygons. • `gc_errory`: the y coordinate of point outside the polygons. • `gc_error` • `UNIQUE_ID` field: the unique ID of the input feature that is outside the polygons.

Python code

Algorithm ID: `native:checkgeometrypointinpolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Self-contacts

Checks if the geometry has self contact points (in line or polygon), i.e., a vertex that touches more than two segments of the same ring. Self contacts are errors.

Nota

This algorithm detects invalid polygon geometries (self-contacts). To ensure these invalid features are not filtered out before processing, open Advanced options next to the *Input layer* and set *Invalid feature filtering* to `Do not Filter (Better Performance)`.

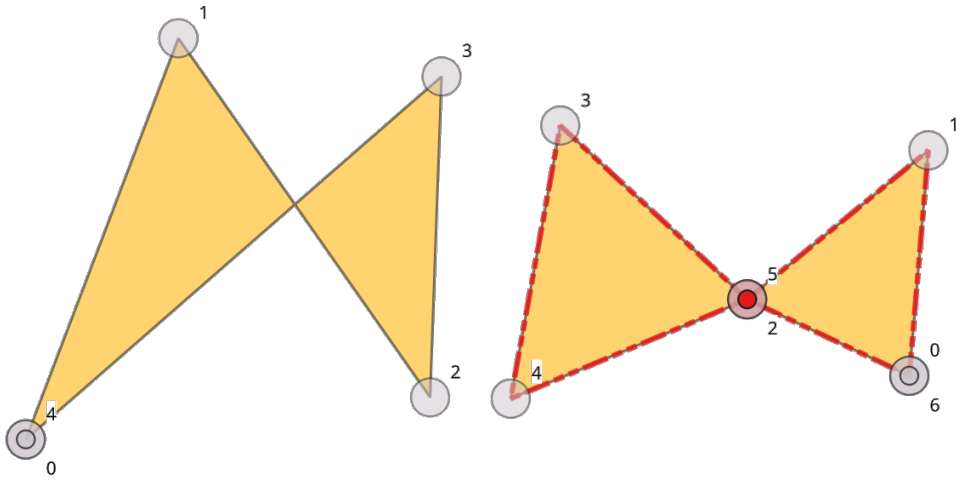


Figura 24.9: Self-intersection vs self-contact.

Ver también

Self-intersections

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line, polygon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.

continúe en la próxima página

Tabla 24.76 – proviene de la página anterior

Label	Name	Type	Description
Self contact error points	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Self contact features Optional	`OUTPUT`	[same as input] Default: `[Skip output]`	Line or polygon layer containing self-contact features. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Self contact error points	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part where the self-contact occurs. • `gc_ringidx`: the index of the ring where the self-contact occurs. • `gc_vertidx`: the index of the vertex where the self-contact occurs. • `gc_errorx`: the x coordinate of the self-contact. • `gc_errory`: the y coordinate of the self-contact. • `gc_error` • `UNIQUE_ID` field: the unique ID of the self-contacting input feature.
Self contact features	`OUTPUT`	[vector: same as input]	Output polygon or line layer with features containing the self-contact features. If no self-contact features are found, the output layer will be empty. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometryselfcontact`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Self-intersections

Detects self-intersections in line or polygon geometries, and reports them as errors. Self-intersections occur when the segments of a geometry cross over each other without having a common vertex.

Nota

This algorithm detects invalid polygon geometries (self-intersections). To ensure these invalid features are not filtered out before processing, open Advanced options next to the *Input layer* and set *Invalid feature filtering* to `Do not Filter (Better Performance)`.

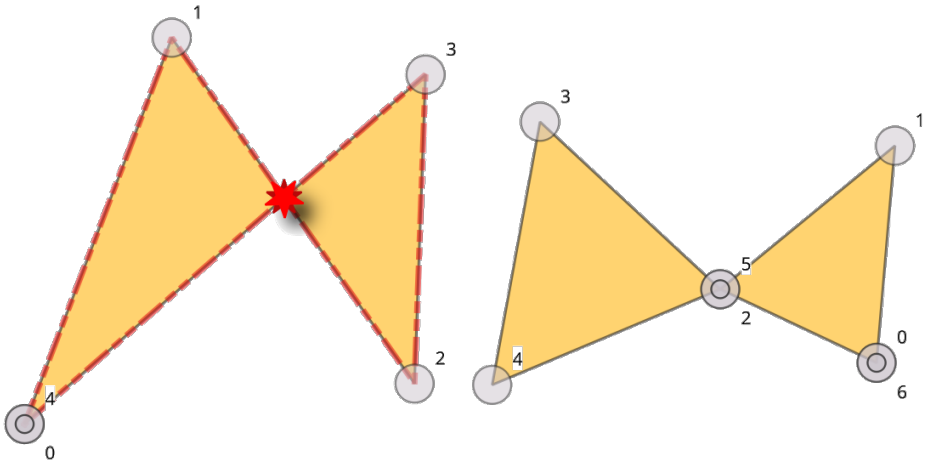


Figura 24.10: Self-intersection vs self-contact.

Ver también

Split self-intersecting geometries, *Self-contacts*

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line, polygon]	Layer with the geometries to check
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification
Self-intersecting errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Self-intersecting features Optional	`OUTPUT`	[same as input] Default: `[Skip output]`	Line or polygon layer containing self-intersected features. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Self-intersecting errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part where the self-intersection occurs. • `gc_ringidx`: the index of the feature's geometry ring where the self-intersection occurs. • `gc_vertidx` • `gc_errorx`: the x coordinate of the self-intersection. • `gc_errory`: the y coordinate of the self-intersection. • `gc_error` • `gc_segment_1`: the index of the first segment involved in the intersection. • `gc_segment_2`: the index of the second segment involved in the intersection. • `UNIQUE_ID` field: the unique ID of the self-intersecting input feature.
Self-intersecting features	`OUTPUT`	[vector: line, polygon]	Output polygon or line layer with features containing the self-intersecting features. If no self-intersecting features are found, the output layer will be empty. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometryselfintersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Sliver polygons

Detects sliver polygons in a polygon layer, i.e., polygons with a thinness greater than a specified value.

The thinness is the ratio between the area of the minimum square containing the polygon and the area of the polygon itself (a square has a thinness value of 1). The thinness value is between 1 and +infinity. A maximum area can be set for limiting the checks to polygons of a lower area. Polygons having a thinness higher than the maximum thinness are errors. To fix sliver polygons, use the *Fix small polygons* algorithm.

Nota

This algorithm detects invalid polygon geometries (sliver polygons). To ensure these invalid features are not filtered out before processing, open Advanced options next to the *Input layer* and set *Invalid feature filtering* to `Do not Filter (Better Performance)`.

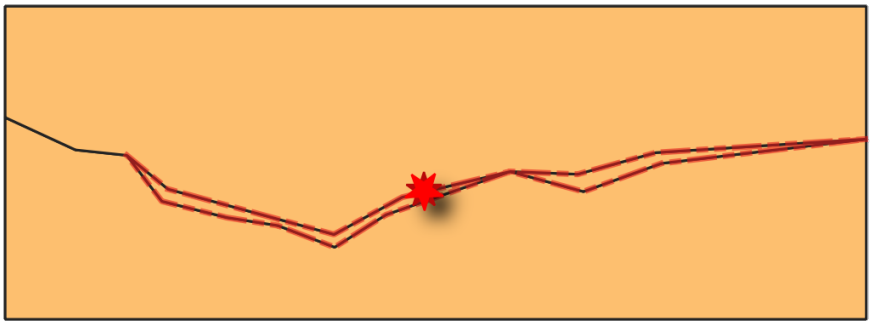


Figura 24.11: Reporting errors on features with sliver polygons.

Ver también

Fix small polygons

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Maximum thinness	`MAX_THINNESS`	[numeric: double] Default: 20.0	Maximum thinness value of a polygon to be considered valid. Any polygon with a thinness value above this threshold is reported as error.
Maximum area (map units squared)	`MAX_AREA`	[numeric: double] Default: 0.0	Maximum area of a polygon to be checked (in map units). Any polygon with an area above this threshold is skipped. A value of 0 means all polygons are checked.
Sliver polygon errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the centroid points of sliver polygon features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Sliver polygon features Optional	`OUTPUT`	[vector: polygon] Default: `[Skip output]`	Polygon layer containing sliver polygon features. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label		Name	Type	Description
Sliver errors	**polygon**	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the sliver polygon feature. • `gc_errory`: the y coordinate of the centroid of the sliver polygon feature. • `gc_error`: the thinness value of the sliver polygon feature. • `UNIQUE_ID` field: the unique ID of the input feature that is a sliver polygon.
Sliver features	**polygon**	`OUTPUT`	[vector: polygon]	Output layer with features containing the sliver polygons. If no sliver polygon features are found, the output layer will be empty. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometrysliverpolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Small angles

Compares the angles within line or polygon geometries to a specified threshold, and reports as error any angle below that value.

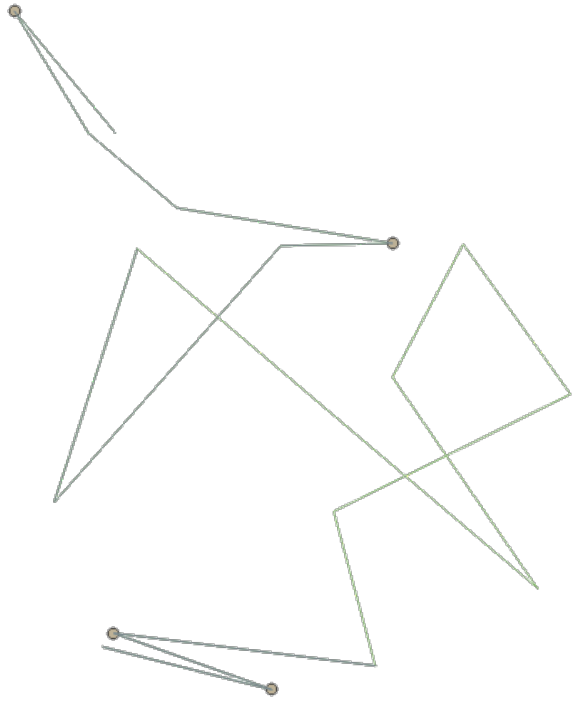


Figura 24.12: Reporting errors on a line feature for angles lower than 15°.

Ver también

Delete small angles

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line, polygon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Minimum angle	`MIN_ANGLE`	[numeric: double]	Allowed minimum angle between adjacent segments in a feature's geometry. Any angle below this threshold is reported as error.

continúe en la próxima página

Tabla 24.85 – proviene de la página anterior

Label	Name	Type	Description
Small angle errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the errors location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Small angle errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part containing the small angle. • `gc_ringidx`: the index of the feature's geometry ring containing the small angle. • `gc_vertidx`: the index of the vertex with the small angle in the feature's geometry ring. • `gc_errorx`: the x coordinate of the vertex with the small angle. • `gc_errory`: the y coordinate of the vertex with the small angle. • `gc_error`: the error angle value. • `UNIQUE_ID` field: the unique ID of the input feature with the small angle.

Python code

Algorithm ID: `native:checkgeometryangle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Small polygons

Detects polygon features whose area is below a specified value as errors.

Nota

This algorithm detects invalid polygon geometries (small polygons). To ensure these invalid features are not filtered out before processing, open Advanced options next to the *Input layer* and set *Invalid feature filtering* to `Do not Filter (Better Performance)`.

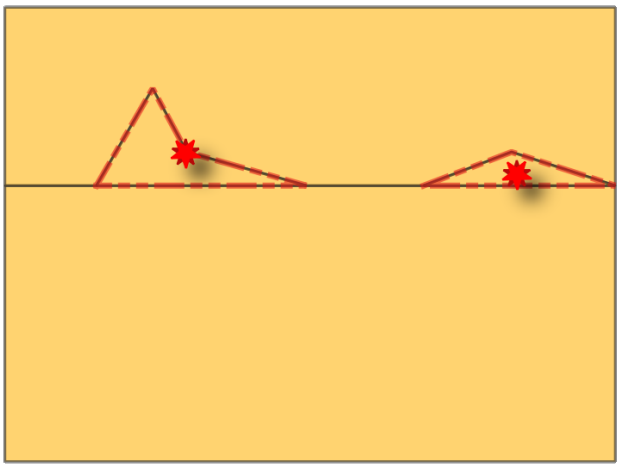


Figura 24.13: Reporting errors on features with small polygons (below given area threshold).

Ver también

Fix small polygons

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Area threshold	`AREATHRESHOLD`	[numeric: double] Default: 0	Minimum area of a polygon to be considered valid, in selected units. Any polygon with an area below this threshold is reported as error.
Small polygons errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing centroid points of small polygons. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Small polygons features Optional	`OUTPUT`	[vector: polygon] Default: `[Skip output]`	Specification of the output layer for features containing small polygon features. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Small polygons errors	ERRORS	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • gc_layerid: the ID of the input layer. • gc_layername: the name of the input layer. • gc_partidx • gc_ringidx • gc_vertidx • gc_errorx: the x coordinate of the centroid of the small polygon. • gc_errory: the y coordinate of the centroid of the small polygon. • gc_error: the area of the small polygon. • UNIQUE_ID field: the unique ID of the input feature that is a small polygon.
Small polygons features	OUTPUT	[vector: polygon]	Output polygon layer with features containing the small polygon features. If no small polygon features are found, the output layer will be empty. Available fields are the same as in the ERRORS output.

Python code

Algorithm ID: native:checkgeometryarea

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Small gaps

Checks for gaps between polygons in the input layer. Gaps with an area smaller than the gap threshold are reported as errors. If an allowed gaps layer is provided, the algorithm ignores gaps that fall entirely within polygons from this layer. An optional buffer can be applied to the allowed gaps.

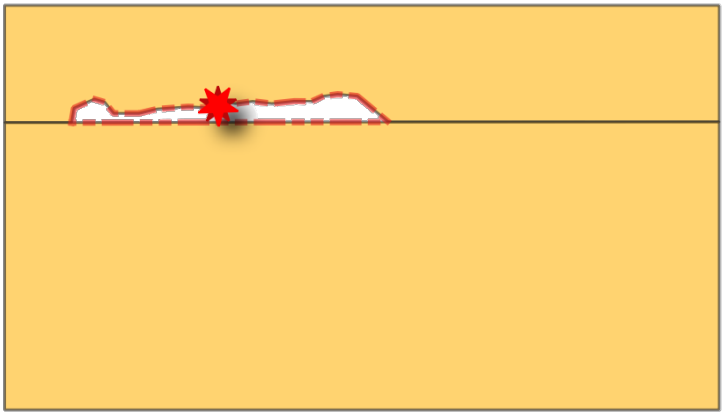


Figura 24.14: Reporting errors on polygon features for gaps smaller than the specified threshold.

Ver también

Fill gaps, *Validar la cobertura*

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Gap threshold	`GAP_THRESHOLD`	[numeric: double] Default: 0.0	Maximum area of gaps to be reported as errors, in map units. If set to 0, all the gaps are reported.
Allowed gaps layer Optional	`ALLOWED_GAPS_L`	[vector: polygon]	Optional layer specifying polygons whose areas should be ignored during the gap check. Gaps that fall entirely within these polygons will not be reported as errors.
Allowed gaps buffer Optional	`ALLOWED_GAPS_B`	[numeric: double] Default: 0.0	Buffer distance to apply to the allowed gaps layer, in selected units. Gaps located within this buffered area are ignored. Allows adding a spatial tolerance around allowed gaps to avoid reporting near-boundary gaps as errors.

continúe en la próxima página

Tabla 24.91 – proviene de la página anterior

Label	Name	Type	Description
Neighbors layer	`NEIGHBORS`	[vector: table] Default: `[Create temporary layer]`	Specification of the output table representing the gap ID and the unique ID of its neighbor features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Gap errors Optional	`ERRORS`	[vector: point] Default: `[Skip output]`	Specification of the output layer containing the centroid points of the gaps. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Gap features	`OUTPUT`	[vector: polygon] Default: `[Create temporary layer]`	Specification of the output layer containing the gap geometries. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Neighbors layer	`NEIGHBORS`	[vector: table]	A 1–N relational table, meaning that one gap can be associated with multiple neighboring polygons. The output table contains the following fields: • `gc_errorid`: the ID of the gap. • `UNIQUE_ID` field: the unique ID of an input feature that is a neighbor of the gap.

continúe en la próxima página

Tabla 24.93 – proviene de la página anterior

Label	Name	Type	Description
Gap errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the gap. • `gc_errory`: the y coordinate of the centroid of the gap. • `gc_error`: the area of the gap. • `gc_errorid`: the ID of the gap.
Gap features	`OUTPUT`	[vector: polygon]	Output layer containing the gap geometries. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometrygap`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Small segments

Calculates length of individual segments in line or polygon geometries, and reports segments shorter than a minimum length as errors.

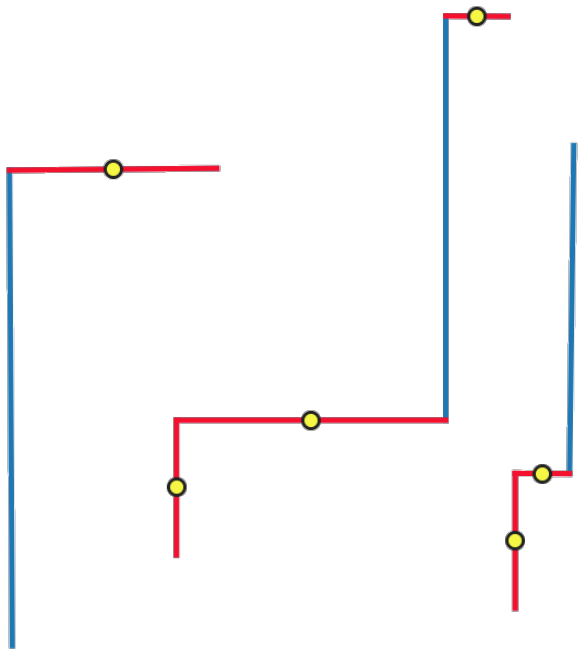


Figura 24.15: Reporting errors (in red) on single line segments shorter than the given threshold.

Ver también

Delete features

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line, polygon]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
Minimum segment length	`MIN_SEGMENT_LE`	[numeric: double] Default: 0.0	Minimum length of segments to be considered valid, in map units.
Short segments errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing centroid of short segments. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

continúe en la próxima página

Tabla 24.94 – proviene de la página anterior

Label	Name	Type	Description
Short segments features Optional	`OUTPUT`	[vector: same as input] Default: `[Skip output]`	Specification of the output layer for features containing short segments. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Short segments errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part containing the small segment. • `gc_ringidx`: the index of the feature's geometry ring containing the small segment. • `gc_vertidx`: the index of the end vertex of the small segment in the feature's geometry ring. • `gc_errorx`: the x coordinate of the centroid of the small segment. • `gc_errory`: the y coordinate of the centroid of the small segment. • `gc_error`: the error segment length. • `UNIQUE_ID` field: the unique ID of the input feature with the small segment.

continúe en la próxima página

Tabla 24.96 – proviene de la página anterior

Label	Name	Type	Description
Short segments features	`OUTPUT`	[vector: same as input]	Output layer containing, for each identified short segment, the feature it belongs to. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometrysegmentlength`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Strictly multipart

Checks if multipart geometries have more than one part. Multipart geometries with only one part are errors.

Ver también

Convert to strictly multipart

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: geometry]	Layer with the geometries to check.
Unique feature identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification.
One-part geometry errors	`ERRORS`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing centroid of the multipart geometries that have only one part. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

continúe en la próxima página

Tabla 24.97 – proviene de la página anterior

Label	Name	Type	Description
One-part geometry features Optional	`OUTPUT`	[vector: same as input] Default: `[Skip output]`	Specification of the output layer for features containing multipart geometries that have only one part. *One of*: • Skip Output • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
One-part geometry errors	`ERRORS`	[vector: point]	Output point layer representing the error locations and information. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the multipart geometry that has only one part. • `gc_errory`: the y coordinate of the centroid of the multipart geometry that has only one part. • `gc_error` • `UNIQUE_ID` field: the unique ID of the input feature that is multipart but has only one part.
One-part geometry features	`OUTPUT`	[vector: same as input]	Output layer containing features that are multipart but have only one part. Available fields are the same as in the `ERRORS` output.

Python code

Algorithm ID: `native:checkgeometrymultipart`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

24.1.4 Base de datos

Exportar a PostgreSQL

Exporta una capa vectorial a una base de datos PostgreSQL, creando una nueva relación. Si existe una relación con el mismo nombre, se puede eliminar antes de que se cree la nueva relación. Antes de esto, se debe crear una conexión entre QGIS y la base de datos PostgreSQL (ver por ejemplo *Crear una conexión almacenada*).

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa a importar	`INPUT`	[vectorial: cualquiera]	Capa vectorial a añadir a la base de datos
Base de datos (nombre de conexión)	`DATABASE`	[cadena de texto]	Nombre de la conexión a la base de datos (no el nombre de la base de datos). Las conexiones existentes se mostrarán en el cuadro combinado.
Esquema (nombre de esquema) Opcional	`SCHEMA`	[cadena de texto] Predeterminado: “public”	Nombre del esquema para almacenar los datos. Puede ser uno nuevo o ya existir.
Tabla a la que importar (dejar en blanco para usar el nombre de la capa) Opcional	`TABLENAME`	[cadena de texto] Predeterminado: “”	Define un nombre de tabla para el archivo vectorial importado. Si no se agrega nada, se utilizará el nombre de la capa.
Clave primaria de campo Opcional	`PRIMARY_KEY`	[campo de tabla: cualquiera]	Establece el campo de clave principal de un campo existente en la capa vectorial. Una columna con valores **únicos** se puede utilizar como clave principal para la base de datos.
Columna geometría	`GEOMETRY_COLUM`	[cadena de texto] Predeterminado: “geom”	Defines the name of the geometry column in the new PostgreSQL table. Geometry information for the features is stored in this column.
Codificación Opcional	`ENCODING`	[cadena de texto] Predeterminado: “UTF-8”	Define la codificación de la capa de salida.

continúe en la próxima página

Tabla 24.100 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Sobreescribir	`OVERWRITE`	[booleano] Predeterminado: Verdadero	Si la tabla especificada existe, establecer esta opción en `True` garantizará que se elimine y se creará una nueva tabla antes de agregar las características. Si esta opción es `False` y la tabla existe, el algoritmo lanzará una excepción («la relación ya existe»).
Crear índice espacial	`CREATEINDEX`	[booleano] Predeterminado: Verdadero	Especifica si crear un índice espacial o no
Convertir nombre de campos a minúsculas	`LOWERCASE_NAME`	[booleano] Predeterminado: Verdadero	Convierte los nombres de campo de la capa de vector de entrada a minúsculas
Restricción de longitud de cadena en campos de caracteres	`DROP_STRING_LE`	[booleano] Predeterminado: Falso	Si deben eliminarse o no las restricciones de longitud en los campos de caracteres
Crear geometrías de una sola pieza en lugar de varias piezas	`FORCE_SINGLEPA`	[booleano] Predeterminado: Falso	Si deben las entidades de la capa de salida ser de una sola parte en lugar de varias partes. De forma predeterminada, se conserva la información de geometrías existentes.

Salidas

El algoritmo no tiene salida.

Código Python

Algorithm ID: `native:importintopostgis`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar a SpatiaLite

Exporta una capa vectorial a una base de datos SpatiaLite. Antes de esto, se debe crear una conexión entre QGIS y la base de datos SpatiaLite (ver eg *Capas SpatiaLite*).

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa a importar	`INPUT`	[vectorial: cualquiera]	Capa vectorial a añadir a la base de datos
Base de datos de archivos	`DATABASE`	[vectorial: cualquiera]	La base de datos de archivos SQLite/SpatiaLite a la que conectar
Tabla a la que importar (dejar en blanco para usar el nombre de la capa) Opcional	`TABLENAME`	[cadena de texto] Predeterminado: ""	Define el nombre de la tabla para el archivo vectorial importado. Si no se especifica nada, se utilizará el nombre de la capa.
Clave primaria de campo Opcional	`PRIMARY_KEY`	[campo de tabla: cualquiera]	Utilizar un campo en la capa vectorial de entrada como clave principal
Columna geometría	`GEOMETRY_COLUM`	[cadena de texto] Predeterminado: "geom"	Define el nombre de la columna de geometría en la nueva tabla SpatiaLite. La información de geometría de las entidades se almacena en esta columna.
Codificación Opcional	`ENCODING`	[cadena de texto] Predeterminado: "UTF-8"	Define la codificación de la capa de salida.
Sobreescribir	`OVERWRITE`	[booleano] Predeterminado: Verdadero	Si la tabla especificada existe, establecer esta opción en `True` asegurará que se elimine y se creará una nueva tabla antes de agregar las entidades de la capa. Si esta opción es `False` y la tabla existe, el algoritmo lanzará una excepción («la tabla ya existe»).
Crear índice espacial	`CREATEINDEX`	[booleano] Predeterminado: Verdadero	Especifica si crear un índice espacial o no
Convertir nombre de campos a minúsculas	`LOWERCASE_NAME`	[booleano] Predeterminado: Verdadero	Convertir los nombres de campo de la capa vectorial entrante a minúsculas
Restricción de longitud de cadena en campos de caracteres	`DROP_STRING_LE`	[booleano] Predeterminado: Falso	Si deben eliminarse o no las restricciones de longitud en los campos de caracteres
Crear geometrías de una sola pieza en lugar de varias piezas	`FORCE_SINGLEPA`	[booleano] Predeterminado: Falso	Si deben las entidades de la capa de salida ser de una sola parte en lugar de varias partes. De forma predeterminada, se conserva la información de geometrías existentes.

Salidas

El algoritmo no tiene salida.

Código Python

Algorithm ID: `native:importintospatialite`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Paquete de capas

Añade capas a un GeoPackage.

Si el GeoPackage existe y «Sobrescribir GeoPackage existente» está marcado, se sobrescribirá (se eliminará y se volverá a crear). Si el GeoPackage existe y «Sobrescribir GeoPackage existente» no está marcado, la capa se agregará.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`LAYERS`	[vectorial: cualquiera] [lista]	Las capas (vectoriales) a importar al GeoPackage. Las capas ráster no son compatibles. Si se agrega una capa ráster, se lanzará `QgsProcessingException`.
Sobreescribir GeoPackage existente	`OVERWRITE`	[booleano] Predeterminado: Falso	Si existe el GeoPackage especificado, configurar esta opción en `True` asegurará que se elimine y se creará uno nuevo antes de agregar las capas. Si se establece en `True`, se agregarán capas.
Guardar estilos de capa en GeoPackage	`SAVE_STYLES`	[booleano] Predeterminado: Verdadero	Save the layer styles.
Guardar solo funciones seleccionadas	`SELECTED_FEATU`	[booleano] Predeterminado: Falso	Si una capa tiene una selección, establecer esta opción en `True` hará que solo se guarden las entidades seleccionadas. Para las capas sin una selección, se guardarán todas las características.
Exportar capas relacionadas siguiendo las relaciones definidas en el proyecto	`EXPORT_RELATED`	[booleano] Predeterminado: Falso	Si una capa de entrada tiene *relaciones* definida en el proyecto, al establecer esta opción como `Verdadero` se exportarán también su(s) capa(s) relacionada(s). Si la capa tiene objetos espaciales seleccionados, sólo se exportarán sus objetos espaciales relacionados, a menos que la capa relacionada sea también una capa de entrada.

continúe en la próxima página

Tabla 24.102 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
GeoPackage destinatario	`OUTPUT`	[archivo] Predeterinado: `[Save to temporary file]`	Especifique dónde almacenar el archivo GeoPackage. Uno de • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capas dentro de nuevo paquete	`OUTPUT_LAYERS`	[cadena] [lista]	La lista de capas añadidas al GeoPackage.

Código Python

Algoritmo ID: `native:package`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

PostgreSQL ejecuta y carga SQL

Permite realizar una consulta de base de datos SQL en una base de datos PostgreSQL conectada a QGIS y carga el resultado. El algoritmo **no** creará una nueva capa: está diseñado para ejecutar consultas en la propia capa.

Ejemplo

1. Establecer todos los valores de un campo existente en un valor fijo. La cadena de consulta SQL será:

   ```
   UPDATE your_table SET field_to_update=20;
   ```

   En el ejemplo anterior, los valores del campo `field_to_update` de la tabla `your_table` estarán todos configurados en `20`.

2. Cree una nueva columna``área`` y calcule el área de cada entidad con la función PostGIS `ST_AREA`.

   ```
   -- Create the new column "area" on the table your_table"
   ALTER TABLE your_table ADD COLUMN area double precision;
   -- Update the "area" column and calculate the area of each feature:
   UPDATE your_table SET area=ST_AREA(geom);
   ```

Ver también

PostgreSQL ejecuta SQL, *Ejecutar SQL*, *SpatiaLite ejecuta SQL*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Base de datos (nombre de conexión)	`DATABASE`	[cadena de texto]	La conexión de la base de datos (no el nombre de la base de datos). Las conexiones existentes se mostrarán en el cuadro combinado.
****Consulta SQL*	`SQL`	[cadena de texto]	Define la consulta SQL, por ejemplo `'UPDATE my_table SET field=10'`.
ID única de nombre de campo	`ID_FIELD`	[cadena de texto] Predeterminado: id	Establece el campo de clave principal (una columna en la tabla de resultados)
Nombre de campo de Geometría Opcional	`GEOMETRY_FIELD`	[cadena de texto] Predeterminado: "geom"	Nombre de la columna de geometría (una columna en la tabla de resultados)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Consulta SQL	`OUTPUT`	[vectorial: cualquiera]	La capa vectorial resultante se cargará en QGIS.

Código Python

Algorithm ID: `qgis:postgisexecuteandloadsql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

PostgreSQL ejecuta SQL

Permite realizar una consulta de base de datos SQL en una base de datos PostgreSQL conectada a QGIS. El algoritmo **won`t** creará una nueva capa: está diseñado para ejecutar consultas en la propia capa.

Ejemplo

1. Establecer todos los valores de un campo existente en un valor fijo. La cadena de consulta SQL será:

   ```
   UPDATE your_table SET field_to_update=20;
   ```

   En el ejemplo anterior, los valores del campo `field_to_update` de la tabla `your_table` estarán todos configurados en `20`.

2. Cree una nueva columna``área`` y calcule el área de cada entidad con la función PostGIS `ST_AREA`.

   ```
   -- Create the new column "area" on the table your_table"
   ALTER TABLE your_table ADD COLUMN area double precision;
   ```

(continúe en la próxima página)

(proviene de la página anterior)

```
-- Update the "area" column and calculate the area of each feature:
UPDATE your_table SET area=ST_AREA(geom);
```

Ver también

PostgreSQL ejecuta y carga SQL, *Ejecutar SQL*, *SpatiaLite ejecuta SQL*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Base de datos (nombre de conexión)	`DATABASE`	[cadena de texto]	La conexión de la base de datos (no el nombre de la base de datos). Las conexiones existentes se mostrarán en el cuadro combinado.
****Consulta SQL*	`SQL`	[cadena de texto]	Define la consulta SQL, por ejemplo `'UPDATE my_table SET field=10'`.

Salidas

No se crea ninguna salida. La consulta SQL se ejecuta en su lugar.

Código Python

Algoritmo ID: `native:postgisexecutesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

SpatiaLite ejecuta SQL

Permite realizar una consulta de base de datos SQL en una base de datos SpatiaLite. La base de datos se determina mediante una capa o un archivo de entrada. El algoritmo **no** creará una nueva capa: está diseñado para ejecutar consultas en la propia capa.

Ver también

PostgreSQL ejecuta SQL, *Ejecutar SQL*

For some SQL query examples see *PostgreSQL Query Examples*.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de base de datos (o archivo)	`DATABASE`	[archivo]	La base de datos de archivos SQLite/SpatiaLite a la que conectar
**Consulta SQL*	`SQL`	[cadena de texto] Predeterminado: ""	Define la consulta SQL, por ejemplo `'UPDATE my_table SET field=10'`.

Salidas

No se crea ninguna salida. La consulta SQL se ejecuta en su lugar.

Código Python

Algoritmo ID: `native:spatialiteexecutesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

SpatiaLite ejecuta SQL (DB registrada)

Permite realizar una consulta de base de datos SQL en una base de datos SpatiaLite conectada a QGIS. El algoritmo **no** creará una nueva capa: está diseñado para ejecutar consultas en la propia capa.

Ver también

PostgreSQL ejecuta SQL, *Ejecutar SQL*

For some SQL query examples see *PostgreSQL Query Examples*.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Base de datos	`DATABASE`	[enumeración] Predeterminado: no configurada	Seleccione una base de datos SQLite/SpatiaLite conectada a la sesión actual
**Consulta SQL*	`SQL`	[cadena de texto] Predeterminado: ""	Define la consulta SQL, por ejemplo `'UPDATE my_table SET field=10'`.

Salidas

No se crea ninguna salida. La consulta SQL se ejecuta en su lugar.

Código Python

Algoritmo ID: `native:spatialiteexecutesqlregistered`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.5 Herramientas de Archivo

Descargar archivo a través de HTTP(S)

Descarga una URL al sistema de archivos con una petición HTTP(S) GET o POST.

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
URL	`URL`	[cadena de texto]	La URL del archivo a descargar.
File destination	`OUTPUT`	[cadena de texto] Predeterminado: `[Save to temporary file]`	Especificación del destino del archivo. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Methodo	`METHOD`	[enumeración] Predeterminado: 0	El método HTTP que se usará para la solicitud. Las opciones son: • 0 — OBTENER • 1 — PUBLICAR
Datos Opcional	`DATA`	[cadena de texto]	Los datos para agregar en el cuerpo si la solicitud es un POST.

Salidas

Etiqueta	Nombre	Tipo	Descripción
File destination	`OUTPUT`	[cadena de texto]	La localización del archivo descargado

Código Python

Algorithm ID: `native:filedownloader`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Solicitud HTTP(S) POST/GET

Realiza una petición HTTP(S) POST/GET y devuelve el código de estado HTTP y los datos de respuesta. Si se produce un error, se devuelve el código de error y el mensaje. Opcionalmente, el resultado puede escribirse en un fichero del disco. Por defecto, el algoritmo avisa en caso de error. Opcionalmente, el algoritmo puede ser configurado para tratar los errores HTTP como fallos.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
URL o ruta de archivo	`URL`	[cadena de texto]	La URL o ruta del archivo a abrir.
Methodo	`METHOD`	[enumeración] Predeterminado: 0	El método HTTP que se usará para la solicitud. Las opciones son: • 0 — OBTENER • 1 — PUBLICAR
Datos POST Opcional	`DATA`	[cadena de texto]	Los datos para agregar en el cuerpo si la solicitud es un POST.
Autenticación Opcional	`AUTH_CONFIG`	[authconfig] Predeterminado: Sin autenticación	Una configuración de autenticación para pasar
Considere los errores HTTP como fallos	`FAIL_ON_ERROR`	[booleano] Predeterminado: False	Si se establece, el algoritmo fallará al encontrar un error HTTP.
File destination Opcional	`OUTPUT`	[cadena de texto] Predeterminado: `[Omitir salida]`	El resultado puede escribirse en un archivo en lugar de devolverse como cadena. Especificación del destino del archivo. *Uno de*: • Saltar salida • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
File destination	`OUTPUT`	[archivo]	El archivo descargado con el resultado devuelto
Estado HTTP	`STATUS_CODE`	[cadena de texto]	El estado HTTP
Código de error de red	`ERROR_CODE`	[cadena de texto]	El código de error cuando falla (como también en 404 HTTP Estado etc.)
Mensaje de error de red	`ERROR_MESSAGE`	[cadena de texto]	Una cadena que contiene el mensaje de error en caso de fallo
Datos de resultados	`RESULT_DATA`	[cadena de texto]	Una cadena que contiene los datos del resultado en caso de éxito

Código Python

ID del algoritmo: `native:httprequest`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Abrir archivo o URL

Abre archivos en su aplicación asociada por defecto, o URLs en el navegador web por defecto del usuario.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
URL o ruta de archivo	`URL`	[cadena de texto]	La URL o ruta del archivo a abrir.

Salidas

El algoritmo no tiene salida.

Código Python

ID del algoritmo: `native:openurl`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.6 Fix Geometry

Add missing vertices along borders

Adds the missing vertices along polygons borders, based on an error layer from the *Missing vertices along borders* algorithm.

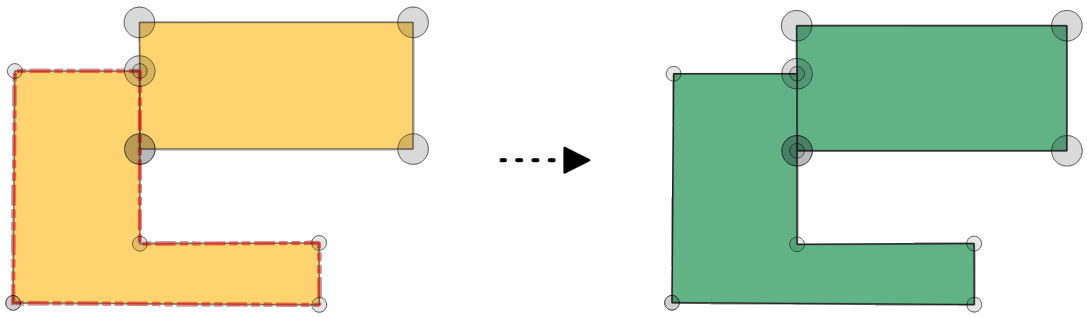


Figura 24.16: Fixing missing vertices along polygon borders.

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vector: point]	Point layer with the errors location, generated by the check algorithm.
Field of original feature unique identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification in input layer, as set in the check algorithm.
Field of part index	`PART_IDX`	[tablefield: integer] Default: `gc_partidx`	Field storing the erroneous feature's geometry part number.
Field of ring index	`RING_IDX`	[tablefield: integer] Default: `gc_ringidx`	Field storing the erroneous feature's geometry ring number.
Field of vertex index	`VERTEX_IDX`	[tablefield: integer] Default: `gc_vertidx`	Field storing the erroneous feature's geometry vertex number.
Border vertices fixed layer	`OUTPUT`	[vector: polygon] Default: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

continúe en la próxima página

Tabla 24.115 – proviene de la página anterior

Label	Name	Type	Description
Report layer from fixing border vertices	`REPORT`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Report layer from fixing border vertices	`REPORT`	[vector: point]	Output point layer representing the error locations and fix applied. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the missing vertex. • `gc_errory`: the y coordinate of the missing vertex. • `gc_error` • `report`: a text field describing the fix applied. • `error_fixed`: a boolean field indicating whether the error was fixed. • `UNIQUE_ID` field: the unique ID of the input feature that has a missing vertex.
Border vertices fixed layer	`OUTPUT`	[vector: polygon]	Output layer with the geometry fix applied to the input features. The layer contains the same fields as in the input layer.

Python code

Algorithm ID: `native:fixgeometrymissingvertex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Convert to strictly multipart

Converts multipart geometries that consists of only one geometry into singlepart geometries, based on an error layer from the *Strictly multipart* algorithm. Layer geometry type will not be changed and will remain multipart.

Ver también

Elevar a multiparte

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line, polygon]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vector: point]	Point layer with the errors location, generated by the check algorithm.
Field of original feature unique identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification in input layer, as set in the check algorithm.
Strictly-multipart layer	`OUTPUT`	[vector: same as input] Default: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Report layer from fixing multipart	`REPORT`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the centroids of the fixed geometries. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Report layer from fixing multipart	`REPORT`	[vector: point]	Output point layer representing the error locations and fix applied. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the multipart geometry that has only one part. • `gc_errory`: the y coordinate of the centroid of the multipart geometry that has only one part. • `gc_error` • `report`: a text field describing the fix applied. • `error_fixed`: a boolean field indicating whether the error was fixed. • `UNIQUE_ID` field: the unique ID of the input feature that is multipart but has only one part.
Strictly-multipart layer	`OUTPUT`	[vector: same as input]	Output layer with the geometry fix applied to the input features. The layer contains the same fields as in the input layer.

Python code

Algorithm ID: `native:fixgeometrymultipart`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Delete duplicated vertices

Deletes duplicated vertices from the input geometries, based on errors reported by the *Duplicated vertices* algorithm.

Ver también

Eliminar vértices duplicados

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line, polygon]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vector: point]	Point layer with the errors location, generated by the check algorithm.
Field of original feature unique identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification in input layer, as set in the check algorithm.
Field of part index	`PART_IDX`	[tablefield: integer] Default: `gc_partidx`	Field storing the erroneous feature's geometry part number.
Field of ring index	`RING_IDX`	[tablefield: integer] Default: `gc_ringidx`	Field storing the erroneous feature's geometry ring number.
Field of vertex index	`VERTEX_IDX`	[tablefield: integer] Default: `gc_vertidx`	Field storing the erroneous feature's geometry vertex number.
Fixed duplicate vertices layer	`OUTPUT`	[vector: same as input] Default: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Report layer from fixing duplicate vertices	`REPORT`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Report layer from fixing duplicate vertices	`REPORT`	[vector: point]	Output point layer representing the error locations and fix applied. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part containing the duplicate vertex. • `gc_ringidx`: the index of the feature's geometry ring containing the duplicate vertex. • `gc_vertidx`: the index of the duplicate vertex in the feature's geometry ring. • `gc_errorx`: the x coordinate of the duplicate vertex. • `gc_errory`: the y coordinate of the duplicate vertex. • `gc_error` • `report`: a text field describing the fix applied. • `error_fixed`: a boolean field indicating whether the error was fixed. • `UNIQUE_ID` field: the unique ID of the input feature that has duplicate vertices.
Fixed duplicate vertices layer	`OUTPUT`	[vector: same as input]	Output layer with the geometry fix applied to the input features. The layer contains the same fields as in the input layer.

Python code

Algorithm ID: `native:fixgeometryduplicatenodes`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Delete features

Deletes error features based on an error layer from some check geometry algorithms, such as:

- *Features inside polygon*
- *Degenerate polygons*
- *Small segments*
- *Duplicated geometries*

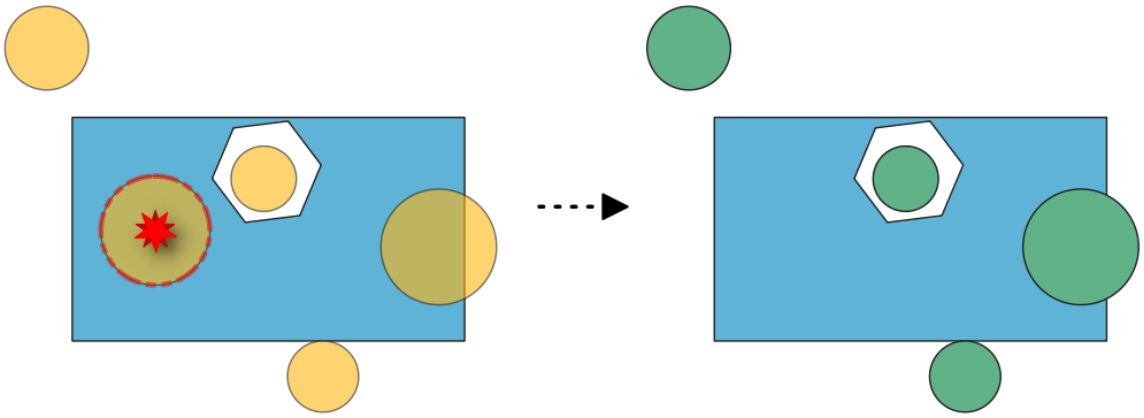


Figura 24.17: Fixing errors from the «*Features inside polygon*» algorithm check.

Ver también

Borrar geometrías duplicadas, *Descartar geometrías*

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: geometry]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vector: point]	Point layer with the errors location, generated by the check algorithm.

continúe en la próxima página

Tabla 24.124 – proviene de la página anterior

Label	Name	Type	Description
Field of original feature unique identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification in input layer, as set in the check algorithm.
Cleaned layer	`OUTPUT`	[vector: same as input] Default: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Report layer from deleting features	`REPORT`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Outputs

Label	Name	Type	Description
Report layer from deleting features	`REPORT`	[vector: point]	Output point layer representing the error locations and fix applied. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the erroneous geometry. • `gc_errory`: the y coordinate of the centroid of the erroneous geometry. • `gc_error`: the index of the feature where the error belongs. • `report`: a text field describing the fix applied. • `error_fixed`: a boolean field indicating whether the error was fixed. • `UNIQUE_ID` field: the unique ID of the input feature with the error.
Cleaned layer	`OUTPUT`	[vector: same as input]	Output layer with features removed based on detected errors. The layer contains the same fields as in the input layer.

Python code

Algorithm ID: `native:fixgeometrydeletefeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Delete overlaps

Deletes overlapping areas based on an error layer from the *Overlaps* algorithm.

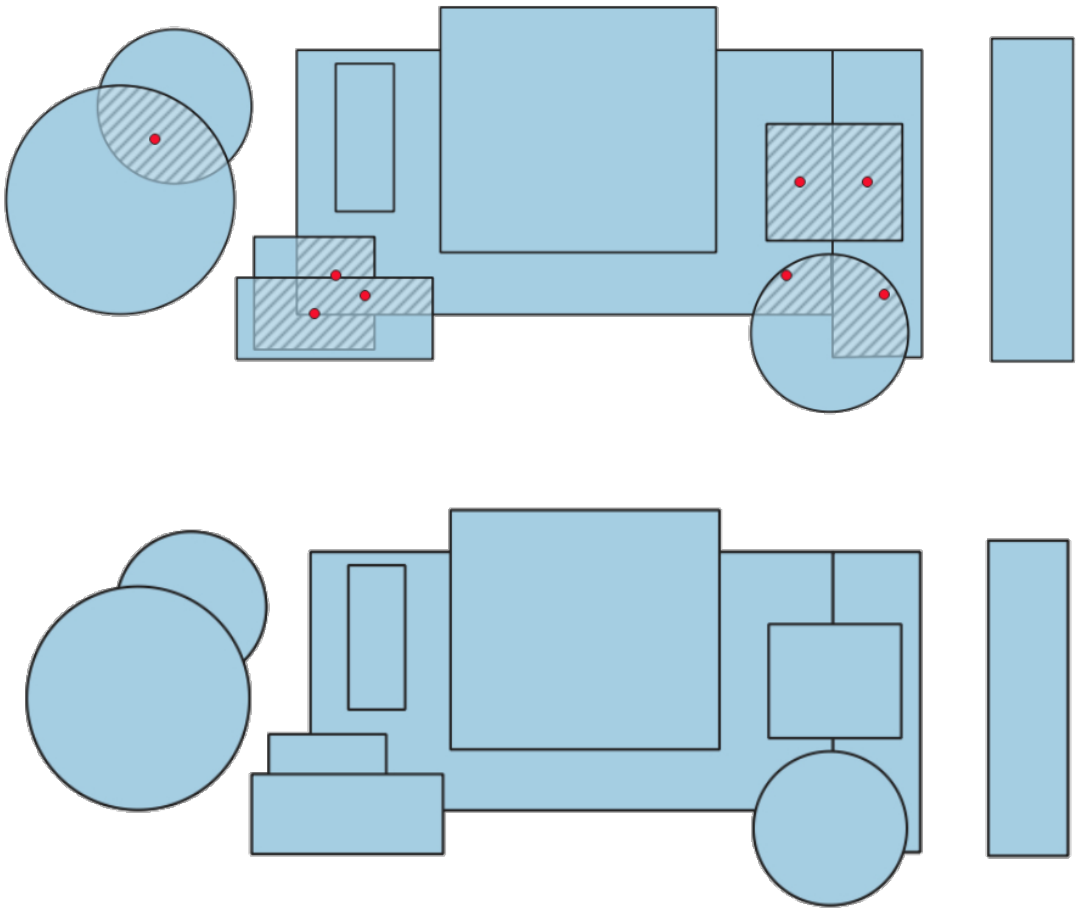


Figura 24.18: Deleting overlapping areas.

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vector: point]	Point layer with the errors location, generated by the check algorithm.
Field of original feature unique identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification in input layer, as set in the check algorithm.
Field of overlap feature unique identifier	`OVERLAP_FEATUR`	[tablefield: any]	Field storing unique values for the overlapping feature identification in input layer, as set in the check algorithm.
Field of error value	`ERROR_VALUE_ID`	[tablefield: any]	Field storing the error value for each feature, as set in the check algorithm.
No-overlap layer	`OUTPUT`	[vector: polygon] Default: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Report layer from fixing overlaps	`REPORT`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Report layer from fixing overlaps	`REPORT`	[vector: point]	Output point layer representing the error locations and fix applied. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_errorx`: the x coordinate of the centroid of the overlapping area. • `gc_errory`: the y coordinate of the centroid of the overlapping area. • `gc_error`: the area of the overlapping geometry. • `UNIQUE_ID` field: the unique ID of the unique ID of an overlapped input feature. • `gc_overlap_feature_{unique_id}`: the `UNIQUE_ID` field value for the other overlapping feature. • `report`: a text field describing the fix applied or justifying the failure. Possible values are: – Remove overlapping area from neighboring polygon with shortest shared edge – Error is obsolete • `error_fixed`: a boolean field indicating whether the error was fixed.
No-overlap layer	`OUTPUT`	[vector: polygon]	Output layer with input features edited. Overlapping areas reported as errors are removed. The layer contains the same fields as in the input layer.

Python code

Algorithm ID: `native:fixgeometryoverlap`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Delete small angles

Deletes vertices based on an error layer from the *Small angles* algorithm. When deletion of a vertex results in a duplicate vertex (when a spike vertex is deleted), the duplicate vertex is deleted to keep a single vertex and preserve topology.

> **Atención**
>
> This algorithm removes the vertex at the reported small angles, generating new segments that may form a new small angle.

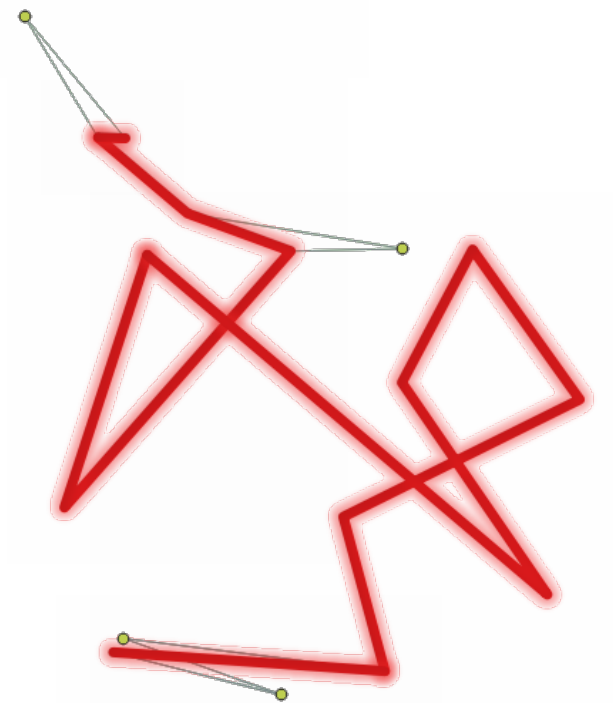


Figura 24.19: Fixing reported errors on a line feature for angles lower than 15°.

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line, polygon]	Layer with the geometries to fix, same as input in the check algorithm
Error layer	`ERRORS`	[vector: point]	Point layer with the errors location, generated by the check algorithm
Field of original feature unique identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification in input layer, as set in the check algorithm
Field of part index	`PART_IDX`	[tablefield: integer] Default: `gc_partidx`	Field storing the erroneous feature's geometry part number.

continúe en la próxima página

Tabla 24.129 – proviene de la página anterior

Label	Name	Type	Description
Field of ring index	`RING_IDX`	[tablefield: integer] Default: `gc_ringidx`	Field storing the erroneous feature's geometry ring number.
Field of vertex index	`VERTEX_IDX`	[tablefield: integer] Default: `gc_vertidx`	Field storing the erroneous feature's geometry vertex number.
Small angle fixed layer	`OUTPUT`	[vector: same as input] Default: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Report layer from fixing small angles	`REPORT`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Report layer from fixing small angles	`REPORT`	[vector: point]	Output point layer representing the error locations and fix applied. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part containing the small angle. • `gc_ringidx`: the index of the feature's geometry ring containing the small angle. • `gc_vertidx`: the index of the vertex with the small angle in the feature's geometry ring. • `gc_errorx`: the x coordinate of the vertex with the small angle. • `gc_errory`: the y coordinate of the vertex with the small angle. • `gc_error`: the error angle value. • `UNIQUE_ID` field: the unique ID of the input feature with the small angle. • `report`: a text field describing the fix applied or justifying the failure. • `error_fixed`: a boolean field indicating whether the error was fixed.
Small angle fixed layer	`OUTPUT`	[same as input]	Output layer with the geometry fix applied to the input features The layer contains the same fields as in the input layer.

Python code

Algorithm ID: `native:fixgeometryangle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Fill gaps

Fills the gaps based on a gap and neighbors layer from the *Small gaps* algorithm. Three different fixing methods are available.

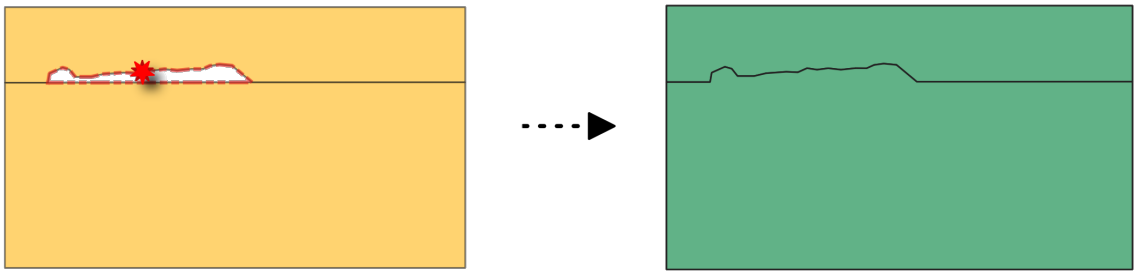


Figura 24.20: Fixing gap between polygon features using the `Add to largest neighbouring area` method.

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to fix, same as input in the check algorithm.
Neighbors layer	`NEIGHBORS`	[vector: table]	Relational table with the unique ID of the gap and its neighbors in input layer, generated by the check algorithm.
Gaps layer	`GAPS`	[vector: polygon]	Polygon layer with the gaps geometries, generated by the check algorithm.
Method	`METHOD`	[enumeration] Default: 0	Method to apply to the gap fixing. Possible values are: • 0 — `Add to longest shared edge`: adds the gap to the neighboring feature with the longest shared edge. • 1 — `Create new feature`: creates a new feature for each gap. • 2 — `Add to largest neighbouring area`: adds the gap to the neighboring feature with the largest area.
Field of original feature unique identifier	`UNIQUE_ID`	[tablefield: any] Default: gc_errorid	Field storing unique values for feature identification in input layer, as set in the check algorithm.
Field of error id	`ERROR_ID_IDX`	[tablefield: any]	Field storing the error ID for each gap, as output by the check algorithm.

continúe en la próxima página

Tabla 24.132 – proviene de la página anterior

Label	Name	Type	Description
Gaps-filled layer	`OUTPUT`	[vector: polygon] Default: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Report layer from fixing gaps	`REPORT`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Report layer from fixing gaps	`REPORT`	[vector: point]	Output point layer representing the error locations and fix applied. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the gap. • `gc_errory`: the y coordinate of the centroid of the gap. • `gc_error`: the area of the gap. • `gc_errorid`: the ID of the gap. • `report`: a text field mentioning the method used to fix the gap. • `error_fixed`: a boolean field indicating whether the error was fixed.
Gaps-filled layer	`OUTPUT`	[vector: polygon]	Output layer of polygons without gaps. The layer contains the same fields as in the input layer.

Python code

Algorithm ID: `native:fixgeometrygap`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Fill holes

Deletes holes in polygon geometries based on an error layer from the *Holes* algorithm.

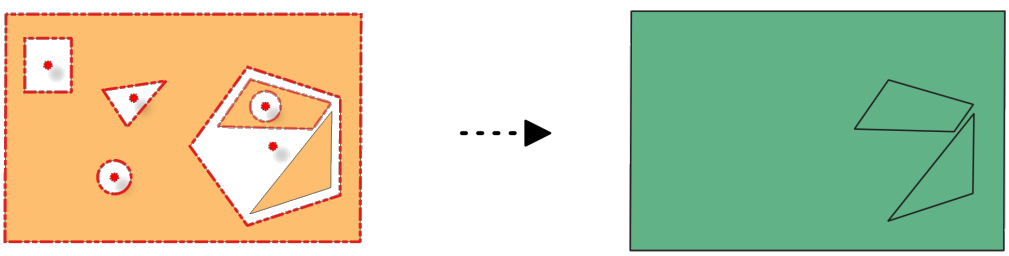


Figura 24.21: Deleting holes in a polygon feature.

Ver también

Borrar agujeros

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line, polygon]	Layer with the geometries to fix, same as input in the check algorithm
Error layer	`ERRORS`	[vector: point]	Point layer with the errors location, generated by the check algorithm
Field of original feature unique identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification in input layer, as set in the check algorithm
Field of part index	`PART_IDX`	[tablefield: integer] Default: `gc_partidx`	Field storing the erroneous feature's geometry part number.
Field of ring index	`RING_IDX`	[tablefield: integer] Default: `gc_ringidx`	Field storing the erroneous feature's geometry ring number.
Field of vertex index	`VERTEX_IDX`	[tablefield: integer] Default: `gc_vertidx`	Field storing the erroneous feature's geometry vertex number.
Holes-filled layer	`OUTPUT`	[vector: polygon] Default: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Report layer from fixing holes	`REPORT`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Report layer from fixing holes	`REPORT`	[vector: point]	Output point layer representing the error locations and fix applied. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part containing the hole. • `gc_ringidx`: the index of the feature's geometry ring containing the hole. • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the hole. • `gc_errory`: the y coordinate of the centroid of the hole. • `gc_error` • `UNIQUE_ID` field: the unique ID of the input feature that has a hole. • `report`: a text field describing the fix applied or justifying the failure. • `error_fixed`: a boolean field indicating whether the error was fixed.
Holes-filled layer	`OUTPUT`	[vector: polygon]	Output layer of polygons without holes. The layer contains the same fields as in the input layer.

Python code

Algorithm ID: `native:fixgeometryhole`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Fix small polygons

Merges neighboring polygons according to the chosen method, based on an error layer from the *Small polygons* or *Sliver polygons* algorithm.

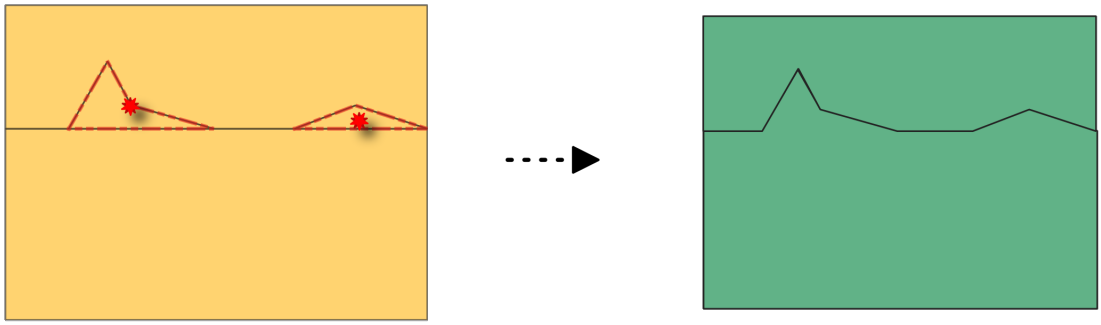


Figura 24.22: Before and after fixing small polygons (method: merge with neighboring polygon with largest area).

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: polygon]	Layer with the geometries to fix, same as input in the check algorithm.
Error layer	`ERRORS`	[vector: point]	Point layer with the errors location, generated by the check algorithm.
Method	`METHOD`	[enumeration] Default: 0	Method to apply to the small polygon geometries: • 0 — `Merge with neighboring polygon with longest shared edge`: merges the small polygon geometry with the neighboring polygon that has the longest shared edge. • 1 — `Merge with neighboring polygon with largest area`: merges the small polygon geometry with the neighboring polygon that has the largest area. • 2 — `Merge with neighboring polygon with identical attribute value, if any, or leave as is`: merges the small polygon geometry with a neighboring polygon that has the same attribute value, if such a polygon exists.

continúe en la próxima página

Tabla 24.138 – proviene de la página anterior

Label	Name	Type	Description
Field to consider when merging polygons with the identical attribute method Optional	`MERGE_ATTRIBUT`	[tablefield: any]	Field in the input layer to consider when merging polygons with the identical attribute method.
Field of original feature unique identifier	`UNIQUE_ID`	[tablefield: any]	Field storing unique values for feature identification in input layer, as set in the check algorithm.
Field of part index	`PART_IDX`	[tablefield: integer] Default: `gc_partidx`	Field storing the erroneous feature's geometry part number.
Field of ring index	`RING_IDX`	[tablefield: integer] Default: `gc_ringidx`	Field storing the erroneous feature's geometry ring number.
Field of vertex index	`VERTEX_IDX`	[tablefield: integer] Default: `gc_vertidx`	Field storing the erroneous feature's geometry vertex number.
Small polygons merged layer	`OUTPUT`	[vector: polygon] Default: `[Create temporary layer]`	Specification of the output layer containing fixed features. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Report layer from merging small polygons	`REPORT`	[vector: point] Default: `[Create temporary layer]`	Specification of the output layer containing the fixes location. *One of*: • Create Temporary Layer (`TEMPORARY_OUTPUT`) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Report layer from merging small polygons	`REPORT`	[vector: point]	Output point layer representing the error locations and fix applied. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx` • `gc_ringidx` • `gc_vertidx` • `gc_errorx`: the x coordinate of the centroid of the small polygon. • `gc_errory`: the y coordinate of the centroid of the small polygon. • `gc_error`: the area of the small polygon. • `UNIQUE_ID` field: the unique ID of the input feature that is a small polygon. • `report`: a text field describing the fix applied or justifying the failure. • `error_fixed`: a boolean field indicating whether the error was fixed.
Small polygons merged layer	`OUTPUT`	[vector: polygon]	Output layer of polygons with small polygons merged according to the chosen method. The layer contains the same fields as in the input layer.

Python code

Algorithm ID: `native:fixgeometryarea`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Split self-intersecting geometries

Splits self-intersecting geometries based on an error layer from the *Self-intersections* algorithm.

Nota

This algorithm detects invalid polygon geometries (self-intersections). To ensure these invalid features are not filtered out before processing, open Advanced options next to the *Input layer* and set *Invalid feature filtering* to `Do not Filter (Better Performance)`.

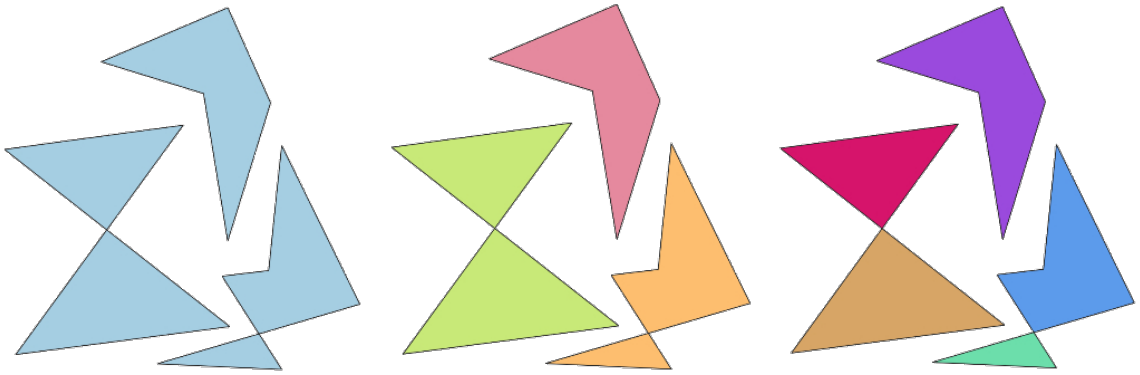


Figura 24.23: Fixing self-intersection using various methods: as multi-part (middle), and multiple single-part geometries (right).

Parameters

Basic parameters

Label	Name	Type	Description
Input layer	`INPUT`	[vector: line, polygon]	Layer with the geometries to fix, same as input in the check algorithm
Error layer	`ERRORS`	[vector: point]	Point layer with the errors location, generated by the check algorithm

continúe en la próxima página

Tabla 24.141 – proviene de la página anterior

Label	Name	Type	Description
Method	METHOD	[enumeration]	Method to apply to the self-intersecting geometries: • 0 – Split feature into a multi-part feature: splits the self-intersecting geometry into multiple parts stored in a single multipart feature. • 1 – Split feature into multiple single-part features: splits the self-intersecting geometry into multiple parts, each part stored in a separate feature.
Field of original feature unique identifier	UNIQUE_ID	[tablefield: any]	Field storing unique values for feature identification in input layer, as set in the check algorithm
Field of part index	PART_IDX	[tablefield: integer] Default: gc_partidx	Field storing the erroneous feature's geometry part number.
Field of ring index	RING_IDX	[tablefield: integer] Default: gc_ringidx	Field storing the erroneous feature's geometry ring number.
Field of vertex index	VERTEX_IDX	[tablefield: integer] Default: gc_vertidx	Field storing the erroneous feature's geometry vertex number.
Field of segment 1	SEGMENT_1	[tablefield: integer] Default: gc_segment_1	Field storing the number of the first segment involved in the erroneous intersection.
Field of segment 2	SEGMENT_2	[tablefield: integer] Default: gc_segment_2	Field storing the number of the second segment involved in the erroneous intersection.
Self-intersections fixed layer	OUTPUT	[vector: same as input] Default: [Create temporary layer]	Specification of the output layer containing fixed features. *One of*: • Create Temporary Layer (TEMPORARY_OUTPUT) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.
Report layer from fixing self-intersections	REPORT	[vector: point] Default: [Create temporary layer]	Specification of the output layer containing the fixes location. *One of*: • Create Temporary Layer (TEMPORARY_OUTPUT) • Save to File… • Save to Geopackage… • Save to Database Table… The file encoding can also be changed here.

Advanced parameters

Label	Name	Type	Description
Tolerance	`TOLERANCE`	[numeric: integer] Default: 8	Numerical precision of geometric operations, given as an integer n, meaning that two vertices less than 10^{-n} apart (in map units) are considered to be merged.

Outputs

Label	Name	Type	Description
Report layer from fixing self-intersections	`REPORT`	[vector: point]	Output point layer representing the error locations and fix applied. The output layer contains the following fields: • `gc_layerid`: the ID of the input layer. • `gc_layername`: the name of the input layer. • `gc_partidx`: the index of the feature's geometry part where the self-intersection occurs. • `gc_ringidx`: the index of the feature's geometry ring where the self-intersection occurs. • `gc_vertidx` • `gc_errorx`: the x coordinate of the self-intersection. • `gc_errory`: the y coordinate of the self-intersection. • `gc_error` • `gc_segment_1`: the index of the first segment involved in the intersection. • `gc_segment_2`: the index of the second segment involved in the intersection. • `UNIQUE_ID` field: the unique ID of the self-intersecting input feature. • `report`: a text field describing the fix applied. • `error_fixed`: a boolean field indicating whether the error was fixed.
Self-intersections fixed layer	`OUTPUT`	[vector: same as input]	Output layer with the geometry fix applied to the input features. The layer contains the same fields as in the input layer.

Python code

Algorithm ID: `native:fixgeometryselfintersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

24.1.7 GPS

Convertir datos GPS

Utiliza la herramienta GPSBabel para convertir un archivo de datos GPS de una serie de formatos al formato estándar GPX.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Archivo de entrada	`INPUT`	[archivo]	Archivo que contiene los datos a convertir
Formato	`FORMAT`	[enumeración]	Formato del archivo a convertir, de esta lista.
Tipo de objeto espacial	`FEATURE_TYPE`	[enumeración] Predeterminado: 0	El tipo de datos a convertir • 0 — Puntos de vía • 1 — Rutas • 2 — Trazas
Salida	`OUTPUT`	[vectorial: cualquiera] Predeterminado: `[Save to temporary file]`	Especificación del archivo GPX de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`CAPA_SALIDA`	[vectorial: cualquiera]	Capa de salida con datos en formato estándar GPX

Código Python

Algorithm ID: `native:convertgpsdata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Convertir tipo de objeto espacial GPX

Utiliza la herramienta GPSBabel para convertir características GPX de un tipo a otro (por ejemplo, convertir todas los objetos espaciales de punto de vía a una ruta).

Parametros

Etiqueta	Nombre	Tipo	Descripción
Archivo de entrada	`INPUT`	[archivo]	Archivo que contiene los datos a convertir
Conversión	`CONVERSIÓN`	[enumeración] Predeterminado: 0	El tipo de conversión a aplicar • 0 — Puntos de vía a partir de una ruta • 1 — Puntos de vía a partir de una traza • 2 — Rutas a partir de puntos de vía • 3 — Trazas a partir de puntos de vía
Salida	`OUTPUT`	[vector: punto o línea] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Salida	`OUTPUT`	[vectorial: cualquiera]	Capa de salida con objetos espaciales GPX convertidos

Código Python

Algorithm ID: `native:convertgpxfeaturetype`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Descargar datos GPS desde dispositivo

Utiliza la herramienta GPSBabel para descargar los datos de un dispositivo GPS en el formato estándar GPX.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Dispositivo	`DISPOSITIVO`	[enumeración] Predeterminado:*Garmin serial*	El dispositivo GPS utilizado para crear los datos. Debe declararse en el cuadro de diálogo *Configuración GPS*.
Puerto	`PORT`	[enumeración]	El puerto al que está conectado el dispositivo. Los puertos disponibles dependen del sistema operativo.
Tipo de objeto espacial	`FEATURE_TYPE`	[enumeración] Predeterminado: 0	El tipo de datos a convertir • 0 — Puntos de vía • 1 — Rutas • 2 — Trazas
Salida	`OUTPUT`	[vectorial: cualquiera] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`CAPA_SALIDA`	[vectorial: cualquiera]	Capa de salida con datos en formato estándar GPX

Código Python

Algorithm ID: `native:downloadgpsdata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Cargar los datos del GPS en el dispositivo

Utiliza la herramienta GPSBabel para cargar datos en un dispositivo GPS a partir del formato estándar GPX.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Archivo de entrada	`INPUT`	[archivo]	`.GPX` archivo que contiene los datos a cargar
Dispositivo	`DISPOSITIVO`	[enumeración] Predeterminado:*Garmin serial*	El dispositivo GPS al que desea cargar los datos. Debe declararse en el cuadro de diálogo *Configuración GPS*.
Puerto	`PORT`	[enumeración]	El puerto al que está conectado el dispositivo. Los puertos disponibles dependen del sistema operativo.
Tipo de objeto espacial	`FEATURE_TYPE`	[enumeración] Predeterminado: 0	El tipo de datos a cargar • 0 — Puntos de vía • 1 — Rutas • 2 — Trazas

Salidas

No se proporciona ninguna salida. Si tiene éxito, los datos se cargan en el dispositivo.

Código Python

Algorithm ID: `native:uploadgpsdata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.8 Interpolación

Mapa de calor (estimación de densidad kernel)

Crea un ráster de densidad (mapa de calor) de una capa vectorial de puntos entrante usando estimación de densidad kernel.

La densidad se calcula en función del número de puntos en una ubicación, con un mayor número de puntos agrupados que dan como resultado valores más grandes. Los mapas de calor permiten una fácil identificación de *puntos calientes* y la agrupación de puntos.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de puntos	`INPUT`	[vector: punto]	Capa vectorial de puntos a usar para el mapa de calor
Radio	`RADIUS`	[numérico: doble] Predeterminado: 100.0	Radio de búsqueda de mapa de calor (o ancho de banda del kernel) en unidades de mapa. El radio especifica la distancia alrededor de un punto en el que se sentirá la influencia del punto. Los valores más grandes dan como resultado un mayor suavizado, pero los valores más pequeños pueden mostrar detalles más finos y variaciones en la densidad de puntos.
Tamaño del ráster de salida	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 0.1	Tamaño de píxel de la capa ráster de salida en unidades de capa. En la GUI, el tamaño se puede especificar por el número de filas (`Número de filas`)/ columnas(`Número de columnas`) **o** el tamaño de píxel (`Tamaño de píxel X` / `Tamaño de píxel Y`). El aumento del número de filas o columnas disminuirá el tamaño de la celda y aumentará el tamaño del archivo del ráster de salida. Los valores en `Filas`, `Columnas`, `Tamaño de píxel X` y `Tamaño de píxel Y` se actualizarán simultáneamente; al duplicar el número de filas, se duplicará el número de columnas, y la celda el tamaño se reducirá a la mitad. La extensión del ráster de salida seguirá siendo la misma (aproximadamente).
Radio a partir de campo Opcional	`RADIUS_FIELD`	[campo de tabla: numérico]	Establece el radio de búsqueda para cada entidad a partir de un atributo en la capa entrante.
Peso a partir de un campo Opcional	`WEIGHT_FIELD`	[campo de tabla: numérico]	Permite a las entidades entrantes ser ponderadas por un atributo de campo. Esto puede ser empleado para incrementar la influencia que ciertas entidades tienen en el mapa de calor resultante.

continúe en la próxima página

Tabla 24.151 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Forma kernel	`KERNEL`	[enumeración] Predeterminado: *0*	Controla la velocidad a la que la influencia de un punto disminuye a medida que aumenta la distancia desde el punto. Los diferentes núcleos se descomponen a diferentes velocidades, por lo que un núcleo de tres pesos da a las entidades un mayor peso para distancias más cercanas al punto que el kernel de Epanechnikov. En consecuencia, el peso triple da como resultado puntos calientes «más nítidos» y Epanechnikov da como resultado puntos calientes «más suaves». Hay muchas formas disponibles (consulte la página de Wikipedia <https://en.wikipedia.org/wiki/Kernel_(statistics)#Kernel_functions_in_common_use>`_ para mas información): • 0 — Cuartico • 1 — Triangular • 2 — Uniforme • 3 — Triweight • 4 — Epanechnikov
Relación de decaimiento (solo kernels triangulares) Opcional	`DECAY`	[numérico: doble] Predetermminado: *0.0*	Se puede usar con núcleos triangulares para controlar aún más cómo el calor de una entidad disminuye con la distancia a la entidad. • Un valor de 0(=mínimo) indica que el calor se concentrará en el centro del radio dado y se extinguirá completamente en el borde. • Un valor de 0,5 indica que los píxeles en el borde del radio recibirán la mitad de calor que los píxeles en el centro del radio de búsqueda. • Un valor de 1 significa que el calor se distribuye uniformemente por todo el círculo del radio de búsqueda. (Esto es equivalente al kernel “uniforme”). • Un valor mayor que 1 indica que el calor es mayor hacia el borde del radio de búsqueda que hacia el centro.
Escala de valor de salida	`OUTPUT_VALUE`	[enumeración] Predeterminado: *0*	Permitir cambiar los valores del ráster del mapa de calor de salida. Uno de: • 0 — Bruto • 1 — Escalado
Mapa de calor	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida con valores de densidad del núcleo. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Mapa de calor	`OUTPUT`	[ráster]	Capa ráster con valores de densidad de kernel

Ejemplo: Creación de una mapa de calor

Para el siguiente ejemplo, usaremos la capa vectorial de puntos `airports` del conjunto de datos de muestra de QGIS (ver *Descargando datos de muestra*). Otro excelente tutorial de QGIS sobre cómo hacer mapas de calor se puede encontrar en http://qgistutorials.com.

En Figura 24.24, se muestran los aeropuertos de Alaska.

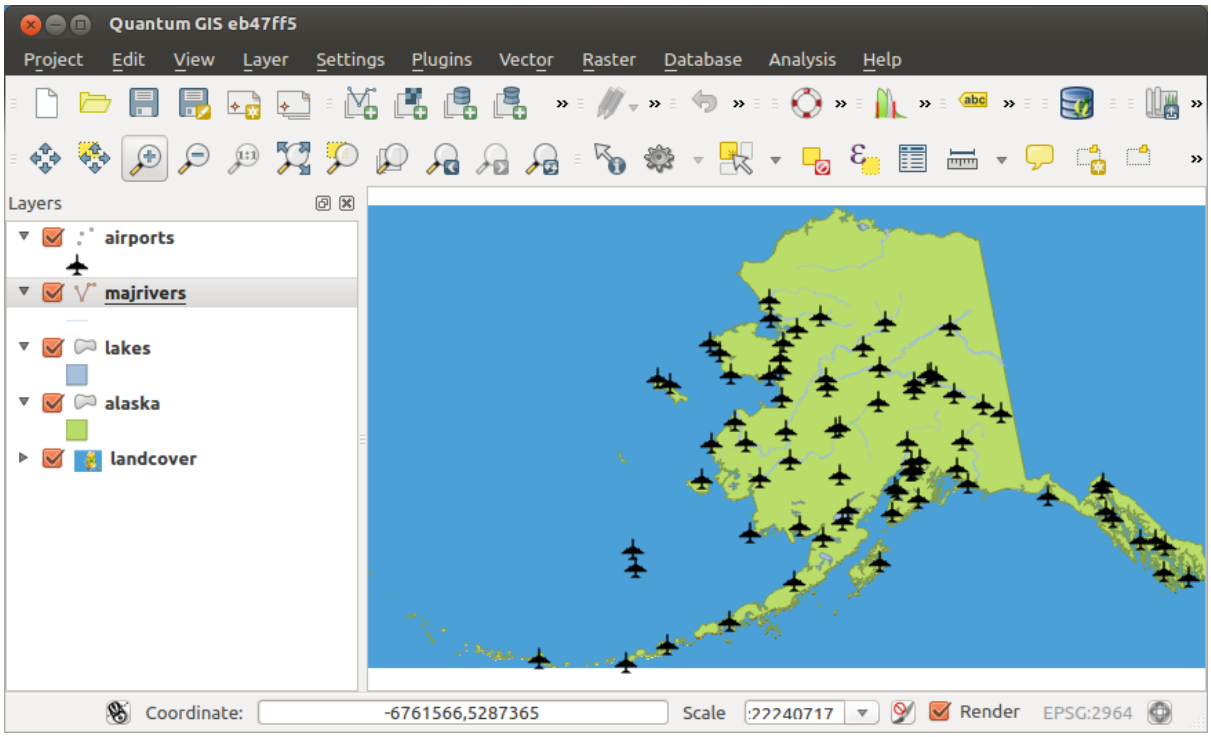


Figura 24.24: Aeropuertos de Alaska

1. Abra el algoritmo del QGIS *Mapa de Calor (Estimación de Densidad Kernel)* del grupo *Interpolación*
2. En el campo *capa de Puntos* , seleccione `airports` de la lista de capas de puntos cargadas en el proyecto actual.
3. Cambie el *Radio* a `1000000` metros.
4. Cambie el *tamaño de Pixel X* a `1000`. El *tamaño de Pixel Y*, *Filas* y *Columnas* se actualizarán automáticamente.
5. Click en *Run* para crear y cargar el mapa de calor de aeropuertos (ver Figura 24.26).

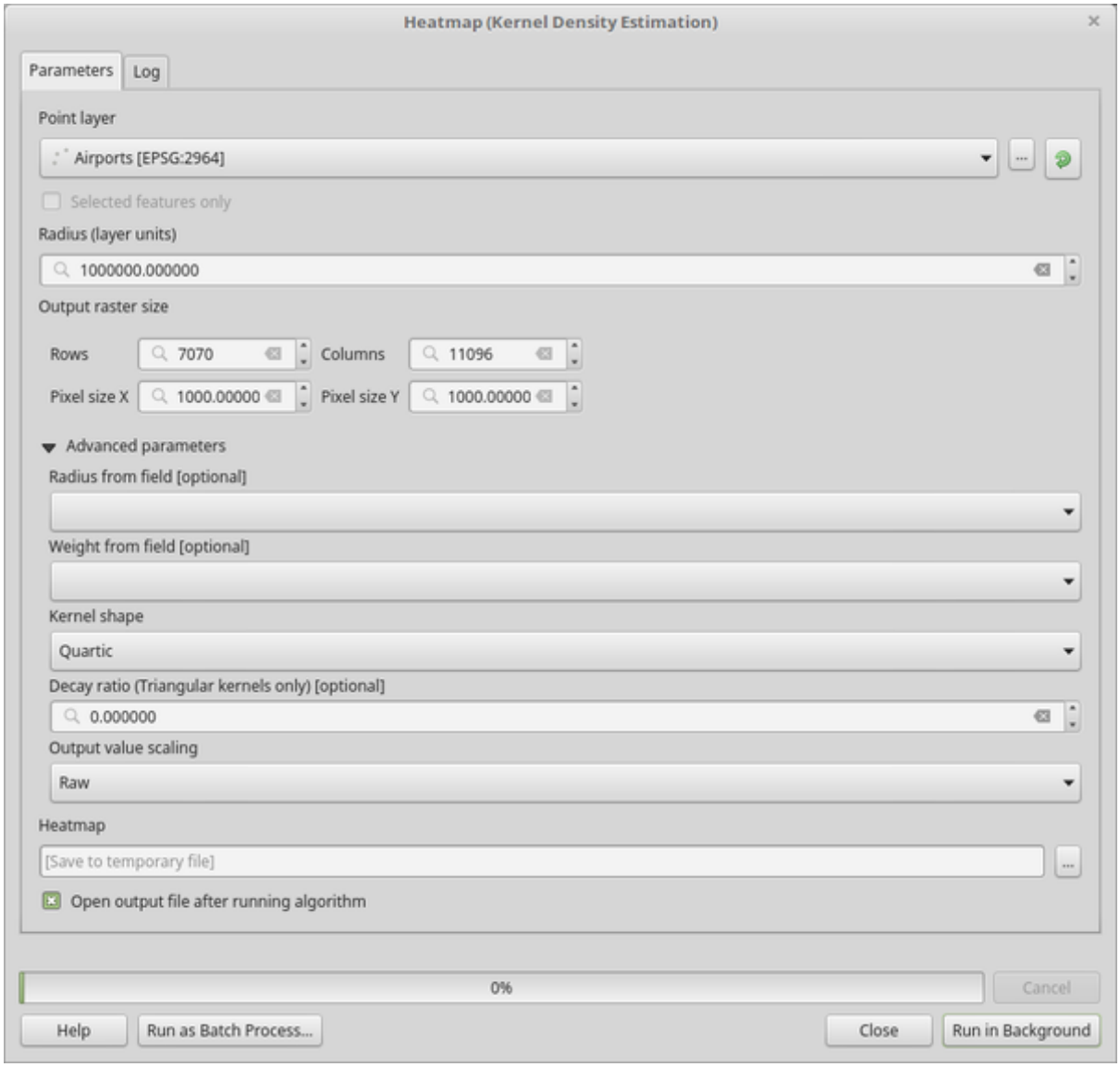


Figura 24.25: El Diálogo Mapa de Calor

QGIS generará el mapa de calor y lo agregará a su ventana de mapa. De forma predeterminada, el mapa de calor está sombreado en escala de grises, y las áreas más claras muestran concentraciones más altas de aeropuertos. El mapa de calor ahora se puede diseñar en QGIS para mejorar su apariencia.

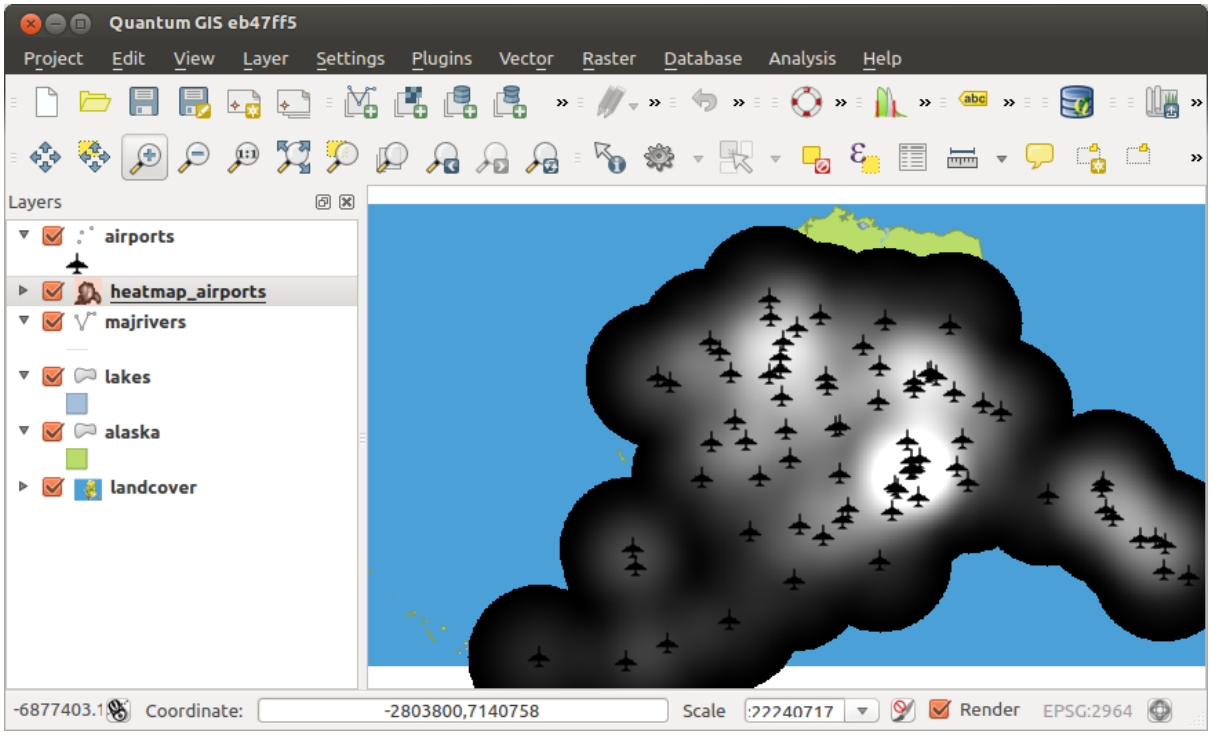


Figura 24.26: El mapa de calor después de la carga parece una superficie gris

1. Abra el diálogo de propiedades de la capa `heatmap_airports` (seleccione la capa `heatmap_airports`, abra el menú contextual con el botón derecho del ratón y seleccione *Propiedades*).
2. Seleccione la pestaña *Simbología*
3. Cambie el *tipo de Representación* a “pseudocolor Banda única”.
4. Seleccione una adecuada *rampa de color* , por ejemplo `YlOrRd`.
5. Click en el botón *Clasificar*.
6. Presiona *Aceptar* para actualizar la capa.

El resultado final es mostrado en Figura 24.27.

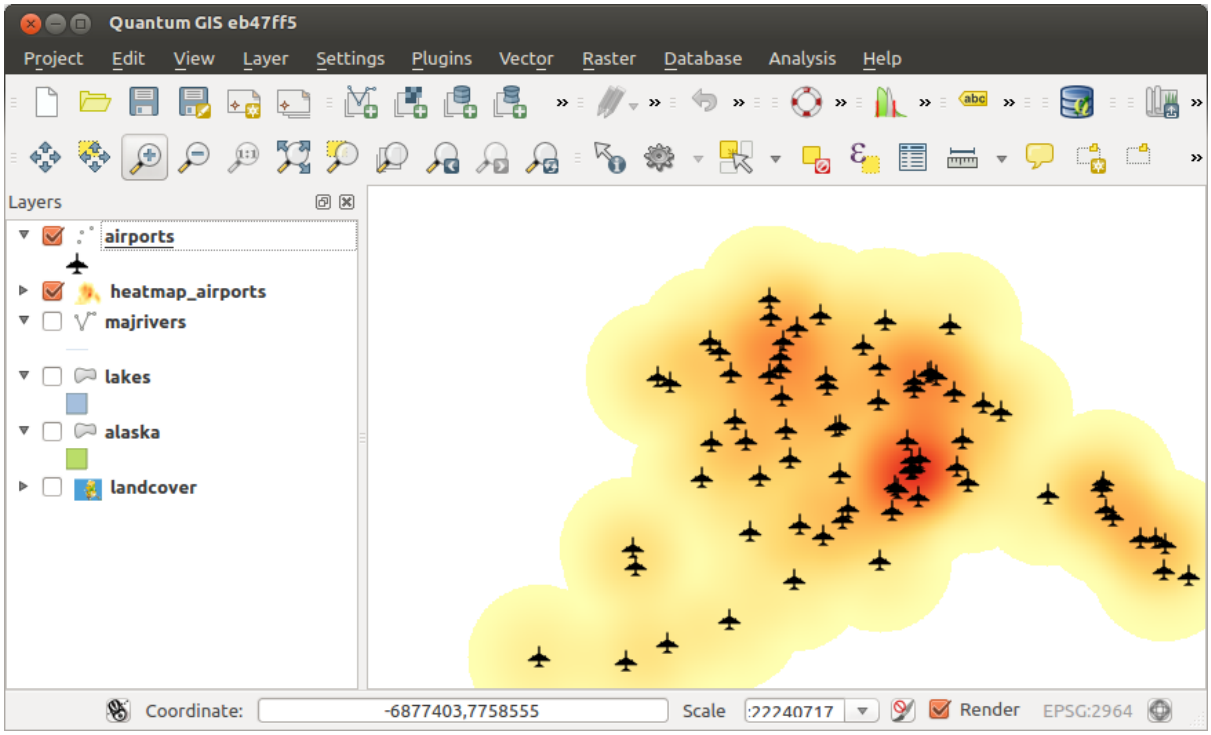


Figura 24.27: Mapa de calor con estilo de los aeropuertos de Alaska

Código Python

Algorithm ID: `qgis:heatmapkerneldensityestimation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Interpolación IDW

Genera una interpolación de distancia inversa ponderada (IDW) de una capa vectorial de puntos.

Los puntos de muestra se ponderan durante la interpolación de modo que la influencia de un punto en relación con otro disminuye con la distancia desde el punto desconocido que desea crear.

El método de interpolación IDW también tiene algunas desventajas: la calidad del resultado de la interpolación puede disminuir si la distribución de los puntos de datos de la muestra es desigual.

Además, los valores máximo y mínimo en la superficie interpolada solo pueden ocurrir en puntos de datos de muestra.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa(s) de entrada	`INTERPOLATION_`	[cadena de texto]	Capa(s) de vector y campo(s) a utilizar para la interpolación, codificados en una cadena (consulte la clase `ParameterInterpolationData` en InterpolationWidgets para mas detalles). Se proporcionan los siguientes elementos de la GUI para componer la cadena de datos de interpolación: • **capa Vectorial** [vector: any] • **atributo interpolación** [tablefield: numeric]: Atributo a emplear en la interpolación • **Usar coordenada Z para interpolación** [boolean]: Utiliza los valores Z almacenados de la capa (predeterminado: False) Para cada una de las combinaciones de campo de capa agregadas, se puede elegir un tipo: • *Points*: Elevation sample locations containing the values to be interpolated (e.g., terrain height, Z). These points are the primary input for calculating the output surface. • *Structured lines*: Lines representing known continuous surface features that guide the terrain shape or structure (e.g., ridges, road centerlines, levees). They describe terrain direction while assuming continuity along the line. • *Break lines*: Lines representing known surface discontinuities or hard boundaries (e.g., cliffs, river banks, excavation edges, lake/pond outlines). They mark where the surface may change sharply. It is required if there are significant changes in elevation along the line. En la cadena, los elementos del campo de capa están separados por `'::|::'`. Los subelementos de los elementos de campo de capa están separados por `'::~::'`.
Coeficiente de Distancia P	`DISTANCE_COEFF`	[numérico: doble] Preestablecido: 2.0	Establece el coeficiente de distancia para la interpolación. Mínimo: 0.0, máximo: 100.0.

continúe en la próxima página

Tabla 24.153 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extesión (xmin, xmax, ymin, ymax)	`EXTENT`	[extensión]	Extensión de la capa ráster de salida. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Tamaño del ráster de salida	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 0.1	Tamaño de píxel de la capa ráster de salida en unidades de capa. En la GUI, el tamaño se puede especificar por el número de filas (`Número de filas`)/ columnas(`Número de columnas`) **o** el tamaño de píxel (`Tamaño de píxel X` / `Tamaño de píxel Y`). El aumento del número de filas o columnas disminuirá el tamaño de la celda y aumentará el tamaño del archivo del ráster de salida. Los valores en `Filas`, `Columnas`, `Tamaño de píxel X` y `Tamaño de píxel Y` se actualizarán simultáneamente; al duplicar el número de filas, se duplicará el número de columnas, y la celda el tamaño se reducirá a la mitad. La extensión del ráster de salida seguirá siendo la misma (aproximadamente).
Interpolada	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Capa ráster de valores interpolados. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Interpolada	`OUTPUT`	[ráster]	Capa ráster de valores interpolados

Código Python

Algorithm ID: `qgis:idwinterpolation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Densidad lineal

Calcula para cada celda ráster, la medida de densidad de entidades lineales dentro de una vecindad circular. Esta medida se obtiene sumando todos los segmentos de línea que intersecan la vecindad circular y dividiendo esta suma por el área de dicha vecindad. Se puede aplicar un factor de ponderación a los segmentos de línea.

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

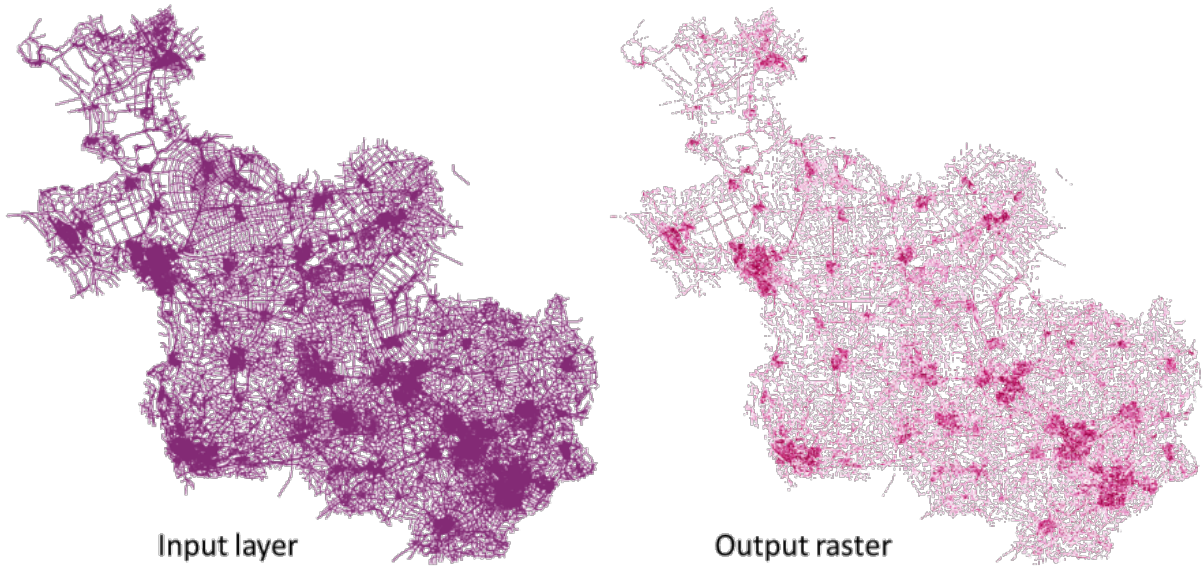


Figura 24.28: Ejemplo de densidad lineal. Fuente de capa entrante: Roads Overijssel - The Netherlands (OSM).

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de líneas de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante contenedora de entidades lineales

continúe en la próxima página

Tabla 24.155 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Campo de peso	`WEIGHT`	[campo de tabla: numérico]	Campo de la capa contenedora del factor de ponderación a usar durante el cálculo
Radio de Búsqueda	`RADIUS`	[numérico: doble] Predeterminado: 10.0	Radio del vecindario circular. Las unidades pueden ser especificadas aquí.
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 10.0	Tamaño de píxel de la capa ráster saliente en unidades de capa. El ráster tiene pixels cuadrados.
Ráster de densidad lineal	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	La salida como capa ráster. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de densidad lineal	`OUTPUT`	[ráster]	La capa ráster saliente de densidad lineal.

Código Python

Algoritmo ID: `native:linedensity`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Interpolación TIN

Genera una interpolación de red irregular triangulada (TIN) de una capa vectorial de puntos.

Con el método TIN puede crear una superficie formada por triángulos de puntos vecinos más cercanos. Para hacer esto, se crean circunferencias alrededor de los puntos de muestra seleccionados y sus intersecciones se conectan a una red de triángulos no superpuestos y tan compactos como sea posible. Las superficies resultantes no son lisas.

El algoritmo crea tanto la capa ráster de los valores interpolados como la capa de línea vectorial con los límites de triangulación.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa(s) de entrada	`INTERPOLATION_`	[cadena de texto]	Capa(s) de vector y campo(s) a utilizar para la interpolación, codificados en una cadena (consulte la clase `ParameterInterpolationData` en InterpolationWidgets para mas detalles). Se proporcionan los siguientes elementos de la GUI para componer la cadena de datos de interpolación: • **capa Vectorial** [vector: any] • **atributo interpolación** [tablefield: numeric]: Atributo a emplear en la interpolación • **Usar coordenada Z para interpolación** [boolean]: Utiliza los valores Z almacenados de la capa (predeterminado: False) Para cada una de las combinaciones de campo de capa agregadas, se puede elegir un tipo: • *Points*: Elevation sample locations containing the values to be interpolated (e.g., terrain height, Z). These points are the primary input for calculating the output surface. • *Structured lines*: Lines representing known continuous surface features that guide the terrain shape or structure (e.g., ridges, road centerlines, levees). They describe terrain direction while assuming continuity along the line. • *Break lines*: Lines representing known surface discontinuities or hard boundaries (e.g., cliffs, river banks, excavation edges, lake/pond outlines). They mark where the surface may change sharply. It is required if there are significant changes in elevation along the line. En la cadena, los elementos del campo de capa están separados por `'::|::'`. Los subelementos de los elementos de campo de capa están separados por `'::~::'`.
Método interpolación	`METHOD`	[enumeración] Predeterminado: 0	Configura el método de interpolación a ser empleada. Una de: • *Lineal* • *Clough-Toucher (cúbico)*

continúe en la próxima página

Tabla 24.158 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extesión (xmin, xmax, ymin, ymax)	`EXTENT`	[extensión]	Extensión de la capa ráster de salida. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Tamaño del ráster de salida	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 0.1	Tamaño de píxel de la capa ráster de salida en unidades de capa. En la GUI, el tamaño se puede especificar por el número de filas (`Número de filas`)/ columnas(`Número de columnas`) **o** el tamaño de píxel (`Tamaño de píxel X` / `Tamaño de píxel Y`). El aumento del número de filas o columnas disminuirá el tamaño de la celda y aumentará el tamaño del archivo del ráster de salida. Los valores en `Filas`, `Columnas`, `Tamaño de píxel X` y `Tamaño de píxel Y` se actualizarán simultáneamente; al duplicar el número de filas, se duplicará el número de columnas, y la celda el tamaño se reducirá a la mitad. La extensión del ráster de salida seguirá siendo la misma (aproximadamente).
Interpolada	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	La interpolación TIN resultante como capa ráster. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…
Triangulación	`TRIANGULATION`	[vectorial: lineal] Preestablecido: `[Omitir salida]`	El resultado TIN como capa vectorial. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Interpolada	`OUTPUT`	[ráster]	La salida de la interpolación TIN como una capa ráster
Triangulación	`TRIANGULATION`	[vectorial: lineal]	La TIN saliente como capa vectorial.

Código Python

Algorithm ID: `qgis:tininterpolation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.9 Herramientas de capa

Exportar información de capa(s)

Crea una capa de polígono con características correspondientes a la extensión de la(s) capa(s) seleccionada(s)

Los detalles adicionales de la capa (CRS, nombre del proveedor, ruta del archivo, nombre de la capa, filtro de subconjunto, resumen y atribución) se adjuntan como atributos a cada entidad.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`LAYERS`	[layer] [list]	Capas de entrada para obtener información.
Salida	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especificación de la capa de salida con información. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Salida	`OUTPUT`	[vectorial: poligonal]	Capa vectorial de polígono que muestra la extensión de las capas de entrada y la información asociada en los atributos.

Código Python

Algoritmo ID: `native:exportmeshvertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Exportar a hoja de cálculo

Exporta los atributos de una selección de capas vectoriales a un documento de hoja de cálculo o, opcionalmente, los agrega a una hoja de cálculo existente como hojas adicionales.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`LAYERS`	[vectorial: cualquiera] [lista]	Capas vectoriales de entrada. La hoja de cálculo de salida constará de una hoja, para cada capa, que contiene los atributos de esta capa.
Utilice alias de campo como encabezados de columna	`USE_ALIAS`	[booleano] Predeterminado: False	Utilice los alias de campo de la tabla de atributos para la hoja de cálculo.
Exportar valores formateados en lugar de valores sin formato	`FORMATTED_VALU`	[booleano] Predeterminado: False	Si `Verdadero`, exporta los valores formateados legibles por humanos (p. ej., de un :ref:`value map or value relation <edit_widgets>) a la hoja de cálculo.
Sobrescribir hoja de cálculo existente	`OVERWRITE`	[booleano] Predeterminado: Verdadero	Si la hoja de cálculo especificada existe, establecer esta opción en `Verdadero` sobrescribirá la hoja de cálculo existente. Si esta opción es `Falso` y la hoja de cálculo existe, las capas se agregarán como hojas adicionales.
Hoja de cálculo de destino	`OUTPUT`	[archivo] Predeterminado: `[Save to temporary file]`	Hoja de cálculo de salida con una hoja para cada capa. *Una de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Hoja de cálculo de destino	`OUTPUT`	[archivo]	Hoja de cálculo con una hoja para cada capa.
Capas dentro de la hoja de cálculo	`OUTPUT_LAYERS`	[lista]	La lista de hojas agregadas a la hoja de cálculo.

Código Python

Algoritmo ID: `native:exportmeshedges`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Extraer extensión de la capa

Genera una capa vectorial con el mínimo marco límite (rectángulo con orientación N-S) que cubre todas las entidades entrantes.

La capa saliente contiene un marco límite para toda la capa entrante.

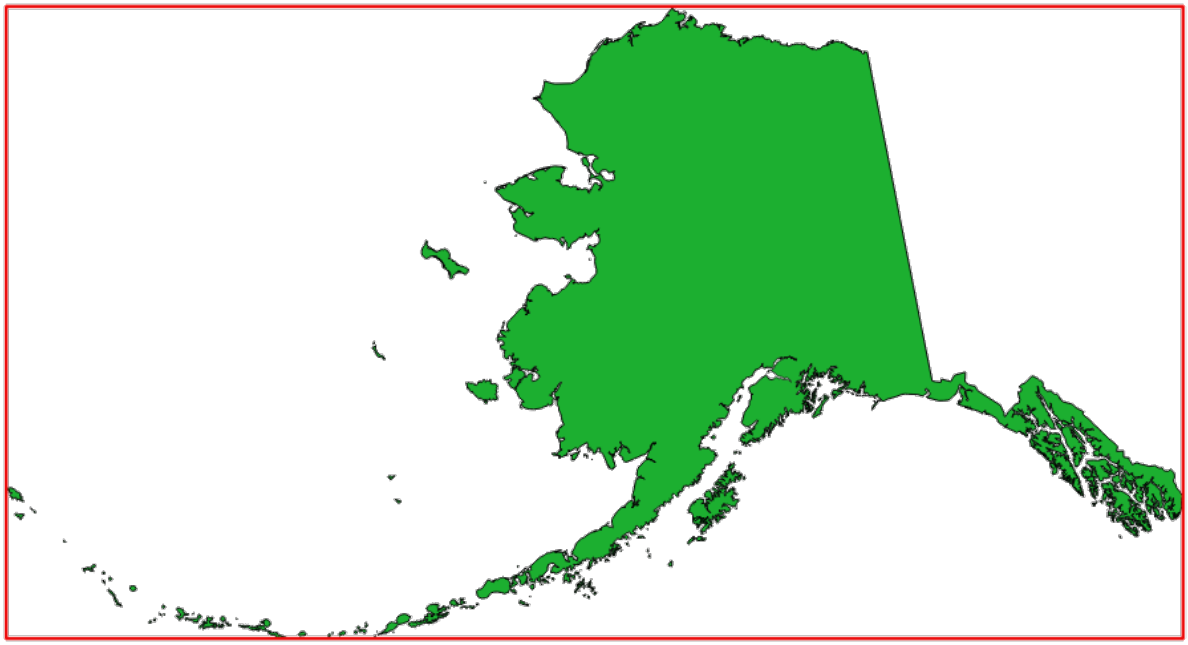


Figura 24.29: En rojo el marco límite de la capa fuente

menú predeterminado: *Vectorial* ► *Herramientas de Investigación*

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Layer	`INPUT`	[layer]	Capa entrante
Extensión	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial poligonal para la extensión de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extensión	`OUTPUT`	[vectorial: poligonal]	Capa vectorial saliente (poligonal) con la extensión (mínimo marco límite)

Código Python

Algorithm ID: `native:polygonfromlayerextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.10 Malla

Exportar contornos

Crea contornos como una capa vectorial a partir de un conjunto de datos escalares de malla.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de malla de entrada	`INPUT`	[malla]	La capa de malla para exportar datos desde
Grupos de conjuntos de datos	`DATASET_GROUPS`	[layer] [list]	Los grupos de conjuntos de datos

continúe en la próxima página

Tabla 24.166 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tiempo del conjunto de datos	`DATASET_TIME`	[datetime]	El intervalo de tiempo a tener en cuenta • 0 — Tiempo de lienzo actual • 1 — Fecha/hora definida • 2 — Paso de tiempo del grupo de conjuntos de datos
Incremento entre niveles de curvas Opcional	`INCREMENT`	[numérico: doble] Predeterminado: No establecido	Intervalo entre los niveles generados.
Nivel de contorno mínimo Opcional	`MINIMUM`	[numérico: doble] Predeterminado: No establecido	Valores del nivel inicial de los contornos. Ningún nivel generado será inferior a este valor.
Nivel de contorno máximo Opcional	`MAXIMUM`	[numérico: doble] Predeterminado: No establecido	Valores máximos de contornos, es decir, ningún nivel generado será mayor que este valor.
Lista de niveles de contornos Opcional	`CONTOUR_LEVEL_`	[cadena de texto] Predeterminado: No establecido	Lista de niveles de contorno deseados (separados por comas). Si se rellena, no se tendrán en cuenta los campos de incremento, mínimo y máximo.
Sistema de coordenadas de salida Opcional	`CRS_OUTPUT`	[src]	Sistema de referencia de coordenadas para asignar a la salida
Líneas de contorno exportadas	`OUTPUT_LINES`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de línea de salida que representa los contornos de la capa de malla. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Polígonos de contorno exportados	`OUTPUT_POLYGON`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de polígonos de salida que representa los contornos de la capa de malla. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Líneas de contorno exportadas	`OUTPUT_LINES`	[vectorial: lineal]	Capa de línea que representa los contornos de la capa de malla.
Polígonos de contorno exportados	`OUTPUT_POLYGON`	[vectorial: poligonal]	Capa de polígono que representa los contornos de la capa de malla.

Código Python

Algoritmo ID: `native:meshcontours`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar valores de conjuntos de datos de sección transversal en líneas de malla

Extrae los valores de un conjunto de datos de malla de las líneas contenidas en una capa vectorial.

Cada línea se discretiza con un parámetro de distancia de resolución para la extracción de valores en sus vértices.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de malla de entrada	`INPUT`	[malla]	La capa de malla para exportar datos desde
Grupos de conjuntos de datos	`DATASET_GROUPS`	[layer] [list]	Los grupos de conjuntos de datos
Tiempo del conjunto de datos	`DATASET_TIME`	[datetime]	El intervalo de tiempo a tener en cuenta • 0 — Tiempo de lienzo actual • 1 — Fecha/hora definida • 2 — Paso de tiempo del grupo de conjuntos de datos
Líneas para exportación de datos	`INPUT_LINES`	[vectorial: lineal]	Líneas donde se extraerán los datos de la malla del conjunto de datos
Resolución de segmentación de línea	`RESOLUTION`	[numérico: doble] Predeterminado: 10.0	La distancia entre los puntos de las líneas donde se extraerán los datos de la malla del conjunto de datos. Se pueden seleccionar unidades.

continúe en la próxima página

Tabla 24.168 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Los dígitos cuentan para las coordenadas	`COORDINATES_DI`	[numérico: entero] Preestablecido: 2	Número de dígitos para redondear los valores de las coordenadas
Los dígitos cuentan para el valor del conjunto de datos	`DATASET_DIGITS`	[numérico: entero] Preestablecido: 2	Número de dígitos para redondear los valores del conjunto de datos
Archivo CSV de datos exportados	`OUTPUT`	[archivo] Predeterminado: `[Save to temporary file]`	Especificación del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo CSV de datos exportados	`OUTPUT`	[archivo]	

Código Python

Algoritmo ID: `native:meshexportcrosssection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar bordes de malla

Exporta los bordes de una capa de malla a una capa de vector de línea, con los valores del conjunto de datos en los bordes como valores de atributo.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de malla de entrada	`INPUT`	[malla]	La capa de malla para exportar datos desde
Grupos de conjuntos de datos	`DATASET_GROUPS`	[layer] [list]	Los grupos de conjuntos de datos

continúe en la próxima página

Tabla 24.170 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tiempo del conjunto de datos	`DATASET_TIME`	[datetime]	El intervalo de tiempo a tener en cuenta • 0 — Tiempo de lienzo actual • 1 — Fecha/hora definida • 2 — Paso de tiempo del grupo de conjuntos de datos
Sistema de coordenadas de salida Opcional	`CRS_OUTPUT`	[src]	Sistema de referencia de coordenadas para asignar a la salida
Opción de exportar vector	`VECTOR_OPTION`	[enumeración]	Tipo de coordenadas de exportación de valores vectoriales. • 0 — Cartesianas (x,y) • 1 — Polar (magnitud, grado) • 2 — Cartesianas y polares
Capa vectorial de salida	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especificación del archivo de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial de salida	`OUTPUT`	[vectorial: lineal]	Capa de línea de vector de salida que contiene los bordes de la capa de malla de entrada con valores de conjunto de datos asociados

Código Python

Algoritmo ID: `native:exportmeshedges`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar caras de malla

Exporta las caras de una capa de malla a una capa vectorial de polígono, con los valores del dataset en las caras como valores de atributo.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de malla de entrada	`INPUT`	[malla]	La capa de malla para exportar datos desde
Grupos de conjuntos de datos	`DATASET_GROUPS`	[layer] [list]	Los grupos de conjuntos de datos
Tiempo del conjunto de datos	`DATASET_TIME`	[datetime]	El intervalo de tiempo a tener en cuenta • 0 — Tiempo de lienzo actual • 1 — Fecha/hora definida • 2 — Paso de tiempo del grupo de conjuntos de datos
Sistema de coordenadas de salida Opcional	`CRS_OUTPUT`	[src]	Sistema de referencia de coordenadas para asignar a la salida
Opción de exportar vector	`VECTOR_OPTION`	[enumeración]	Tipo de coordenadas de exportación de valores vectoriales. • 0 — Cartesianas (x,y) • 1 — Polar (magnitud, grado) • 2 — Cartesianas y polares
Capa vectorial de salida	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especificación del archivo de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial de salida	`OUTPUT`	[vectorial: poligonal]	Capa de polígono vectorial de salida que contiene las caras de la capa de malla de entrada con valores de conjunto de datos asociados

Código Python

Algoritmo ID: `native:exportmeshfaces`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Formato de salida de la capa generada

Exporta los valores del conjunto de datos de una capa de malla a una capa vectorial de puntos cuadriculada, con los valores del conjunto de datos en este punto como valores de atributo.

Para datos sobre volumen (valores de conjuntos de datos apilados en 3D), los valores de conjuntos de datos exportados se promedian en caras utilizando el método definido en:ref:*the mesh layer properties <mesh_stacked_averaging>* (el valor predeterminado es el método de promedio de varios niveles). Las mallas 1D no son compatibles.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de malla de entrada	`INPUT`	[malla]	La capa de malla para exportar datos desde
Grupos de conjuntos de datos	`DATASET_GROUPS`	[layer] [list]	Los grupos de conjuntos de datos
Tiempo del conjunto de datos	`DATASET_TIME`	[datetime]	El intervalo de tiempo a tener en cuenta • 0 — Tiempo de lienzo actual • 1 — Fecha/hora definida • 2 — Paso de tiempo del grupo de conjuntos de datos
Extensión Opcional	`EXTENT`	[extensión]	Especifique la extensión espacial en la que procesar los datos. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`

continúe en la próxima página

Tabla 24.174 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Espaciado de rejilla Opcional	`GRID_SPACING`	[numérico: doble] Predeterminado: 10.0	Espaciado entre los puntos de muestra a utilizar
Sistema de coordenadas de salida Opcional	`CRS_OUTPUT`	[src]	Sistema de referencia de coordenadas para asignar a la salida
Opción de exportar vector	`VECTOR_OPTION`	[enumeración]	Tipo de coordenadas de exportación de valores vectoriales. • 0 — Cartesianas (x,y) • 1 — Polar (magnitud, grado) • 2 — Cartesianas y polares
Capa vectorial de salida	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especificación del archivo de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial de salida	`OUTPUT`	[vector: punto]	Capa de puntos vectoriales de salida con valores de conjuntos de datos calculados a partir de la cara superpuesta.

Código Python

Algoritmo ID: `native:exportmeshongrid`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar vértices de malla

Exporta los vértices de una capa de malla a una capa de vector de puntos, con los valores del dataset en los vértices como valores de atributo.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de malla de entrada	`INPUT`	[malla]	La capa de malla para exportar datos desde
Grupos de conjuntos de datos	`DATASET_GROUPS`	[layer] [list]	Los grupos de conjuntos de datos
Tiempo del conjunto de datos	`DATASET_TIME`	[datetime]	El intervalo de tiempo a tener en cuenta • 0 — Tiempo de lienzo actual • 1 — Fecha/hora definida • 2 — Paso de tiempo del grupo de conjuntos de datos
Sistema de coordenadas de salida Opcional	`CRS_OUTPUT`	[src]	Sistema de referencia de coordenadas para asignar a la salida
Opción de exportar vector	`VECTOR_OPTION`	[enumeración]	Tipo de coordenadas de exportación de valores vectoriales. • 0 — Cartesianas (x,y) • 1 — Polar (magnitud, grado) • 2 — Cartesianas y polares
Capa vectorial de salida	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especificación del archivo de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial de salida	`OUTPUT`	[vector: punto]	Output vector point layer containing the vertices of the input mesh layer with associated dataset values

Código Python

Algoritmo ID: `native:exportmeshvertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Exportar valores de series temporales desde puntos de un conjunto de datos de malla

Extrae los valores de serie temporal de un conjunto de datos de malla de puntos contenidos en una capa vectorial.

Si el paso de tiempo se mantiene en su valor predeterminado (0 horas), el paso de tiempo utilizado es uno de los dos primeros conjuntos de datos del primer grupo de conjuntos de datos seleccionado.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de malla de entrada	`INPUT`	[malla]	La capa de malla para extraer datos de
Grupos de conjuntos de datos	`DATASET_GROUPS`	[layer] [list]	Los grupos de conjuntos de datos
Tiempo de empezar	`STARTING_TIME`	[datetime]	El inicio del intervalo de tiempo a tener en cuenta • 0 — Tiempo de lienzo actual • 1 — Fecha/hora definida • 2 — Paso de tiempo del grupo de conjuntos de datos
Tiempo de finalización	`FINISHING_TIME`	[datetime]	El final del intervalo de tiempo a tener en cuenta • 0 — Tiempo de lienzo actual • 1 — Fecha/hora definida • 2 — Paso de tiempo del grupo de conjuntos de datos
Paso de tiempo (horas) Opcional	`TIME_STEP`	[numérico: doble] Predeterminado: 0	Tiempo entre dos pasos consecutivos para extraer. Mantenga `0` para usar el paso de tiempo del primer grupo de conjuntos de datos seleccionado.
Puntos por exportación de datos	`INPUT_POINTS`	[vector: punto]	Capa vectorial que contiene puntos donde se extraerán los datos de la malla del conjunto de datos
Los dígitos cuentan para las coordenadas	`COORDINATES_DI`	[numérico: entero]	Número de dígitos para redondear los valores de las coordenadas Preestablecido: 2
Los dígitos cuentan para el valor del conjunto de datos	`DATASET_DIGITS`	[numérico: entero] Preestablecido: 2	Número de dígitos para redondear los valores del conjunto de datos

continúe en la próxima página

Tabla 24.178 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Archivo CSV de datos exportados	`OUTPUT`	[archivo] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo CSV de datos exportados	`OUTPUT`	[archivo]	`.CSV` file contiene los valores de la serie temporal del conjunto de datos de malla en las entidades de puntos superpuestas

Código Python

Algoritmo ID: `native:meshexporttimeseries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Rasterizar conjunto de datos de malla

Crea una capa ráster a partir de un dataset de malla.

Para datos sobre volumen (valores de conjuntos de datos apilados en 3D), los valores de conjuntos de datos exportados se promedian en caras utilizando el método definido en:ref:*the mesh layer properties <mesh_stacked_averaging>* (el valor predeterminado es el método de promedio de varios niveles). Las mallas 1D no son compatibles.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de malla de entrada	`INPUT`	[malla]	La capa de malla para exportar datos desde
Grupos de conjuntos de datos	`DATASET_GROUPS`	[layer] [list]	Los grupos de conjuntos de datos

continúe en la próxima página

Tabla 24.180 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tiempo del conjunto de datos	`DATASET_TIME`	[datetime]	El intervalo de tiempo a tener en cuenta • 0 — Tiempo de lienzo actual • 1 — Fecha/hora definida • 2 — Paso de tiempo del grupo de conjuntos de datos
Extensión Opcional	`EXTENT`	[extensión]	Especifique la extensión espacial en la que procesar los datos. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 1.0	Tamaño de píxel de la capa ráster de salida.
Sistema de coordenadas de salida Opcional	`CRS_OUTPUT`	[src]	Sistema de referencia de coordenadas para asignar a la salida
Capa ráster de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa ráster de salida	`OUTPUT`	[ráster]	Capa ráster de salida con valores de dataset calculados a partir de la capa de malla.

Código Python

Algoritmo ID: `native:meshrasterize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Surface to Polygon

Exports a polygon file containing the boundary of a mesh layer. The resulting polygon may contain holes and may be a multi-part polygon.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de malla de entrada	`INPUT`	[malla]	The input mesh layer whose boundary will be exported as a polygon
Sistema de coordenadas de salida Opcional	`OUTPUT_CRS`	[src]	The coordinate reference system (CRS) for the output polygon layer. If not specified, the CRS of the input mesh layer will be used.
Capa vectorial de salida	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especificación del archivo de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial de salida	`OUTPUT`	[vectorial: poligonal]	The resulting polygon layer containing the boundary of the mesh layer. The polygon may contain holes and may be multi-part.

Código Python

Algorithm ID: `native:surfacetopolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Creación de mallas TIN

Crea una capa de malla TIN a partir de capas vectoriales. La malla TIN se crea utilizando una triangulación de Delaunay.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`SOURCE_DATA`	[vectorial: cualquiera] [lista]	Capas vectoriales a combinar para generar la capa de malla
Capa vectorial	SOLO GUI	[vectorial: cualquiera] [lista]	Capas vectoriales a combinar para generar la capa de malla
Valor en el vértice	SOLO GUI	[campo de tabla: cualquiera]	Un selector del campo a utilizar de la capa seleccionada. A cada vértice se le asigna el valor correspondiente de su característica original.
Utilice la coordenada Z para el valor en el vértice	SOLO GUI	[booleano] Predeterminado: Falso	Si está marcado, el valor Z de los puntos de la capa vectorial o los vértices de polígonos/líneas se utilizará para asignar el valor Z de la capa de malla de vértice. Solo disponible si las capas de entrada son 3D.
Formato de salida	`MESH_FORMAT`	[enumeración] Predeterminado: 2DM	Formato de salida de la capa generada • 0 — 2DM • 1 — SELAFÍN • 2 — CAPAS • 3 — Ugrid • 4 — Mike21
Sistema de coordenadas de salida Opcional	`CRS_OUTPUT`	[src]	Sistema de referencia de coordenadas para asignar a la salida

continúe en la próxima página

Tabla 24.185 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Archivo de salida	`OUTPUT_MESH`	[malla] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo de salida	`OUTPUT_MESH`	[malla]	Capa de malla de salida con valores de conjuntos de datos calculados a partir de las capas vectoriales.

Código Python

Algoritmo ID: `native:tinmeshcreation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.11 Metadata tools

Add history metadata

Adds a new history entry to the layer's metadata.

Parameters

Label	Name	Type	Description
Layer	`INPUT`	[layer]	The input layer (vector, raster, etc.) to which the history entry will be added.
History entry	`HISTORY`	[string]	The text to be appended as a new entry in the layer's history metadata. This will be added to any existing history entries.

Outputs

Label	Name	Type	Description
Output	`OUTPUT`	[same as input]	The resulting layer with the updated history in its **Metadata properties**.

Python code

Algorithm ID: `native:addhistorymetadata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Copy Layer Metadata

Copies metadata from a source layer to a target layer. Any existing metadata in the target layer will be replaced with the metadata from the source layer. This includes all metadata fields, such as history, abstract, and other properties.

Parameters

Label	Name	Type	Description
Source layer	`SOURCE`	[layer]	The layer from which metadata will be copied.
Target layer	`TARGET`	[layer]	The layer to which metadata will be pasted. Any existing metadata in this layer will be replaced.
Save metadata as default	`DEFAULT`	[boolean] Default: False	If checked, the metadata information will be saved with the layer, hence available by default in subsequent projects.

Outputs

Label	Name	Type	Description
Output layer	`OUTPUT`	[layer]	The target layer with the metadata replaced by the metadata from the source layer. This includes all metadata fields, such as history, abstract, and other properties.

Python code

Algorithm ID: `native:copylayermetadata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Export Layer Metadata

Exports the metadata of a layer to a QMD file.

Parameters

Label	Name	Type	Description
Layer	`INPUT`	[layer]	The layer whose metadata will be exported.
Output	`OUTPUT`	[file] Default: `[Save to temporary file]`	Specification of the output file. *One of*: • Save to a Temporary File • Save to File…

Outputs

Label	Name	Type	Description
Output file	`OUTPUT`	[file]	The `.qmd` file containing the exported metadata. This file can be used to import metadata into another layer.

Python code

Algorithm ID: `native:exportlayermetadata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Set Layer Metadata

Applies metadata to a layer from a `.qmd` file.

Parameters

Label	Name	Type	Description
Layer	`INPUT`	[layer]	The layer to which the metadata will be applied.
Metadata file	`METADATA`	[file]	The `.qmd` file containing the metadata to be applied.
Save metadata as default	`DEFAULT`	[boolean] Default: False	If checked, the metadata information will be saved with the layer, hence available by default in subsequent projects.

Outputs

Label	Name	Type	Description
Output layer	`OUTPUT`	[same as input]	The input layer with the metadata replaced by the metadata from the `.qmd` file.

Python code

Algorithm ID: `native:setlayermetadata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Set Metadata Fields

Sets various metadata fields for a layer.

Parameters

Label	Name	Type	Description
Layer	`INPUT`	[layer]	The layer whose metadata fields will be updated
Identifier Optional	`IDENTIFIER`	[string]	A unique identifier for the layer
Parent identifier Optional	`PARENT_IDENTIF`	[string]	The identifier of the parent resource, if applicable
Title Optional	`TITLE`	[string]	The title of the layer

continúe en la próxima página

Tabla 24.195 – proviene de la página anterior

Label	Name	Type	Description
Type Optional	TYPE	[string]	The type of data stored in the layer
Language Optional	LANGUAGE	[string]	The language of the metadata
Encoding Optional	ENCODING	[string]	The character encoding used in the metadata
Abstract Optional	ABSTRACT	[string]	A brief description or abstract of the layer
Coordinate reference system Optional	CRS	[crs]	The coordinate reference system of the layer
Fees Optional	FEES	[string]	Information about any fees associated with accessing the layer
Ignore empty fields	IGNORE_EMPTY	[boolean] Default: False	If checked, no update will be done to metadata fields that are not filled

Outputs

Label	Name	Type	Description
Output layer	OUTPUT	[same as input]	The input layer with the specified metadata fields updated.

Python code

Algorithm ID: `native:setmetadatafields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

Update Layer Metadata

Copies all non-empty metadata fields from a source layer to a target layer. Leaves empty input fields unchanged in the target.

Parameters

Label	Name	Type	Description
Source layer	SOURCE	[layer]	The layer from which non-empty metadata fields will be copied
Target layer	TARGET	[layer]	The layer to which non-empty metadata fields will be pasted. Empty fields in the source layer will not overwrite existing metadata in the target layer.

Outputs

Label	Name	Type	Description
Target layer	`OUTPUT`	[layer]	The target layer with updated metadata.

Python code

Algorithm ID: `native:updatelayermetadata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

The *algorithm id* is displayed when you hover over the algorithm in the Processing Toolbox. The *parameter dictionary* provides the parameter NAMEs and values. See *Utilizar algoritmos de procesamiento desde la consola* for details on how to run processing algorithms from the Python console.

24.1.12 Herramientas del Modelador

Advertencia

Estas herramientas sólo están disponibles en el diseñador de modelos. No están disponibles en la Caja de herramientas de Procesos.

Calcular expresión

Calcula el resultado de una expresión QGIS y elimina la necesidad de utilizar la misma expresión varias veces a lo largo de un modelo cuando se requiere el mismo resultado más de una vez. Además, permite casos de uso que de otro modo no serían posibles. Por ejemplo, se puede generar un valor de marca de tiempo una vez y utilizarlo múltiples veces dentro del modelo, si la marca de tiempo se recalculara cada vez, los valores variarían durante el tiempo de ejecución del modelo.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Entrada	`INPUT`	[expresión]	Expresión para calcular

Salidas

Etiqueta	Nombre	Tipo	Descripción
Valor	`OUTPUT`	[Valor del resultado]	Valor de resultado calculado, el tipo de datos de la salida variará en función de la expresión específica utilizada en el algoritmo.

Código Python

ID del algoritmo: `native:calculateexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Rama condicional

Agrega una rama condicional a un modelo, lo que permite que se ejecuten partes del modelo en función del resultado de una evaluación de expresión. Principalmente mediante el uso de dependencias de herramientas para controlar el flujo de un modelo.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Campo	`RAMA`	[cadena de texto]	Nombre de la condición
Campo	`CONDICIÓN`	[expresión]	Expresión para evaluar

Salidas

Ninguno.

Código Python

Algoritmo ID: `native:condition`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Crear el directorio

Crea un nuevo directorio en un sistema de archivos. Los directorios se crearán de forma recursiva, creando todos los directorios principales necesarios para construir la ruta completa del directorio especificado. No se generarán errores si el directorio ya existe.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Ruta de directorio	`PATH`	[cadena de texto]	Ruta de carpeta para crear

Salidas

Etiqueta	Nombre	Tipo	Descripción
Salida	`OUTPUT`	[carpeta]	Carpeta creada

Código Python

Algoritmo ID: `native:createdirectory`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Filtro de objetos espaciales

Filtra características de la capa de entrada y las redirige a una o varias salidas. Si no conoce ningún nombre de atributo que sea común a todas las capas de entrada posibles, el filtrado solo es posible en la geometría de características y en los mecanismos generales de registro, como `$id` y `uuid`.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	La capa entrante.
Salidas y filtros (uno o más)	`OUTPUT_<name of the filter>`	[la misma que la entrada]	Las capas de salida con filtros (tantas como filtros haya).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Salida (uno o más)	`native:filter_ of filter>`	[la misma que la entrada]	Las capas de salida con entidades filtradas (tantas como filtros).

Código Python

Algoritmo ID: `native:filter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Filtrar capas por tipo

Filtra capas por su tipo. Las capas entrantes se dirigirán a diferentes salidas en función de si son una capa vectorial o ráster.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[layer]	Capa de mapa genérico

Salidas

Etiqueta	Nombre	Tipo	Descripción
Objetos espaciales vectoriales Opcional	`VECTOR`	[vector]	Una capa vectorial de la entrada, si es compatible
Capa ráster Opcional	`RASTER`	[ráster]	Una capa ráster de la entrada, si es compatible

Código Python

Algoritmo ID: `native:filterlayersbytype`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Cargar capa en proyecto

Carga una capa al proyecto actual.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Layer	`INPUT`	[layer]	Capa a cargar en la leyenda
Nombre de la capa cargada	`NAME`	[cadena de texto]	Nombre de la capa cargada

Salidas

Etiqueta	Nombre	Tipo	Descripción
Layer	`OUTPUT`	[la misma que la entrada]	La capa cargada (renombrada)

Código Python

Algoritmo ID: `native:loadlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Levantar excepción

Genera una excepción y cancela la ejecución de un modelo. El mensaje de excepción se puede personalizar y, opcionalmente, se puede especificar una condición basada en una expresión. Si se usa una condición de expresión, la excepción solo se generará si el resultado de la expresión es verdadero. Un resultado falso indica que no se generará ninguna excepción y que la ejecución del modelo puede continuar sin interrupciones.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Mensaje	`MENSAJE`	[cadena de texto]	Mensaje a mostrar
Condición Opcional	`CONDICIÓN`	[expresión]	Expresión para evaluar si es verdadera

Salidas

Un mensaje en el panel de registro.

Código Python

Algoritmo ID: `native:raiseexception`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Elevar mensaje

Genera un mensaje de información en el registro. El mensaje puede personalizarse y, opcionalmente, puede especificarse una condición basada en una expresión. Si se utiliza una condición de expresión, el mensaje sólo se registrará si el resultado de la expresión es verdadero. Un resultado falso indica que no se registrará ningún mensaje.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Mensaje informativo	`MENSAJE`	[cadena de texto]	Mensaje a mostrar
Condición Opcional	`CONDICIÓN`	[expresión]	Expresión para evaluar si es verdadera

Salidas

Un mensaje en el panel de registro.

Código Python

Identificación del algoritmo: `native:raisemessage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Levantar advertencia

Este algoritmo levanta un mensaje de advertencia en el registro. El mensaje de advertencia se puede personalizar y, opcionalmente se puede especificar una condición basada en una expresión. Si se utiliza una condición de expresión, la advertencia solo se registrará si el resultado de la expresión es verdadero. Un resultado falso indica que no se registrará ninguna advertencia.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Mensaje	`MENSAJE`	[cadena de texto]	Mensaje a mostrar
Condición Opcional	`CONDICIÓN`	[expresión]	Expresión para evaluar si es verdadera

Salidas

Un mensaje en el panel de registro.

Código Python

Algoritmo ID: `native:raisewarning`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Renombrar capa

Renombra una capa.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Layer	`INPUT`	[layer]	Capa a renombrar
Nuevo nombre	`NAME`	[cadena de texto]	El nuevo nombre de la capa

Salidas

Etiqueta	Nombre	Tipo	Descripción
Layer	`OUTPUT`	[la misma que la entrada]	La capa saliente (renombrada)

Código Python

Algoritmo ID: `native:renamelayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Guardar registro en archivo

Guarda el registro de ejecución del modelo en un archivo. Opcionalmente, el registro se puede guardar en una versión con formato HTML.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Usar HTML	`USAR_HTML`	[Booleano] Predeterminado: False	Usar formato HTML

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo	`OUTPUT`	[cadena de texto]	Destino del registro

Código Python

AlgoritmoID: `native:savelog`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Establecer variable de proyecto

Establece una expresión variable para el proyecto actual

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Nombre de la variable	`NAME`	[cadena de texto]	Nombre de la variable
Valor de la variable	`VALUE`	[cadena de texto]	Valor a almacenar

Salidas

Ninguno.

Código Python

Algoritmo ID: `native:setprojectvariable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Concatenación de cadena

Concatena dos cadenas en una simple en el Modelador de Procesos.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Entrada 1	`INPUT_1`	[cadena de texto]	Primera cadena
Entrada 2	`INPUT_2`	[cadena de texto]	Segunda cadena

Salidas

Etiqueta	Nombre	Tipo	Descripción
Concatenación	`CONCATENATION`	[cadena de texto]	La cadena concatenada

Código Python

Algoritmo ID: `native:stringconcatenation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Búfer de distancia variable

Advertencia

Este algoritmo está en desuso y se puede eliminar en cualquier momento. Prefiere usar el algoritmo *Buffer* en su lugar.

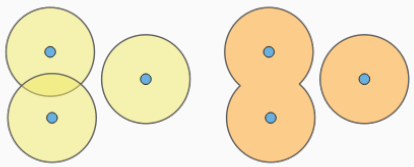
Calcula un área de influencia para todas las entidades en una capa de entrada.

El tamaño de la zona de influencia para una característica determinada se define mediante un atributo, por lo que permite que diferentes funciones tengan diferentes tamaños de zona de influencia.

Ver también

Buffer

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada
Campo de distancia	`DISTANCE`	[campo de tabla: numérico]	Atributo para el radio de distancia del búfer
Segmentos	`SEGMENTS`	[numérico: entero] Predeterminado: *5*	Controla el número de segmentos de linea a usar para aproximadamente un cuarto de circulo cuando se crean compensaciones redondeadas.
Resultado de disolución	`DISSOLVE`	[booleano] Predeterminado: *Falso*	Elija disolver el búfer final, lo que da como resultado una sola entidad que cubre todas las entidades de entrada.  Figura 24.30: Buffer normal y disuelto

continúe en la próxima página

Tabla 24.220 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Estilo tapa final	`END_CAP_STYLE`	[enumeración] Predeterminado: Redondo	Controla cómo se manejan los finales de línea en el búfer.  Figura 24.31: Estilos de tapa redondo, plano y cuadrado
Unir estilo	`JOIN_STYLE`	[enumeración] Predeterminado: Redondo	Especifica si se deben utilizar uniones redondas, a inglete o biseladas al desfasar esquinas en una línea.
Límite de inglete	`MITER_LIMIT`	[numérico: doble] Preestablecido: 2.0	Solo es aplicable para estilos de unión en inglete, y controla la distancia máxima desde la curva de desplazamiento que se utilizará al crear una unión en inglete.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Buffer	`OUTPUT`	[vectorial: poligonal]	Capa vectorial de polígono de búfer.

Código Python

Algoritmo ID: `qgis:variabledistancebuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.13 Análisis de Red

Área de servicio (a partir de capa)

Devuelve todos los bordes o partes de bordes de una red a los que se puede llegar dentro de una distancia o un tiempo determinados, partiendo de una capa de puntos. Esto permite evaluar la accesibilidad dentro de una red, por ejemplo, a qué lugares puedo llegar en una red de carreteras sin gastar más de un valor determinado (el coste puede ser distancia o tiempo).

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial representando red	`INPUT`	[vectorial: lineal]	Capa vectorial lineal representando la red a ser cubierta
Capa vectorial con puntos de inicio	`START_POINTS`	[vector: punto]	Capa vectorial de puntos cuyas entidades son usadas como puntos iniciales para generar las áreas de servicio
Tipo de ruta a calcular	`STRATEGY`	[enumeración] Predeterminado: 0	El tipo de ruta a calcular. Uno de: • 0 — Mas corta • 1 — Mas rápida
Coste del viaje (distancia para «Shortest», tiempo para «Fastest»)	`TRAVEL_COST`	[numérico: doble] Predeterminado: 0.0	El valor se estima como distancia (en unidades de la capa de red) cuando se busca el camino *Más corto* y como tiempo (en horas) para el camino *Más rápido*.
Área de servicio (líneas)	`OUTPUT_LINES`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de línea de salida para el área de servicio. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
** Área de servicio (nodos límite) **	`OUTPUT`	[vector: punto] Predeterminado: `[Omitir salida]`	Especifique la capa de puntos de salida para los nodos del límite del área de servicio. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

continúe en la próxima página

Tabla 24.222 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
**Objetos espaciales no enrutables*	OUTPUT_NON_ROU	[vector: punto] Predeterminado: [Omitir salida]	Especifique la salida que se utilizará para almacenar cualquier objeto espacial de entrada que no se haya podido enrutar (por ejemplo, aquellas que están demasiado lejos de la capa de red). *Una de*: • Saltar salida • Crear capa temporal (TEMPORARY_OUTPUT) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Campo Dirección Opcional	DIRECTION_FIEL	[campo de tabla: cualquiera]	El campo usado para especificar direcciones para las trazaos de red. Los valores utilizados en este campo se especifican con los tres parámetros Valor para la dirección hacia adelante, Valor para la dirección hacia atrás y Valor para ambas direcciones. Las direcciones hacia adelante y hacia atrás corresponden a un borde unidireccional, mientras que ambas direcciones indica un borde bidireccional. Si una característica no tiene un valor en este campo, o si no se ha establecido ningún campo, se utiliza la configuración de dirección predeterminada (proporcionada con el parámetro Dirección predeterminada).
Valor para dirección de avance Opcional	VALUE_FORWARD	[cadena de texto] Predeterminado: “” (cadena vacía)	Valor establecido en el campo de dirección para identificar los trazos con una dirección de avance
Valor para dirección de retorno Opcional	VALUE_BACKWARD	[cadena de texto] Predeterminado: “” (cadena vacía)	Valor establecido en el campo de dirección para identificar los trazos con una dirección de retorno
Valor para ambas direcciones Opcional	VALUE_BOTH	[cadena de texto] Predeterminado: “” (cadena vacía)	Valor establecido en el campo dirección para identificar trazos bidireccionales

continúe en la próxima página

Tabla 24.223 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Dirección predeterimnada	DEFAULT_DIRECT	[enumeración] Preestablecido: 2	Si una entidad no tiene ningún valor establecido en el campo de dirección o si no se establece ningún campo de dirección, se utiliza este valor de dirección. Uno de: • 0 — Dirección de avance • 1 — Dirección de retorno • 2 — Ambas direcciones
Campo velocidad Opcional	SPEED_FIELD	[campo de tabla: numérico]	Campo que proporciona el valor de velocidad (en ``km/h “”) para los tramos de la red cuando se busca la ruta más rápida. Si una función no tiene un valor en este campo, o no se establece ningún campo, se utiliza el valor de velocidad predeterminado (proporcionado con el parámetro `Velocidad predeterminada`).
Velocidad por Defecto (km/h)	DEFAULT_SPEED	[numérico: doble] Predeterminada: 50.0	Valor que se utilizará para calcular el tiempo de viaje si no se proporciona ningún valor de velocidad para un borde en el campo especificado.
Tolerancia de Topología	TOLERANCE	[numérico: doble] Predeterminado: 0.0	Dos líneas con nodos mas cercanos que la tolerancia especificada se consideran conectados
Incluir puntos límite superior/inferior	INCLUDE_BOUNDS	[booleano] Predeterminado: False	Crea una salida de capa de puntos con dos puntos para cada tramo en los límites del área de servicio. Un punto es el comienzo de ese tramo, el otro es el final.
Distancia máxima en puntos desde la red Opcional	POINT_TOLERANC	[numérico: doble] Predeterminado: No establecido	Especifica un límite opcional en la distancia desde los puntos hasta la capa de red. Si un punto está más lejos de la red que esta distancia, se tratará como no enrutable. Si no se establece, los puntos finales se ajustarán al punto más cercano en la capa de red, independientemente de lo lejos que estén realmente de la red.

Salidas

Etiqueta	Nombre	Tipo	Descripción
** Área de servicio (nodos límite) **	OUTPUT	[vector: punto]	La capa de punto de salida con los nodos de límite del área de servicio.
Área de servicio (líneas)	OUTPUT_LINES	[vectorial: lineal]	Capa lineal que representa las partes de la red que pueden ser atendidas por los puntos de inicio, para el costo dado.
Objetos espaciales no enrutables	OUTPUT_NON_ROU	[vector: punto]	Una salida opcional que se utilizará para almacenar cualquier característica de entrada que no haya podido enrutarse (por ejemplo, aquellas que se encuentran demasiado lejos de la capa de red).

Código Python

Algorithm ID: `native:serviceareafromlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

área de servicio (desde punto)

Devuelve todos los bordes o partes de bordes de una red a los que se puede llegar dentro de una distancia o tiempo determinados, partiendo de un objeto espacial punto. Esto permite evaluar la accesibilidad dentro de una red, por ejemplo, a qué lugares puedo llegar en una red de carreteras sin gastar más de un valor determinado (el coste puede ser distancia o tiempo).

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial representando la red	`INPUT`	[vectorial: lineal]	Capa vectorial lineal representando la red a ser cubierta
Tipo de ruta a calcular	`STRATEGY`	[enumeración] Predeterminado: 0	El tipo de ruta a calcular. Uno de: • 0 — Mas corta • 1 — Mas rápida
Punto de partida	`START_POINT`	[coordenadas]	Coordenadas del punto alrededor del cual calcular el área de servicio. Pulse el botón … situado junto a la opción y haga clic en el lienzo para rellenar el parámetro con la coordenada del punto en el que ha hecho clic.
Coste del viaje (distancia para «Shortest», tiempo para «Fastest»)	`TRAVEL_COST`	[numérico: doble] Predeterminado: 0.0	El valor se estima como distancia (en unidades de la capa de red) cuando se busca el camino *Más corto* y como tiempo (en horas) para el camino *Más rápido*.

continúe en la próxima página

Tabla 24.225 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Área de servicio (líneas) Opcional	`OUTPUT_LINES`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de línea de salida para el área de servicio. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
** Área de servicio (nodos límite) ** Opcional	`OUTPUT`	[vector: punto] Predeterminado: `[Omitir salida]`	Especifique la capa de puntos de salida para los nodos del límite del área de servicio. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Campo Dirección Opcional	`DIRECTION_FIEL`	[campo de tabla: cualquiera]	El campo usado para especificar direcciones para las trazaos de red. Los valores utilizados en este campo se especifican con los tres parámetros `Valor para la dirección hacia adelante`, `Valor para la dirección hacia atrás` y `Valor para ambas direcciones`. Las direcciones hacia adelante y hacia atrás corresponden a un borde unidireccional, mientras que `ambas direcciones` indica un borde bidireccional. Si una característica no tiene un valor en este campo, o si no se ha establecido ningún campo, se utiliza la configuración de dirección predeterminada (proporcionada con el parámetro `Dirección predeterminada`).
Valor para dirección de avance Opcional	`VALUE_FORWARD`	[cadena de texto] Predeterminado: “” (cadena vacía)	Valor establecido en el campo de dirección para identificar los trazos con una dirección de avance
Valor para dirección de retorno Opcional	`VALUE_BACKWARD`	[cadena de texto] Predeterminado: “” (cadena vacía)	Valor establecido en el campo de dirección para identificar los trazos con una dirección de retorno

continúe en la próxima página

Tabla 24.226 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Valor para ambas direcciones Opcional	`VALUE_BOTH`	[cadena de texto] Predeterminado: "" (cadena vacía)	Valor establecido en el campo dirección para identificar trazos bidireccionales
Dirección predeterimnada	`DEFAULT_DIRECT`	[enumeración] Preestablecido: 2	Si una entidad no tiene ningún valor establecido en el campo de dirección o si no se establece ningún campo de dirección, se utiliza este valor de dirección. Uno de: • 0 — Dirección de avance • 1 — Dirección de retorno • 2 — Ambas direcciones
Campo velocidad Opcional	`SPEED_FIELD`	[campo de tabla: numérico]	Campo que proporciona el valor de velocidad (en ``km/h "") para los tramos de la red cuando se busca la ruta más rápida. Si una función no tiene un valor en este campo, o no se establece ningún campo, se utiliza el valor de velocidad predeterminado (proporcionado con el parámetro `Velocidad predeterminada`).
Velocidad por Defecto (km/h)	`DEFAULT_SPEED`	[numérico: doble] Predeterminada: 50.0	Valor que se utilizará para calcular el tiempo de viaje si no se proporciona ningún valor de velocidad para un borde en el campo especificado.
Tolerancia de Topología	`TOLERANCE`	[numérico: doble] Predeterminado: 0.0	Dos líneas con nodos mas cercanos que la tolerancia especificada se consideran conectados
Distancia máxima en puntos desde la red Opcional	`POINT_TOLERANC`	[numérico: doble] Predeterminado: 0.0	Especifica un límite opcional para la distancia desde el punto de inicio hasta la capa de red. Si el punto se encuentra más lejos de la red que esta distancia, se generará un error. Si no se establece, el punto se ajustará al punto más cercano de la capa de red, independientemente de la distancia real a la que se encuentre de la red.
Incluir puntos límite superior/inferior	`INCLUDE_BOUNDS`	[booleano] Predeterminado: False	Crea una salida de capa de puntos con dos puntos para cada tramo en los límites del área de servicio. Un punto es el comienzo de ese tramo, el otro es el final.

Salidas

Etiqueta	Nombre	Tipo	Descripción
** Área de servicio (nodos límite) **	`OUTPUT`	[vector: punto]	La capa de punto de salida con los nodos de límite del área de servicio.
Área de servicio (líneas)	`OUTPUT_LINES`	[vectorial: lineal]	Capa lineal que representa las partes de la red que pueden ser atendidas por el punto de inicio, por el costo dado.

Código Python

Algorithm ID: `native:serviceareafrompoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Ruta mas corta (capa a punto)

Calcula las rutas óptimas (mas corta o mas rápida) para múltiples puntos iniciales definidos por un capa vectorial y un punto de destino dado.

> **Nota**
>
> Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial representando red	`INPUT`	[vectorial: lineal]	Capa vectorial lineal representando la red a ser cubierta
Tipo de ruta a calcular	`STRATEGY`	[enumeración] Predeterminado: 0	El tipo de ruta a calcular. Uno de: • 0 — Mas corta • 1 — Mas rápida
Capa vectorial con puntos de inicio	`START_POINTS`	[vector: punto]	Capa vectorial de puntos cuyas entidades son empleadas como puntos iniciales de las rutas
punto Destino (x, y)	`END_POINT`	[coordenadas]	Entidad puntual que representa el punto final de las rutas Pulse el botón … situado junto a la opción y haga clic en el lienzo para rellenar el parámetro con la coordenada del punto en el que ha hecho clic.
Ruta mas corta	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de línea de salida para las rutas más cortas. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

continúe en la próxima página

Tabla 24.228 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
**Objetos espaciales no enrutables* Opcional	`OUTPUT_NON_ROU`	[vector: punto] Predeterminado: `[Omitir salida]`	Especifique la salida que se utilizará para almacenar cualquier objeto espacial de entrada que no se haya podido enrutar (por ejemplo, aquellas que están demasiado lejos de la capa de red). *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Campo Dirección Opcional	`DIRECTION_FIEL`	[campo de tabla: cualquiera]	El campo usado para especificar direcciones para las trazaos de red. Los valores utilizados en este campo se especifican con los tres parámetros `Valor para la dirección hacia adelante`, `Valor para la dirección hacia atrás` y `Valor para ambas direcciones`. Las direcciones hacia adelante y hacia atrás corresponden a un borde unidireccional, mientras que `ambas direcciones` indica un borde bidireccional. Si una característica no tiene un valor en este campo, o si no se ha establecido ningún campo, se utiliza la configuración de dirección predeterminada (proporcionada con el parámetro `Dirección predeterminada`).
Valor para dirección de avance Opcional	`VALUE_FORWARD`	[cadena de texto] Predeterminado: “” (cadena vacía)	Valor establecido en el campo de dirección para identificar los trazos con una dirección de avance
Valor para dirección de retorno Opcional	`VALUE_BACKWARD`	[cadena de texto] Predeterminado: “” (cadena vacía)	Valor establecido en el campo de dirección para identificar los trazos con una dirección de retorno
Valor para ambas direcciones Opcional	`VALUE_BOTH`	[cadena de texto] Predeterminado: “” (cadena vacía)	Valor establecido en el campo dirección para identificar trazos bidireccionales

continúe en la próxima página

Tabla 24.229 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Dirección predeterimnada	`DEFAULT_DIRECT`	[enumeración] Preestablecido: 2	Si una entidad no tiene ningún valor establecido en el campo de dirección o si no se establece ningún campo de dirección, se utiliza este valor de dirección. Uno de: • 0 — Dirección de avance • 1 — Dirección de retorno • 2 — Ambas direcciones
Campo velocidad Opcional	`SPEED_FIELD`	[campo de tabla: numérico]	Campo que proporciona el valor de velocidad (en ``km/h “”) para los tramos de la red cuando se busca la ruta más rápida. Si una función no tiene un valor en este campo, o no se establece ningún campo, se utiliza el valor de velocidad predeterminado (proporcionado con el parámetro `Velocidad predeterminada`).
Velocidad por Defecto (km/h)	`DEFAULT_SPEED`	[numérico: doble] Predeterminada: 50.0	Valor que se utilizará para calcular el tiempo de viaje si no se proporciona ningún valor de velocidad para un borde en el campo especificado.
Tolerancia de Topología	`TOLERANCE`	[numérico: doble] Predeterminado: 0.0	Dos líneas con nodos mas cercanos que la tolerancia especificada se consideran conectados
Distancia máxima en puntos desde la red Opcional	`POINT_TOLERANC`	[numérico: doble] Predeterminado: No establecido	Especifica un límite opcional en la distancia entre los puntos inicial y final y la capa de red. Si una característica inicial se encuentra más lejos de la red que esta distancia, se considerará no enrutable. Si el punto final se encuentra más lejos de la red que esta distancia, se generará un error. Si no se establece, los puntos se ajustarán al punto más cercano de la capa de red, independientemente de la distancia real a la que se encuentren de la red.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ruta mas corta	`OUTPUT`	[vectorial: lineal]	Capa lineal de la ruta mas corta o mas rápida desde cada uno de los puntos de inicio hasta el punto final
Objetos espaciales no enrutables	`OUTPUT_NON_ROU`	[vector: punto]	Una capa de salida opcional que se utilizará para almacenar cualquier objeto espacial de entrada que no haya podido ser enrutada (por ejemplo, aquellas que se encuentran demasiado lejos de la capa de red).

Código Python

Algoritmo ID: `native:shortestpathlayertopoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Ruta mas corta (punto a capa)

Calcula las rutas óptimas (más cortas o más rápidas) entre un punto de inicio determinado y varios puntos finales definidos por una capa vectorial de puntos.

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial representando red	`INPUT`	[vectorial: lineal]	Capa vectorial lineal representando la red a ser cubierta
Tipo de ruta a calcular	`STRATEGY`	[enumeración] Predeterminado: 0	El tipo de ruta a calcular. Uno de: • 0 — Mas corta • 1 — Mas rápida
Punto de partida	`START_POINT`	[coordenadas]	Entidad de punto que representa el punto de inicio de las rutas Pulse el botón … situado junto a la opción y haga clic en el lienzo para rellenar el parámetro con la coordenada del punto en el que ha hecho clic.
Capa vectorial con puntos finales	`END_POINTS`	[vector: punto]	Capa de vector de puntos cuyas entidades se utilizan como puntos finales de las rutas

continúe en la próxima página

Tabla 24.231 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Ruta mas corta	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de línea de salida para las rutas más cortas. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
**Objetos espaciales no enrutables*	`OUTPUT_NON_ROU` Opcional	[vector: punto] Predeterminado: `[Omitir salida]`	Especifique la salida que se utilizará para almacenar cualquier objeto espacial de entrada que no se haya podido enrutar (por ejemplo, aquellas que están demasiado lejos de la capa de red). *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Campo Dirección Opcional	`DIRECTION_FIEL`	[campo de tabla: cualquiera]	El campo usado para especificar direcciones para las trazaos de red. Los valores utilizados en este campo se especifican con los tres parámetros `Valor para la dirección hacia adelante`, `Valor para la dirección hacia atrás` y `Valor para ambas direcciones`. Las direcciones hacia adelante y hacia atrás corresponden a un borde unidireccional, mientras que `ambas direcciones` indica un borde bidireccional. Si una característica no tiene un valor en este campo, o si no se ha establecido ningún campo, se utiliza la configuración de dirección predeterminada (proporcionada con el parámetro `Dirección predeterminada`).
Valor para dirección de avance Opcional	`VALUE_FORWARD`	[cadena de texto] Predeterminado: “” (cadena vacía)	Valor establecido en el campo de dirección para identificar los trazos con una dirección de avance
Valor para dirección de retorno Opcional	`VALUE_BACKWARD`	[cadena de texto] Predeterminado: “” (cadena vacía)	Valor establecido en el campo de dirección para identificar los trazos con una dirección de retorno

continúe en la próxima página

Tabla 24.232 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Valor para ambas direcciones Opcional	`VALUE_BOTH`	[cadena de texto] Predeterminado: "" (cadena vacía)	Valor establecido en el campo dirección para identificar trazos bidireccionales
Dirección predeterimnada	`DEFAULT_DIRECT`	[enumeración] Preestablecido: 2	Si una entidad no tiene ningún valor establecido en el campo de dirección o si no se establece ningún campo de dirección, se utiliza este valor de dirección. Uno de: • 0 — Dirección de avance • 1 — Dirección de retorno • 2 — Ambas direcciones
Campo velocidad Opcional	`SPEED_FIELD`	[campo de tabla: numérico]	Campo que proporciona el valor de velocidad (en ``km/h "") para los tramos de la red cuando se busca la ruta más rápida. Si una función no tiene un valor en este campo, o no se establece ningún campo, se utiliza el valor de velocidad predeterminado (proporcionado con el parámetro `Velocidad predeterminada`).
Velocidad por Defecto (km/h)	`DEFAULT_SPEED`	[numérico: doble] Predeterminada: 50.0	Valor que se utilizará para calcular el tiempo de viaje si no se proporciona ningún valor de velocidad para un borde en el campo especificado.
Tolerancia de Topología	`TOLERANCE`	[numérico: doble] Predeterminado: 0.0	Dos líneas con nodos mas cercanos que la tolerancia especificada se consideran conectados
Distancia máxima en puntos desde la red Opcional	`POINT_TOLERANC`	[numérico: doble] Predeterminado: No establecido	Especifica un límite opcional en la distancia entre los puntos inicial y final y la capa de red. Si el punto inicial se encuentra más lejos de la red que esta distancia, se generará un error. Si una característica final se encuentra más lejos de la red que esta distancia, se considerará no enrutable. Si no se establece, los puntos se ajustarán al punto más cercano de la capa de red, independientemente de la distancia real a la que se encuentren de la red.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ruta mas corta	`OUTPUT`	[vectorial: lineal]	Capa lineal de la ruta mas corta o mas rápida desde cada uno de los puntos de inicio hasta el punto final
Objetos espaciales no enrutables	`OUTPUT_NON_ROU`	[vector: punto]	Una capa de salida opcional que se utilizará para almacenar cualquier objeto espacial de entrada que no haya podido ser enrutada (por ejemplo, aquellas que se encuentran demasiado lejos de la capa de red).

Código Python

Algoritmo ID: `native:shortestpathpointtolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Ruta mas corta (punto a punto)

Calcula la ruta óptima (más corta o más rápida) entre un punto de inicio y un punto final determinados.

> **Nota**
>
> Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial representando red	`INPUT`	[vectorial: lineal]	Capa vectorial lineal representando la red a ser cubierta
Tipo de ruta a calcular	`STRATEGY`	[enumeración] Predeterminado: 0	El tipo de ruta a calcular. Uno de: • 0 — Mas corta • 1 — Mas rápida
Punto inicial (x, y)	`START_POINT`	[coordenadas]	Punto que representa el punto de inicio de las rutas. Pulse el botón … situado junto a la opción y haga clic en el lienzo para rellenar el parámetro con la coordenada del punto en el que ha hecho clic.
punto Destino (x, y)	`END_POINT`	[coordenadas]	Punto que representa el punto final de las rutas. Pulse el botón … situado junto a la opción y haga clic en el lienzo para rellenar el parámetro con la coordenada del punto en el que ha hecho clic.
Ruta mas corta	`OUTPUT`	[vectorial: lineal]	Especifique la capa de línea de salida para las rutas más cortas. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Campo Dirección Opcional	DIRECTION_FIEL	[campo de tabla: cualquiera]	El campo usado para especificar direcciones para las trazaos de red. Los valores utilizados en este campo se especifican con los tres parámetros Valor para la dirección hacia adelante, Valor para la dirección hacia atrás y Valor para ambas direcciones. Las direcciones hacia adelante y hacia atrás corresponden a un borde unidireccional, mientras que ambas direcciones indica un borde bidireccional. Si una característica no tiene un valor en este campo, o si no se ha establecido ningún campo, se utiliza la configuración de dirección predeterminada (proporcionada con el parámetro Dirección predeterminada).
Valor para dirección de avance Opcional	VALUE_FORWARD	[cadena de texto] Predeterminado: "" (cadena vacía)	Valor establecido en el campo de dirección para identificar los trazos con una dirección de avance
Valor para dirección de retorno Opcional	VALUE_BACKWARD	[cadena de texto] Predeterminado: "" (cadena vacía)	Valor establecido en el campo de dirección para identificar los trazos con una dirección de retorno
Valor para ambas direcciones Opcional	VALUE_BOTH	[cadena de texto] Predeterminado: "" (cadena vacía)	Valor establecido en el campo dirección para identificar trazos bidireccionales
Dirección predeterimnada	DEFAULT_DIRECT	[enumeración] Preestablecido: 2	Si una entidad no tiene ningún valor establecido en el campo de dirección o si no se establece ningún campo de dirección, se utiliza este valor de dirección. Uno de: • 0 — Dirección de avance • 1 — Dirección de retorno • 2 — Ambas direcciones
Campo velocidad Opcional	SPEED_FIELD	[campo de tabla: numérico]	Campo que proporciona el valor de velocidad (en ``km/h "") para los tramos de la red cuando se busca la ruta más rápida. Si una función no tiene un valor en este campo, o no se establece ningún campo, se utiliza el valor de velocidad predeterminado (proporcionado con el parámetro Velocidad predeterminada).
Velocidad por Defecto (km/h)	DEFAULT_SPEED	[numérico: doble] Predeterminada: 50.0	Valor que se utilizará para calcular el tiempo de viaje si no se proporciona ningún valor de velocidad para un borde en el campo especificado.
Tolerancia de Topología	TOLERANCE	[numérico: doble] Predeterminado: 0.0	Dos líneas con nodos mas cercanos que la tolerancia especificada se consideran conectados

Distancia máxima en puntos desde la red Opcional	`POINT_TOLERANC`	[numérico: doble] Predeterminado: No establecido	Especifica un límite opcional en la distancia entre los puntos inicial y final y la capa de red. Si cualquiera de los dos puntos se encuentra más lejos de la red que esta distancia, se generará un error. Si no se establece, los puntos se ajustarán al punto más cercano de la capa de red, independientemente de la distancia real a la que se encuentren de la red.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ruta mas corta	`OUTPUT`	[vectorial: lineal]	Capa de línea de la ruta más corta o más rápida desde cada uno de los puntos de inicio hasta el punto final

Código Python

Algoritmo ID: `native:shortestpathpointtopoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.14 Gráficos

Gráfico de barras

Crea un diagrama de barras a partir de una categoría y un campo de capa.

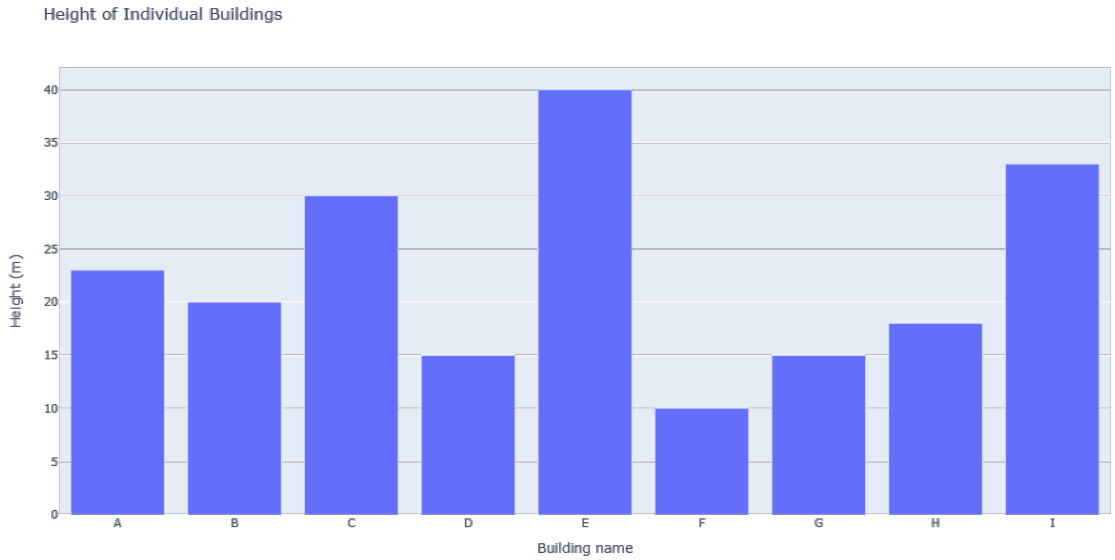


Figura 24.32: Bar plot example

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Nombre de campo de Categoría	`NAME_FIELD`	[campo de tabla: cualquiera]	Campo categórico que se utilizará para agrupar las barras (eje X)
Campo Valor	`VALUE_FIELD`	[campo de tabla: numérico]	Valor a usar para el diagrama (Eje Y).
Title Optional	`TITLE`	[string] Default: «»	Title of the plot
X-axis Title Optional	`XAXIS_TITLE`	[string] Default: «»	If empty, the name of the category field is used. With a single space, the axis title is hidden.
Y-axis Title Optional	`YAXIS_TITLE`	[string] Default: «»	If empty, the name of the value field is used. With a single space, the axis title is hidden.
Diagrama de Barras	`OUTPUT`	[html] Predeterminado: `[Save to temporary file]`	Especifique el archivo HTML para el gráfico. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Diagrama de Barras	`OUTPUT`	[html]	Archivo HTML con el diagrama. Disponible en *Procesos -> Visor de Resultados*.

Código Python

Algorithm ID: `qgis:barplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Cuadro de Diagrama

Crea un diagrama de caja a partir de un campo de categoría y un campo de capa numérico.

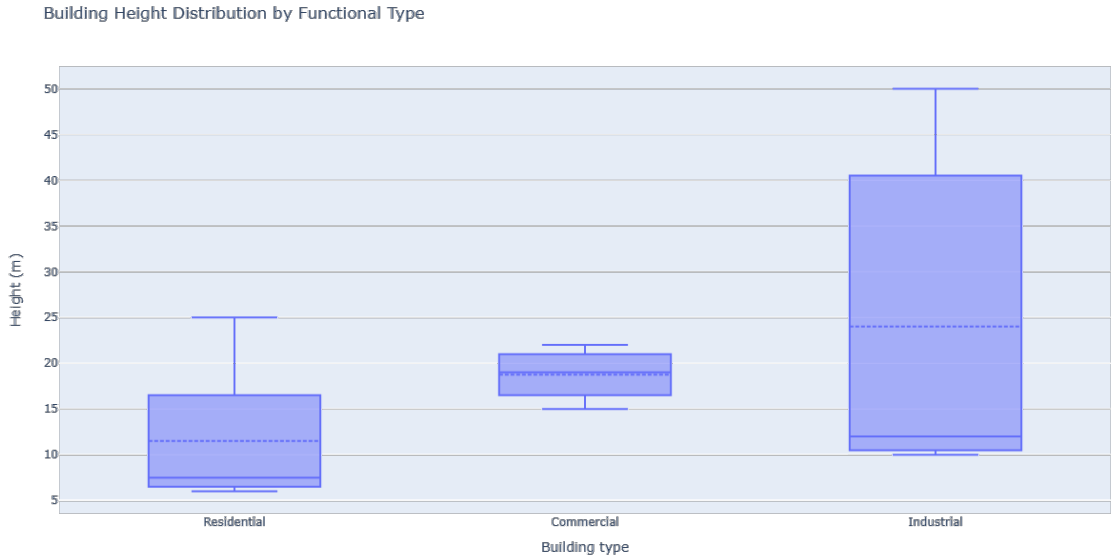


Figura 24.33: Example of box plot

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Nombre de campo de Categorías	`NAME_FIELD`	[campo de tabla: cualquiera]	Campo de categorías a usar para agrupar las cajas (eje X)
Campo Valor	`VALUE_FIELD`	[campo de tabla: numérico]	Valor a usar para el diagrama (Eje Y).
Lïneas estadísticas adicionales	`MSD`	[enumeración] Predeterminado: 0	Información estadística adicional para agregar al diagrama. Uno de: • 0 — Mostrar Media • 1 — Mostrar Desviación Estándar • 2 — No muestrar la desviación estándar ni la media
Title Optional	`TITLE`	[string] Default: «»	Title of the plot

continúe en la próxima página

Tabla 24.239 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
X-axis Title Optional	`XAXIS_TITLE`	[string] Default: «»	If empty, the name of the category field is used. With a single space, the axis title is not shown.
Y-axis Title Optional	`YAXIS_TITLE`	[string] Default: «»	If empty, the name of the value field is used. With a single space, the axis title is not shown.
Diagrama de Caja	`OUTPUT`	[html] Predeterminado: `[Save to temporary file]`	Especifique el archivo HTML para el gráfico. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Diagrama de Caja	`OUTPUT`	[html]	Archivo HTML con el diagrama. Disponible en *Procesos* -> *Visor de Resultados*.

Código Python

Algorithm ID: `qgis:boxplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Generate elevation profile image

Creates an elevation profile image from a list of map layers and an optional terrain layer.

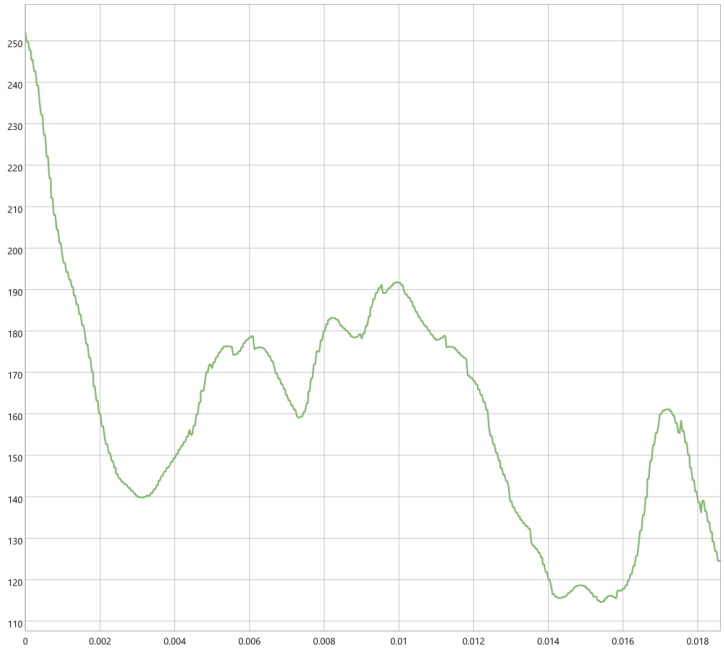


Figura 24.34: Example of elevation profile

Parámetros

Basic parameters

Etiqueta	Nombre	Tipo	Descripción
Profile curve	`CURVE`	[geometry: line]	The LineString geometry (WKT or GeoJSON) representing the profile curve along which the elevation profile will be generated
Map layers	`MAP_LAYERS`	[layer] [list]	The list of map layers to include in the elevation profile
Chart width (in pixels)	`WIDTH`	[numérico: entero] Default: 400 Minimum value: 0	The width of the output chart in pixels.
Chart height (in pixels)	`HEIGHT`	[numérico: entero] Default: 300 Minimum value: 0	The height of the output chart in pixels.
Terrain layer Optional	`TERRAIN_LAYER`	[raster]	A terrain layer (e.g., DEM) to use for elevation data. If not provided, elevation data will be derived from the map layers.

Advanced parameters

Etiqueta	Nombre	Tipo	Descripción
Chart minimum distance (X axis) Optional	MINIMUM_DISTAN	[numeric: double]	The minimum distance (X axis) to display on the chart. If not specified, the chart will auto-scale.
Chart maximum distance (X axis) Optional	MAXIMUM_DISTAN	[numeric: double]	The maximum distance (X axis) to display on the chart. If not specified, the chart will auto-scale.
Chart minimum elevation (Y axis) Optional	MINIMUM_ELEVAT	[numeric: double]	The minimum elevation (Y axis) to display on the chart. If not specified, the chart will auto-scale.
Chart maximum elevation (Y axis) Optional	MAXIMUM_ELEVAT	[numeric: double]	The maximum elevation (Y axis) to display on the chart. If not specified, the chart will auto-scale.
Chart text color Optional	TEXT_COLOR	[color]	The color of the text in the chart (e.g., axis labels, titles).
Chart background color Optional	BACKGROUND_COL	[color]	The background color of the chart.
Chart border color Optional	BORDER_COLOR	[color]	The color of the chart border.
Profile tolerance	TOLERANCE	[numeric: double] Default: 5.0 Minimum value: 0	Defines how far a feature (vector point, line, polygon, or point cloud) can be from the profile line to be included in the results. It uses map units and does not affect other layer types.
Chart DPI	DPI	[numérico: entero] Default: 96 Minimum value: 0	The resolution of the output image in dots per inch (DPI).
Output image	OUTPUT	[file] Predeterminado: [Save to temporary file]	Specify the image file for the plot. *One of*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Output image	OUTPUT	[file]	The generated elevation profile image in the specified format.

Código Python

Algorithm ID: `native:generateelevationprofileimage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Diagrama de media y desviación estándar

Crea un diagrama de caja con valores media y desviación estándar.

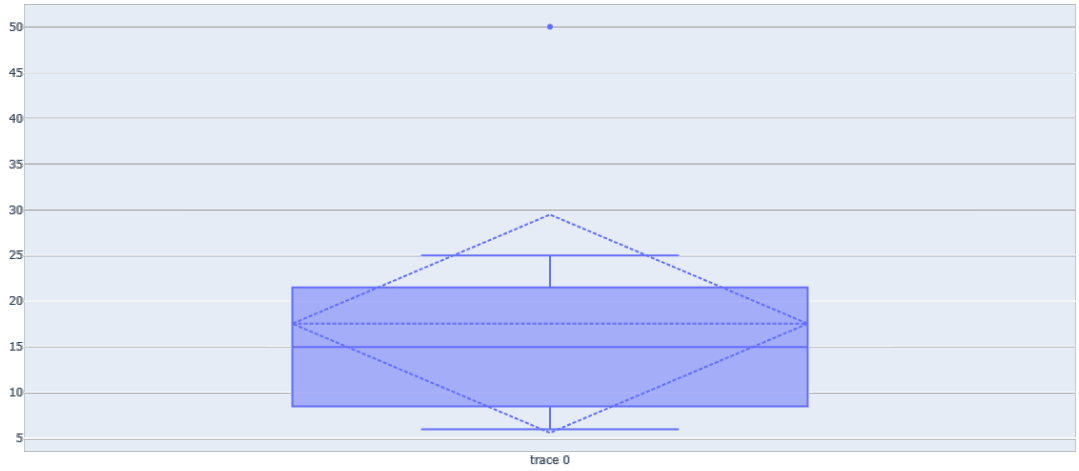


Figura 24.35: Example of mean and standard deviation plot

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Tabla de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Nombre de campo de Categorías	`NAME_FIELD`	[campo de tabla: cualquiera]	Campo de categorías a usar para agrupar las cajas (eje X)
Campo Valor	`VALUE_FIELD`	[campo de tabla: numérico]	Valor a usar para el diagrama (Eje Y).
Diagrama	`OUTPUT`	[html] Predeterminado: `[Save to temporary file]`	Especifique el archivo HTML para el gráfico. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Diagrama	`OUTPUT`	[html]	Archivo HTML con el diagrama. Disponible en *Procesos* -> *Visor de Resultados*.

Código Python

Algorithm ID: `qgis:meanandstandarddeviationplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Diagrama polar

Genera un diagrama polar basado en el valor de una capa de vector de entrada.

Se deben ingresar dos campos como parámetros: uno que define la categoría de cada objeto (para agrupar objetos) y otro con la variable a graficar (esta debe ser numérica).

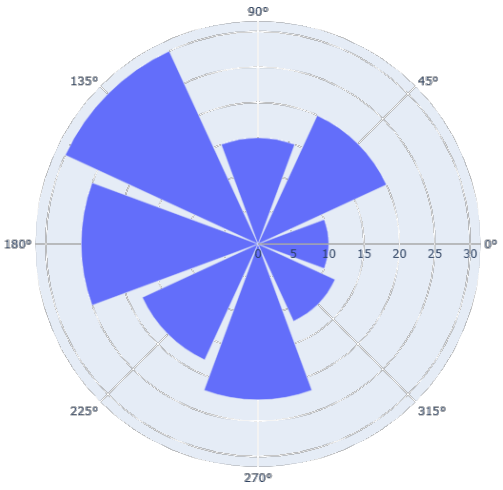


Figura 24.36: Example of polar plot

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Nombre de campo de Categorías	`NAME_FIELD`	[campo de tabla: cualquiera]	Campo de categorías a usar para agrupar las entidades (eje X)
Campo Valor	`VALUE_FIELD`	[campo de tabla: numérico]	Valor a usar para el diagrama (Eje Y).
Diagrama polar	`OUTPUT`	[html] Predeterminado: `[Save to temporary file]`	Especifique el archivo HTML para el gráfico. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Diagrama polar	`OUTPUT`	[html]	Archivo HTML con el diagrama. Disponible en *Procesos -> Visor de Resultados*.

Código Python

Algoritmo ID: `qgis:polarplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Histograma capa Ráster

Genera un histograma con los valores de una capa ráster.

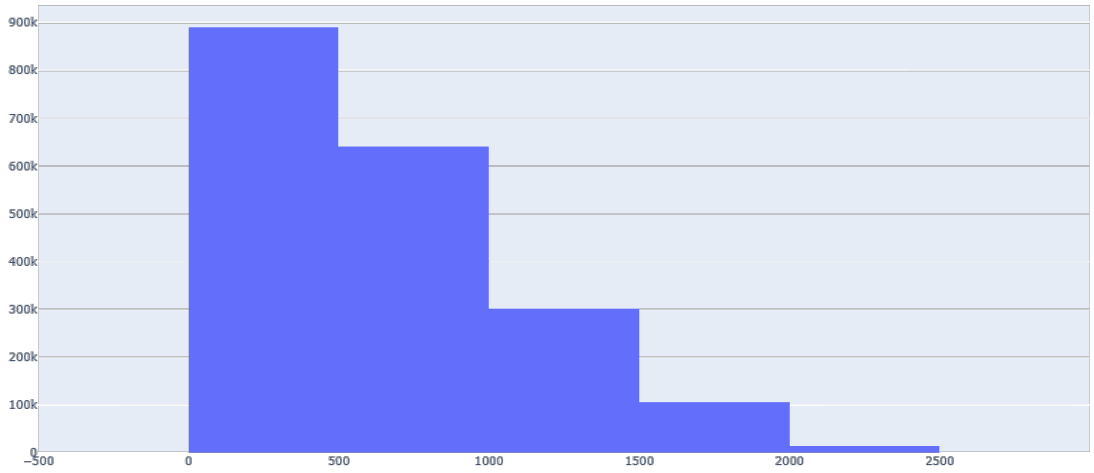


Figura 24.37: Raster layer histogram example

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[raster]	Capa ráster de entrada
Número de banda	`BAND`	[banda ráster]	Banda ráster a usar para el histograma
número de cajas	`BINS`	[numérico: entero] Predeterminado: 10	El número de cajas a usar en el histograma (eje X). Mínimo 2.
Histograma	`OUTPUT`	[html] Predeterminado: `[Save to temporary file]`	Especifique el archivo HTML para el gráfico. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Histograma	`OUTPUT`	[html]	Archivo HTML con el diagrama. Disponible en *Procesos* -> *Visor de Resultados*.

Código Python

Algoritmo ID: `qgis:rasterlayerhistogram`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Histograma de capa vectorial

Genera un histograma con los valores del atributo de una capa vectorial.

El atributo a usar para el cómputo del histograma debe ser genérico.

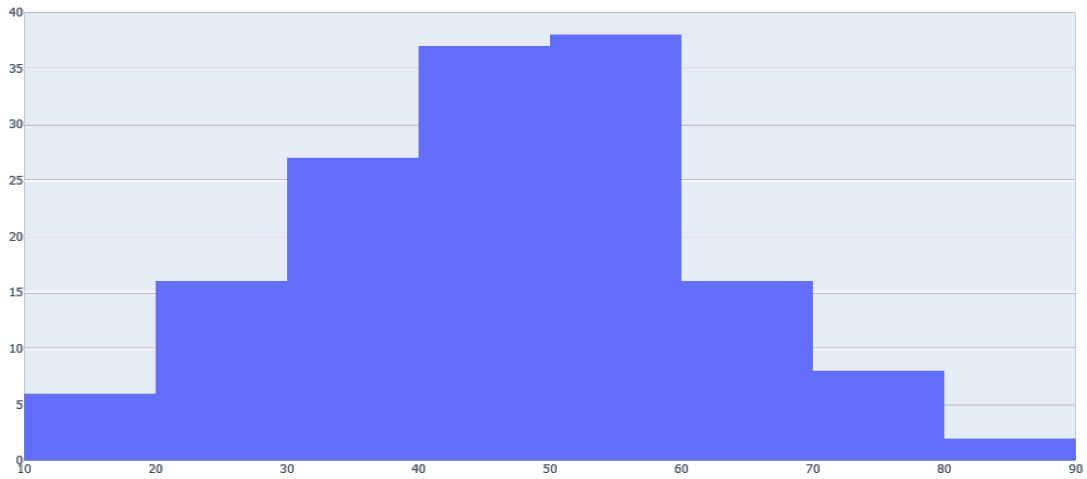


Figura 24.38: Example of vector layer histogram

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada
Atributo	`FIELD`	[campo de tabla: numérico]	Valor a usar para el diagrama (Eje Y).
número de cajas	`BINS`	[numérico: entero] Predeterminado: 10	El número de cajas a usar en el histograma (eje X). Mínimo 2.
Histograma	`OUTPUT`	[html] Predeterminado: `[Save to temporary file]`	Especifique el archivo HTML para el gráfico. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Histograma	`OUTPUT`	[html]	Archivo HTML con el diagrama. Disponible en *Procesos -> Visor de Resultados*.

Código Python

Algoritmo ID: `qgis:vectorlayerhistogram`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Capa vectorial del gráfico de dispersión

Crea un diagrama de dispersión simple `X` - `Y` para una capa vectorial

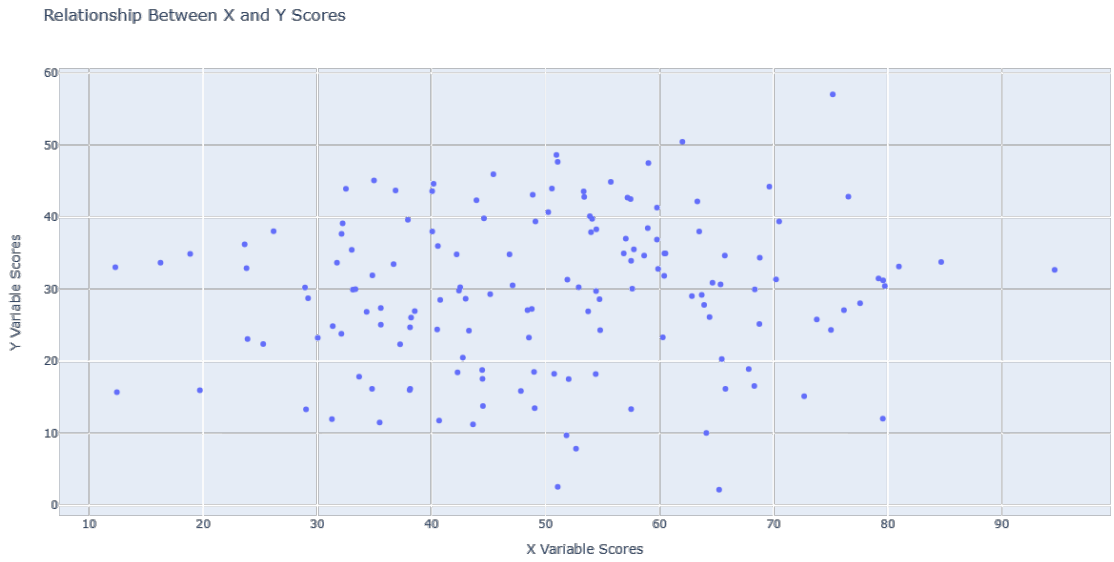


Figura 24.39: Example of vector layer scatterplot

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Atributo X	`XFIELD`	[campo de tabla: numérico]	Campo a usar para el eje X
Atributo Y	`YFIELD`	[campo de tabla: numérico]	Campo a usar para el eje Y
Hover text Optional	`HOVERTEXT`	[expression] Default: «»	Text to be shown when hovering with the mouse over data points. It can be picked from a field or based on an expression.
Title Optional	`TITLE`	[string] Default: «»	Title of the plot
X-axis Title Optional	`XAXIS_TITLE`	[string] Default: «»	If empty, the field name of the x attribute is used. With a single space, the axis title is not shown.
Y-axis Title Optional	`YAXIS_TITLE`	[string] Default: «»	If empty, the field name of the y attribute is used. With a single space, the axis title is not shown.
Use logarithmic scale for x-axis	`XAXIS_LOG`	[boolean] Default: False	When enabled, uses logarithmic scale for the x-axis
Use logarithmic scale for y-axis	`YAXIS_LOG`	[boolean] Default: False	When enabled, uses logarithmic scale for the y-axis
Diagrama de dispersión	`OUTPUT`	[html] Predeterminado: `[Save to temporary file]`	Especifique el archivo HTML para el gráfico. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Diagrama de dispersión	`OUTPUT`	[html]	Archivo HTML con el diagrama. Disponible en *Procesos -> Visor de Resultados*.

Código Python

Algoritmo ID: `qgis:vectorlayerscatterplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Diagrama de dispersión 3D de una capa vectorial

Crea un diagrama de dispersión 3D para una capa vectorial.

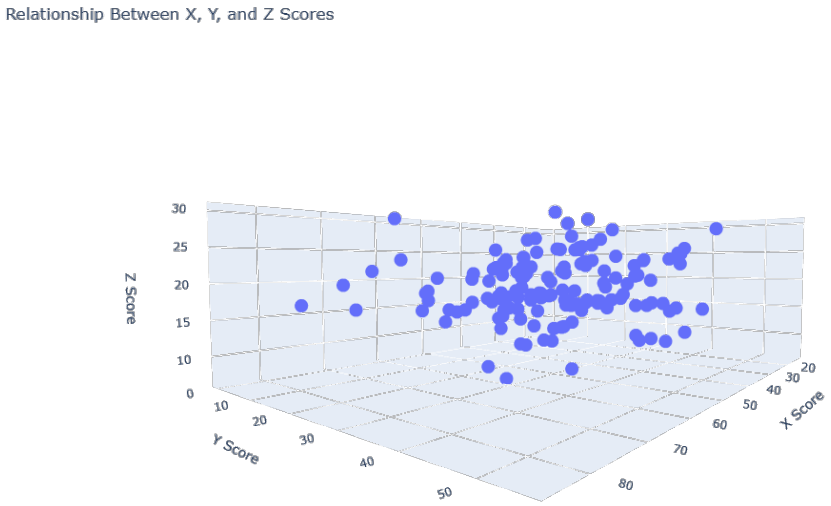


Figura 24.40: Example of 3D scatter plot

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Atributo X	`XFIELD`	[campo de tabla: numérico]	Campo a usar para el eje X
Atributo Y	`YFIELD`	[campo de tabla: numérico]	Campo a usar para el eje Y
Atributo Z	`ZFIELD`	[campo de tabla: numérico]	Campo a usar para el eje Z
Title Optional	`TITLE`	[string] Default: «»	Title of the plot
X-axis Title Optional	`XAXIS_TITLE`	[string] Default: «»	If empty, the field name of the X attribute is used.
Y-axis Title Optional	`YAXIS_TITLE`	[string] Default: «»	If empty, the field name of the Y attribute is used.
Z-axis Title Optional	`ZAXIS_TITLE`	[string] Default: «»	If empty, the field name of the Z attribute is used.
Scatterplot 3D	`OUTPUT`	[html] Predeterminado: `[Save to temporary file]`	Especifique el archivo HTML para el gráfico. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Scatterplot 3D	`OUTPUT`	[html]	Archivo HTML con el diagrama. Disponible en *Procesos* -> *Visor de Resultados*.

Código Python

Algoritmo ID: `qgis:scatter3dplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.15 Conversión de nubes de puntos

> **Atención**
>
> Running these algorithms requires QGIS installed with PDAL >= 2.5.0 (see *Help* ► *About* menu).

Convertir formato

Convierte una nube de puntos a un formato de archivo diferente, por ejemplo, crea un archivo comprimido `.LAZ`.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de nube de puntos de entrada a convertir
Convertido	`OUTPUT`	[nube de puntos] Predeterminado: `[Save to temporary file]`	Especifique el archivo de nube de puntos que se utilizará como salida. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Convertido	`OUTPUT`	[nube de puntos]	Da salida a la capa de nube de puntos en un formato de archivo modificado. Los formatos actualmente soportados son `.LAS`, `.LAZ`, `.COPC.LAZ` y `.VPC`.

Código Python

ID Algoritmo: `pdal:convertformat`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Exportar a ráster

Exporta los datos de la nube de puntos a una grid ráster 2D con un tamaño de celda de resolución dada, escribiendo los valores del atributo especificado.

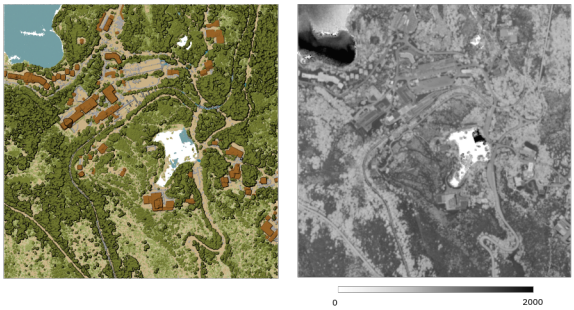


Figura 24.41: Salida raster utilizando el atributo Intensidad de los puntos

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de nube de puntos de entrada para exportar
Atributo	`ATTRIBUTE`	[campo] [enumeración]	A Campo de la capa de nube de puntos de la que extraer los valores
Resolución del ráster de densidad	`RESOLUTION`	[numérico: doble] Predeterminado: 1.0	Tamaño de celda del raster de salida

continúe en la próxima página

Tabla 24.258 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tamaño de las baldosas para recorridos paralelos	`TILE_SIZE`	[numérico: entero] Preestablecido: 1000	
Exportado	`OUTPUT`	[raster] Predeterminado: `[Save to temporary file]`	Especifique el archivo raster al que exportar los datos. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Expresión de filtrado Opcional	`FILTER_EXPRESS`	[expresión]	Una *expresión PDAL* para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos
Extensión del cultivo Opcional	`FILTER_EXTENT`	[extensión]	Una extensión del mapa para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Origen X de una tesela para ejecuciones paralelas Opcional	`ORIGIN_X`	[numérico: doble]	
Origen Y de una tesela para ejecuciones paralelas Opcional	`ORIGIN_Y`	[numérico: doble]	

Salidas

Etiqueta	Nombre	Tipo	Descripción
Exportado	`OUTPUT`	[raster]	Los objetos espaciales de la capa ráster de salida de la capa de nube de puntos se exportan a. El formato actualmente soportado es `.TIF`.

Código Python

ID Algoritmo: `pdal:exportraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Exportar ráster (mediante triangulación)

Exporta datos de nubes de puntos a un grid ráster 2D utilizando una triangulación de puntos e interpolando después los valores de las celdas a partir de los triángulos.

> **Nota**
>
> El uso de este algoritmo puede ser más lento si se trata de un gran conjunto de datos. Si su nube de puntos es densa, puede exportar sus puntos de tierra como un ráster utilizando el algoritmo *Exportar a ráster*.

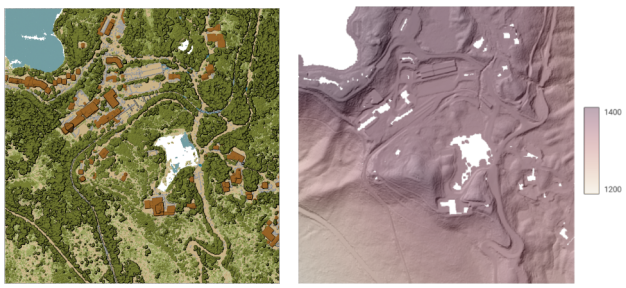


Figura 24.42: Raster del terreno generado por triangulación de nubes de puntos

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de nube de puntos de entrada para exportar

continúe en la próxima página

Tabla 24.261 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Resolución del ráster de densidad	`RESOLUTION`	[numérico: doble] Predeterminado: 1.0	Tamaño de celda del raster de salida
Tamaño de las baldosas para recorridos paralelos	`TILE_SIZE`	[numérico: entero] Preestablecido: 1000	
Exportado	`OUTPUT`	[raster] Predeterminado: `[Save to temporary file]`	Especifique el archivo raster al que exportar los datos. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Expresión de filtrado Opcional	`FILTER_EXPRESS`	[expresión]	Una *expresión PDAL* para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos
Extensión del cultivo Opcional	`FILTER_EXTENT`	[extensión]	Una extensión del mapa para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Origen X de una tesela para ejecuciones paralelas Opcional	`ORIGIN_X`	[numérico: doble]	
Origen Y de una tesela para ejecuciones paralelas Opcional	`ORIGIN_Y`	[numérico: doble]	

Salidas

Etiqueta	Nombre	Tipo	Descripción
Exportado (usando triangulación)	`OUTPUT`	[raster]	Los objetos espaciales de la capa ráster de salida de la capa de nube de puntos se exportan a. El formato actualmente soportado es `.TIF`.

Código Python

Identificación del algoritmo: `pdal:exportrastertin`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Exportar a vector

Exporta datos de nubes de puntos a una capa vectorial con puntos 3D (un GeoPackage), opcionalmente con atributos extra.

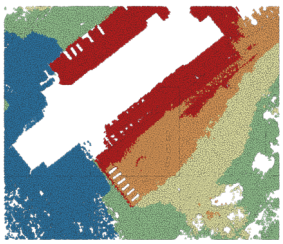


Figura 24.43: Exportación de la nube de puntos (puntos del terreno) a una capa vectorial con un estilo basado en la elevación

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de nube de puntos de entrada para exportar
Atributo Opcional	`ATTRIBUTE`	[field] [list]	Uno o más campos de la capa de nube de puntos a exportar con los puntos.
Exportado	`OUTPUT`	[vector] Predeterminado: `[Save to temporary file]`	Especifique el archivo vectorial al que exportar los datos. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Expresión de filtrado Opcional	`FILTER_EXPRESS`	[expresión]	Una *expresión PDAL* para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos
Extensión del cultivo Opcional	`FILTER_EXTENT`	[extensión]	Una extensión del mapa para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`

Salidas

Etiqueta	Nombre	Tipo	Descripción
Exportado	`OUTPUT`	[vector]	Los objetos espaciales de la capa vectorial de salida de la capa de nube de puntos se exportan a. El formato actualmente soportado es `.GPKG`.

Código Python

ID Algoritmo: `pdal:exportvector`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.16 Gestión de datos de nubes de puntos

Atención

Running these algorithms requires QGIS installed with PDAL >= 2.5.0 (see *Help* ► *About* menu).

Asignar proyección

Asigna un Sistema de Referencia de Coordenadas a una capa de nube de puntos, si falta o es incorrecto. Se crea una nueva capa.

Ver también

Reproyectar

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de nube de puntos de entrada a la que asignar un SRC
SRC Deseado	`CRS`	[src]	El SRC que se aplicará a la capa
Capa de salida	`OUTPUT`	[nube de puntos] Predeterinado: `[Save to temporary file]`	Especifique el archivo de nube de puntos que se utilizará como salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Convertido	`OUTPUT`	[nube de puntos]	Capa de nube de puntos de salida con un nuevo SRC. Los formatos actualmente admitidos son `.LAS`, `.LAZ`, `.COPC.LAZ` y `.VPC`.

Código Python

Identificación del algoritmo: `pdal:assignprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear una nube de puntos virtual (VPC)

Crea una *nube de puntos virtual (VPC)* a partir de los datos de la nube de puntos de entrada.

Si deja los parámetros opcionales sin marcar, el archivo VPC se construirá muy rápidamente ya que el algoritmo sólo leerá los metadatos de los archivos de entrada. Con cualquiera de los parámetros opcionales activados, el algoritmo leerá todos los puntos, lo que puede llevar algún tiempo.

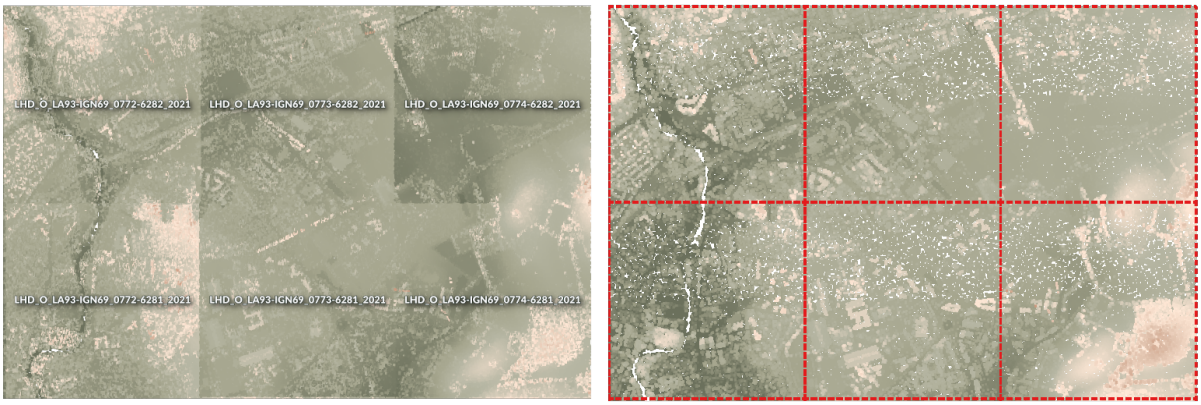


Figura 24.44: Generación de nubes de puntos virtuales con visión de conjunto a partir de un conjunto de mosaicos de nubes de puntos

Ver también

Mezclar

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`LAYERS`	[point cloud] [list]	Capas de nubes de puntos de entrada para combinar dentro de una capa de nube de puntos virtual
Calcular polígono límite	`BOUNDARY`	[booleano] Predeterminado: Falso	Establecer en True para mostrar los límites exactos de los datos (en lugar de sólo la extensión rectangular)
Calcular estadísticas	`STATISTICS`	[booleano] Predeterminado: Falso	Establecer en True para comprender los rangos de valores de los distintos atributos
Construir una nube de puntos general	`OVERVIEW`	[booleano] Predeterminado: Falso	Genera una única nube de puntos «diluida» de todos los datos de entrada (utilizando sólo cada 1000 puntos de los datos originales). La nube de puntos de visión general se creará junto al archivo VPC - por ejemplo, para `mydata.vpc`, la nube de puntos de visión general se llamaría `mydata-overview.copc.laz`.

continúe en la próxima página

Tabla 24.269 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Nube de puntos virtual	`OUTPUT`	[nube de puntos] Predeterinado: `[Save to temporary file]`	Especifique el archivo de nube de puntos en el que construir los datos. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Nube de puntos virtual	`OUTPUT`	[ráster]	Capa de nube de puntos de salida que combina todos los datos de entrada, como archivo virtual.

Código Python

Identificación del algoritmo: `pdal:virtualpointcloud`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cortar

Recorta una capa de nube de puntos por una capa de polígonos para que la nube de puntos resultante contenga sólo puntos dentro de los polígonos.

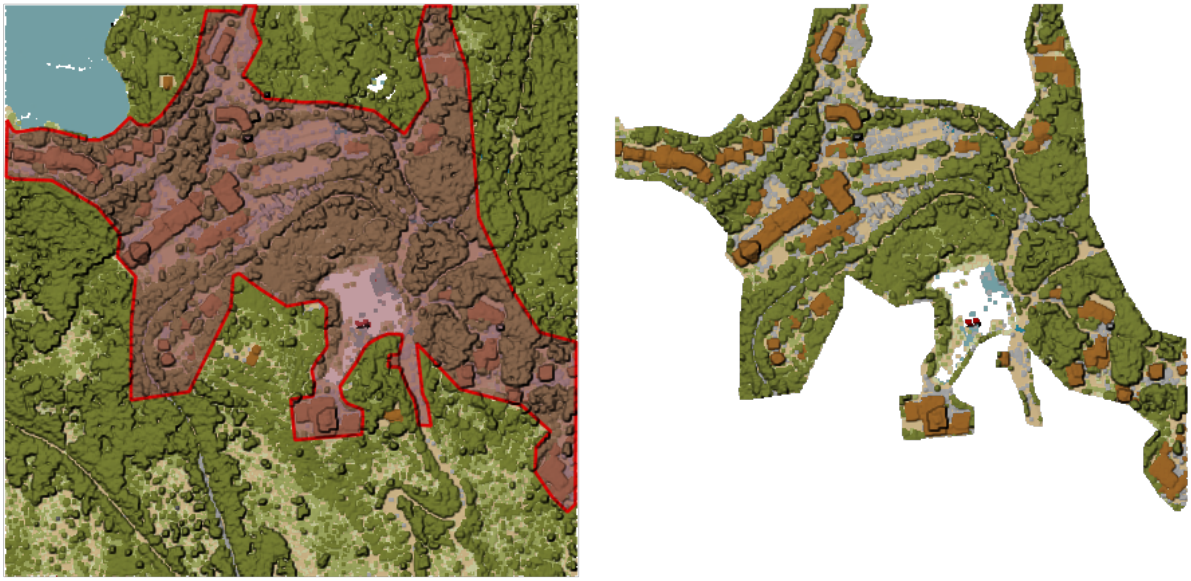


Figura 24.45: Recorte de una capa de nube de puntos de entrada con una cobertura poligonal

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de la nube de puntos de entrada para recortar
Recorte de polígonos	`OVERLAY`	[vectorial: poligonal]	Capa vectorial poligonal que se utilizará como cobertura para recortar los puntos
Cortadas	`OUTPUT`	[nube de puntos] Predeterinado: `[Save to temporary file]`	Especifique el archivo de nube de puntos al que exportar los puntos recortados. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Expresión de filtrado Opcional	`FILTER_EXPRESS`	[expresión]	Una *expresión PDAL* para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos
Extensión del cultivo Opcional	`FILTER_EXTENT`	[extensión]	Una extensión del mapa para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cortadas	`OUTPUT`	[nube de puntos]	Nube de puntos de salida cuyos objetos espaciales son los puntos dentro de la capa del polígono de cobertura.

Código Python

ID Algoritmo: `pdal:clip`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear COPC

Crea el índice de todos los archivos de nubes de puntos de entrada por lotes.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`LAYERS`	[point cloud] [list]	Capas de nubes de puntos de entrada para crear un índice para
Directorio de salida Opcional	`OUTPUT`	[carpeta] Preestablecido: `[Omitir salida]`	Especifique la carpeta en la que se crearán los nuevos archivos. *Una de*: • Saltar salida • Guardar en Directorio Temporal • Guardar en Directorio

Salidas

Etiqueta	Nombre	Tipo	Descripción
Directorio de salida	`OUTPUT`	[carpeta]	Carpeta de salida que contiene las capas de nubes de puntos con los archivos de índice COPC correspondientes.

Código Python

ID Algoritmo: `pdal:createcopc`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Información

Genera metadatos básicos a partir de un archivo de nube de puntos de entrada.

Ejemplo de información de salida:

```
LAS           1.4
point format  6
count         56736130
scale         0.001 0.001 0.001
offset        431749.999 5440919.999 968.898
extent        431250 5440420 424.266
              432249.999 5441419.999 1513.531
crs           ETRS89 / UTM zone 34N (N-E) (EPSG:3046)  (vertical CRS missing!)
units         horizontal=metre  vertical=unknown

Attributes:
- X floating 8
- Y floating 8
- Z floating 8
- Intensity unsigned 2
- ReturnNumber unsigned 1
- NumberOfReturns unsigned 1
- ScanDirectionFlag unsigned 1
- EdgeOfFlightLine unsigned 1
- Classification unsigned 1
- ScanAngleRank floating 4
- UserData unsigned 1
- PointSourceId unsigned 2
- GpsTime floating 8
- ScanChannel unsigned 1
- ClassFlags unsigned 1
```

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de la nube de puntos de entrada de la que extraer información de metadatos
Información de la capa	`OUTPUT`	[archivo] Predeterinado: `[Save to temporary file]`	Especifique el archivo en el que se almacenará la información de metadatos. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Información de la capa	`OUTPUT`	[html]	`HTML` archivo para almacenar la información de metadatos.

Código Python

ID Algoritmo: `pdal:info`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Mezclar

Fusiona varios archivos de nubes de puntos en uno solo.

Ver también

Crear una nube de puntos virtual (VPC)

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`LAYERS`	[point cloud] [list]	Capas de nubes de puntos de entrada para fusionarlas en una sola
Mezclado	`OUTPUT`	[nube de puntos] Predeterinado: `[Save to temporary file]`	Especifique los archivos de entrada de fusión de nubes de puntos de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Expresión de filtrado Opcional	`FILTER_EXPRESS`	[expresión]	Una *expresión PDAL* para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos
Extensión del cultivo Opcional	`FILTER_EXTENT`	[extensión]	Una extensión del mapa para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`

Salidas

Etiqueta	Nombre	Tipo	Descripción
Mezclado	`OUTPUT`	[nube de puntos]	Capa de nube de puntos de salida que fusiona todos los archivos de entrada.

Código Python

ID Algoritmo: `pdal:merge`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Reproyectar

Reproyecta una nube de puntos a un Sistema de Referencia de Coordenadas (SRC) diferente.

Ver también

Asignar proyección

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de nube de puntos de entrada para reproyectar a un SRC diferente
SRC destino	`CRS`	[src]	El SRC que se aplicará a la capa
Reproyectado	`OUTPUT`	[nube de puntos] Predeterinado: `[Save to temporary file]`	Especifica el archivo de nube de puntos reproyectado. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Operación de coordenadas Opcional	`OPERATION`	[datum]	La *transformación de datum* que se utilizará para reproyectar los datos entre los sistemas de origen y destino.

Salidas

Etiqueta	Nombre	Tipo	Descripción
REPROJECTED	`OUTPUT`	[nube de puntos]	Capa de nube de puntos de salida en el SRC de destino.

Código Python

ID Algoritmo: `pdal:reproject`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Delgado (por radio de muestreo)

Crea una versión adelgazada de la nube de puntos realizando un muestreo por punto de distancia (reduce el número de puntos dentro de un radio determinado).

Ver también

Alivianar (saltándose puntos)

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de nube de puntos de entrada para crear una versión diluida a partir de
**Radio de muestreo (en unidades cartográficas)	`SAMPLING_RADIU`	[numérico: doble] Predeterminado: 1.0	Distancia dentro de la cual se muestrean los puntos a un punto único
Reducido (por radio)	`OUTPUT`	[nube de puntos] Predeterinado: `[Save to temporary file]`	Especifica la nube de puntos de salida con puntos reducidos. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Expresión de filtrado Opcional	`FILTER_EXPRESS`	[expresión]	Una *expresión PDAL* para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos

continúe en la próxima página

Tabla 24.285 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extensión del cultivo Opcional	`FILTER_EXTENT`	[extensión]	Una extensión del mapa para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`

Salidas

Etiqueta	Nombre	Tipo	Descripción
Reducido (por radio)	`OUTPUT`	[nube de puntos]	Capa de nube de puntos de salida con puntos reducidos.

Código Python

ID Algoritmo: `pdal:thinbyradius`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Alivianar (saltándose puntos)

Crea una versión reducida de la nube de puntos manteniendo sólo cada N-ésimo punto (reduce el número de puntos omitiendo los puntos cercanos).

Ver también

Delgado (por radio de muestreo)

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de nube de puntos de entrada para crear una versión diluida a partir de
Número de puntos a saltar	`POINTS_NUMBER`	[numérico: entero] Predeterminado: 1	Mantener sólo cada N-ésimo punto en la capa de entrada
Reducido (por diezmación)	`OUTPUT`	[nube de puntos] Predeterinado: `[Save to temporary file]`	Especifica la nube de puntos de salida con puntos reducidos. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Expresión de filtrado Opcional	`FILTER_EXPRESS`	[expresión]	Una *expresión PDAL* para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos

continúe en la próxima página

Tabla 24.288 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extensión del cultivo Opcional	`FILTER_EXTENT`	[extensión]	Una extensión del mapa para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`

Salidas

Etiqueta	Nombre	Tipo	Descripción
Reducido (por diezmación)	`OUTPUT`	[nube de puntos]	Capa de nube de puntos de salida con puntos reducidos.

Código Python

ID Algoritmo: `pdal:thinbydecimate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Tesela

Crea teselas a partir de archivos de nubes de puntos de entrada, recomendadas para obtener el mejor rendimiento (en visualización o análisis) con este tipo de conjuntos de datos en QGIS.

Ver también

Crear una nube de puntos virtual (VPC), *Crear COPC*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`LAYERS`	[point cloud] [list]	Capas de nubes de puntos de entrada para crear teselas a partir de ellas
Longitud de tesela	`LENGTH`	[numérico: doble] Predeterminado: 1000.0	Tamaño del borde de cada tesela generada
Directorio de salida	`OUTPUT`	[carpeta] Predeterminado: `[Guardar en directorio temporal]`	Especifique la carpeta en la que se almacenarán las teselas generadas. *Una de*: • Guardar en Directorio Temporal • Guardar en Directorio

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Asignar SRC Opcional	`CRS`	[src]	El SRC que se aplicará a la capa

Salidas

Etiqueta	Nombre	Tipo	Descripción
Directorio de salida	`OUTPUT`	[carpeta]	Carpeta de salida que contiene las teselas generadas a partir de los archivos de entrada.

Código Python

ID Algoritmo: `pdal:tile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.17 Extracción de nubes de puntos

Atención

Running these algorithms requires QGIS installed with PDAL >= 2.5.0 (see *Help* ► *About* menu).

Contorno

Exporta un archivo poligonal que contiene el límite de capa de la nube de puntos. Puede contener agujeros y puede ser un polígono de varias partes.

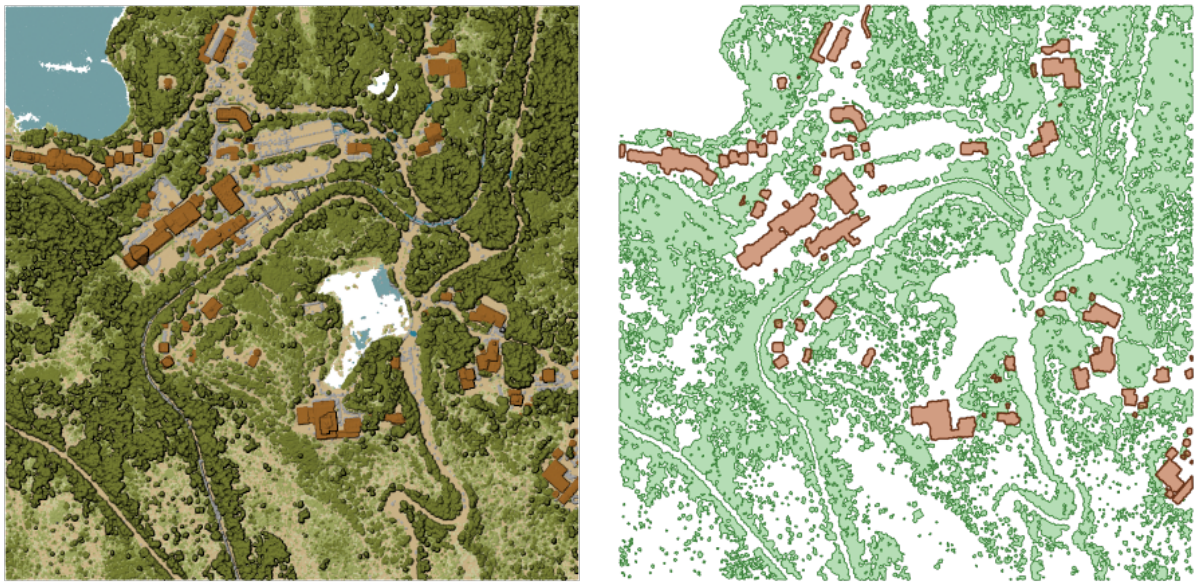


Figura 24.46: Extracción de vegetación alta y polígonos de edificios a partir de una capa de nubes de puntos de entrada

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de la nube de puntos de entrada para calcular el límite de
Resolución de las celdas utilizadas para calcular los límites Opcional	`RESOLUTION`	[numérico: doble]	Resolución de las celdas utilizadas para calcular los límites
Número mínimo de puntos en una celda para considerar que la celda está ocupada Opcional	`THRESHOLD`	[numérico: entero]	Número mínimo de puntos en una celda para considerarla ocupada
Perímetro	`OUTPUT`	[vectorial: poligonal] Predeterinado: `[Save to temporary file]`	Especifique el archivo de nube de puntos que se utilizará como salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Expresión de filtrado Opcional	`FILTER_EXPRESS`	[expresión]	Una *expresión PDAL* para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos
Extensión del cultivo Opcional	`FILTER_EXTENT`	[extensión]	Una extensión del mapa para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`

Salidas

Etiqueta	Nombre	Tipo	Descripción
Perímetro	`OUTPUT`	[vectorial: poligonal]	Capa vectorial poligonal de salida con el límite de la nube de puntos. El formato actualmente soportado es `.GPKG`.

Código Python

ID de algorithm: `pdal:boundary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Densidad

Exporta un archivo ráster basado en el número de puntos de cada celda ráster: útil para comprobar la calidad de los conjuntos de datos de nubes de puntos.

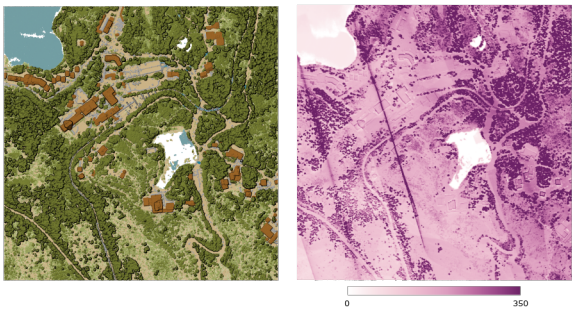


Figura 24.47: Densidad de puntos (número de puntos por 2x2 m) como trama

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de nube de puntos de entrada para exportar
Resolución del ráster de densidad	`RESOLUTION`	[numérico: doble] Predeterminado: 1.0	Tamaño de celda del raster de salida
Tamaño de las baldosas para recorridos paralelos	`TILE_SIZE`	[numérico: entero] Preestablecido: 1000	Tamaño de los mosaicos en los que se dividirán los datos para las ejecuciones paralelas

continúe en la próxima página

Tabla 24.296 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Densidad	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique el archivo raster al que exportar los datos. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Expresión de filtrado Opcional	`FILTER_EXPRESS`	[expresión]	Una *expresión PDAL* para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos
Extensión del cultivo Opcional	`FILTER_EXTENT`	[extensión]	Una extensión del mapa para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Origen X de una tesela para ejecuciones paralelas Opcional	`ORIGIN_X`	[numérico: doble]	
Origen Y de una tesela para ejecuciones paralelas Opcional	`ORIGIN_Y`	[numérico: doble]	

Salidas

Etiqueta	Nombre	Tipo	Descripción
Densidad	`OUTPUT`	[ráster]	Capa ráster de salida con el número de puntos dentro de cada celda. El formato actualmente soportado es `.TIF`.

Código Python

Identificación del algoritmo: `pdal:density`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Filtro

Extrae puntos de la nube de puntos de entrada que coinciden con la expresión PDAL y/o están dentro de un rectángulo de recorte.

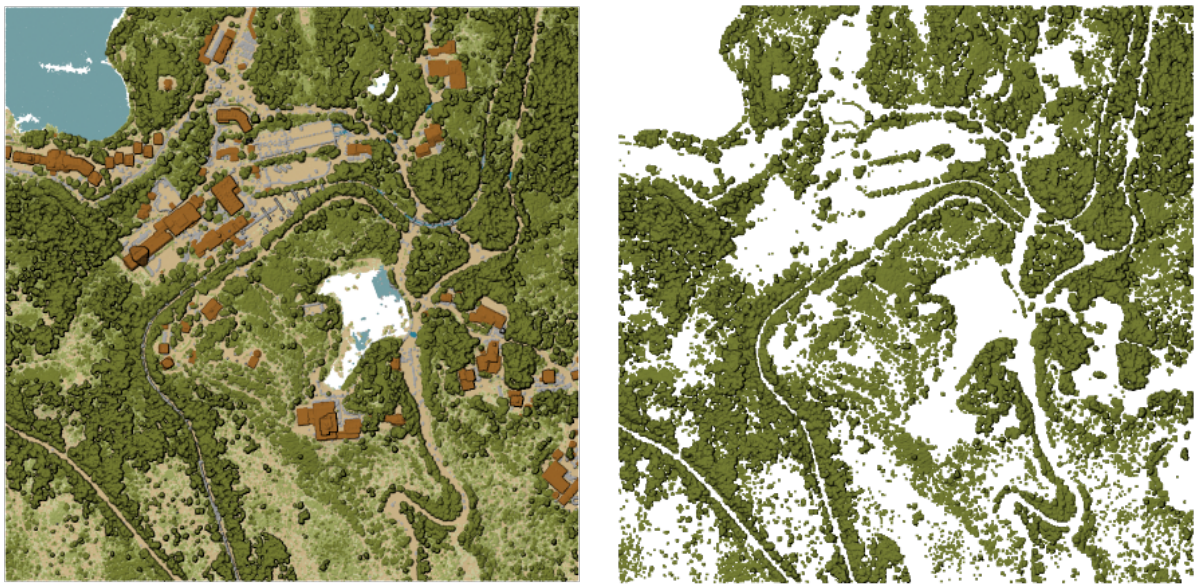


Figura 24.48: Filtrado de la clase de vegetación alta a partir de una capa de nubes de puntos de entrada

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[nube de puntos]	Capa de nube de puntos de entrada para exportar
Expresión de filtrado Opcional	`FILTER_EXPRESS`	[expresión]	Una *expresión PDAL* para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos
Extensión del cultivo Opcional	`FILTER_EXTENT`	[extensión]	Una extensión del mapa para seleccionar un subconjunto de objetos espaciales en los datos de la nube de puntos Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Filtrado	`OUTPUT`	[nube de puntos] Predeterinado: `[Save to temporary file]`	Especifique el archivo de nube de puntos al que exportar los datos. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Filtrado	`OUTPUT`	[nube de puntos]	Capa de nube de puntos de salida con los objetos espaciales filtrados de la capa de nube de puntos de entrada. Los formatos actualmente soportados son `.LAS`, `.LAZ`, `.COPC.LAZ` y `.VPC`.

Código Python

Identificación del algoritmo: `pdal:filter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.18 Análisis raster

Clasificación porcentual de la pila de celdas a partir del valor

Calcula el valor de rango porcentual por celdas de una pila de rásters en función de un único valor de entrada y los escribe en un ráster de salida.

En cada ubicación de celda, el valor especificado se clasifica entre los valores respectivos en la pila de todos los valores de celda superpuestos y ordenados de los rásteres de entrada. Para valores fuera de la distribución de valores de pila, el algoritmo devuelve NoData porque el valor no se puede clasificar entre los valores de celda.

Hay dos métodos para el cálculo de percentiles:

- Interpolación lineal inclusiva (PERCENTRANK.INC)
- Interpolación lineal exclusiva (PERCENTRANK.EXC)

El método de interpolación lineal devuelve el rango porcentual único para diferentes valores. Ambos métodos de interpolación siguen sus métodos equivalentes implementados por LibreOffice or Microsoft Excel.

La extensión y la resolución del ráster de salida se definen mediante un ráster de referencia. Las capas ráster de entrada que no coincidan con el tamaño de celda de la capa ráster de referencia se volverán a muestrear mediante el remuestreo del vecino más cercano. Los valores NoData en cualquiera de las capas de entrada darán como resultado una salida de celda NoData si el parámetro «Ignore NoData values» no está configurado. El tipo de datos ráster de salida siempre será `Float32`.

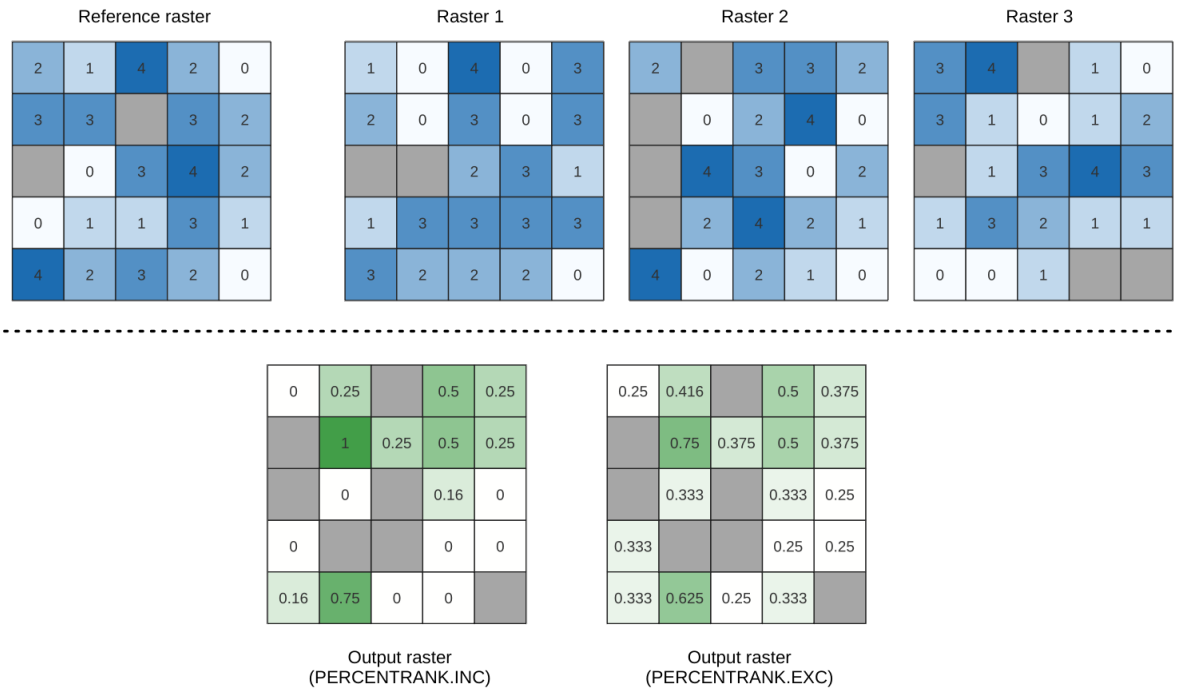


Figura 24.49: Clasificación porcentual Valor = 1. Las celdas `NoData` (grises) se ignoran.

Ver también

percentil de pila de celdas, *Porcentaje de la pila de celdas a partir de la capa ráster*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Capas ráster para evaluar. Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres.
Methodo	`METHOD`	[enumeración] Predeterminado: 0	Método para el cálculo del percentil: • 0 — Interpolación lineal inclusiva (PERCENTRANK.INC) • 1 — Interpolación lineal exclusiva (PERCENTRANK.EXC)
Valor	`VALUE`	[numérico: doble] Predeterminado: 10.0	Valor para clasificar entre los valores respectivos en la pila de todos los valores de celda superpuestos y ordenados de los rásters de entrada
Ignorar valores NoData	`IGNORE_NODATA`	[booleano] Predeterminado: Verdadero	Si no está marcada, cualquier celda NoData en las capas de entrada dará como resultado una celda NoData en el ráster de salida

continúe en la próxima página

Tabla 24.301 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Capa de referencia	`REFERENCE_LAYE`	[ráster]	La capa de referencia para la creación de la capa de salida (extensión, SRC, dimensiones en píxeles)
Capa de salida	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData	`OUTPUT_NODATA_`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

Algoritmo ID: `native:cellstackpercentrankfromvalue`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

percentil de pila de celdas

Calcula el valor del percentil por celdas de una pila de rásters y escribe los resultados en un ráster de salida. El percentil a devolver está determinado por el valor de entrada del percentil (intervalos entre 0 y 1). En cada ubicación de celda, el percentil especificado se obtiene utilizando el valor respectivo de la pila de todos los valores de celda superpuestos y ordenados de los rásters de entrada.

Hay tres métodos para el cálculo de percentiles:

- Rango más cercano: devuelve el valor más cercano al percentil especificado
- Interpolación lineal inclusiva (PERCENTRANK.INC)
- Interpolación lineal exclusiva (PERCENTRANK.EXC)

Los métodos de interpolación lineal devuelven valores únicos para diferentes percentiles. Ambos métodos de interpolación siguen sus métodos homólogos implementados por LibreOffice or Microsoft Excel.

La extensión y la resolución del ráster de salida se definen mediante un ráster de referencia. Las capas ráster de entrada que no coincidan con el tamaño de celda de la capa ráster de referencia se volverán a muestrear mediante el remuestreo del vecino más cercano. Los valores NoData en cualquiera de las capas de entrada darán como resultado una salida de celda NoData si el parámetro «Ignore NoData values» no está configurado. El tipo de datos ráster de salida siempre será `Float32`.

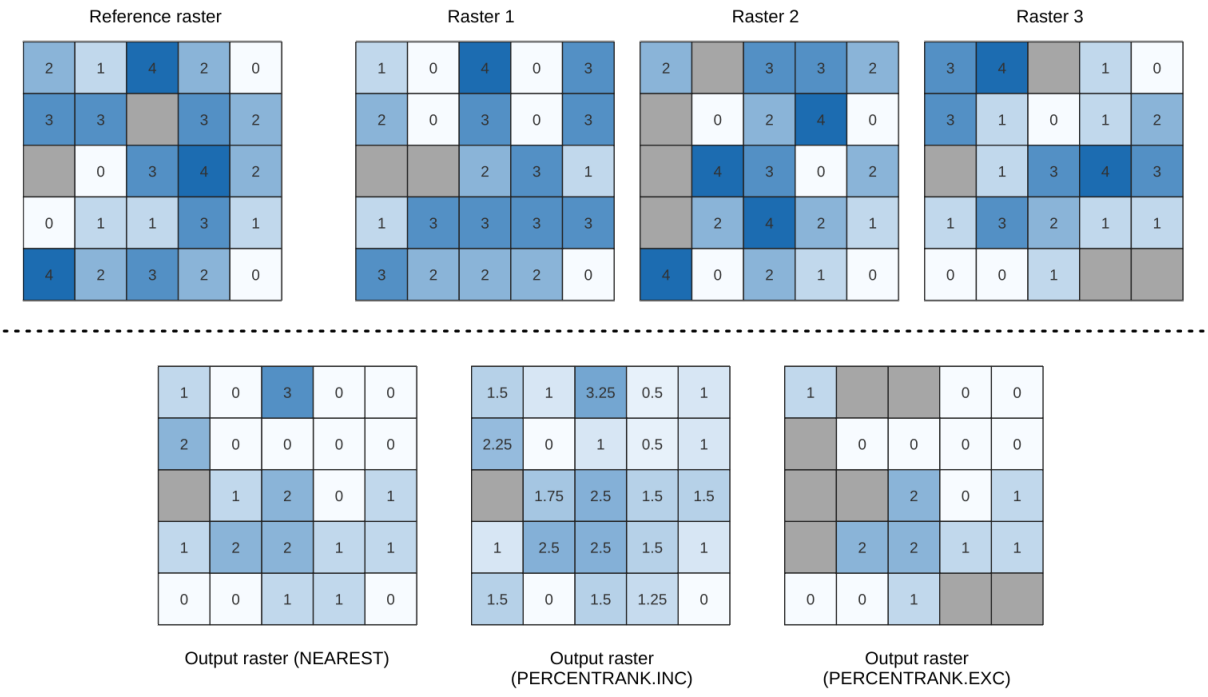


Figura 24.50: Percentil = 0,25. Las celdas `NoData` (grises) se ignoran.

Ver también

percentil de pila de celdas, *Porcentaje de la pila de celdas a partir de la capa ráster*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Capas ráster para evaluar. Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres.
Methodo	`METHOD`	[enumeración] Predeterminado: 0	Método para el cálculo del percentil: • 0 — Nearest rank: devuelve el valor más cercano al percentil especificado • 1 — Interpolación lineal inclusiva (PERCENTILE.INC) • 2 — Interpolación lineal exclusiva (PERCENTILE.EXC)
Percentil	`VALUE`	[numérico: doble] Preestablecido: 0.25	Valor para clasificar entre los valores respectivos en la pila de todos los valores de celda superpuestos y ordenados de los rásteres de entrada. Entre 0 y 1.
Ignorar valores NoData	`IGNORE_NODATA`	[booleano] Predeterminado: Verdadero	Si no está marcada, cualquier celda NoData en las capas de entrada dará como resultado una celda NoData en el ráster de salida
Capa de referencia	`REFERENCE_LAYE`	[ráster]	La capa de referencia para la creación de la capa de salida (extensión, SRC, dimensiones en píxeles)
Capa de salida	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData	`OUTPUT_NODATA_`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida

continúe en la próxima página

Tabla 24.305 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

Algoritmo ID: `native:cellstackpercentile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Porcentaje de la pila de celdas a partir de la capa ráster

Calcula el valor de rango porcentual por celdas de una pila de rásters en función de un ráster de valores de entrada y los escribe en un ráster de salida.

En cada ubicación de celda, el valor actual del ráster de valores se clasifica entre los valores respectivos en la pila de todos los valores de celda superpuestos y ordenados de los rásters de entrada. Para valores fuera de la distribución de valores de la pila, el algoritmo devuelve NoData porque el valor no se puede clasificar entre los valores de celda.

Hay dos métodos para el cálculo de percentiles:

- Interpolación lineal inclusiva (PERCENTRANK.INC)
- Interpolación lineal exclusiva (PERCENTRANK.EXC)

Los métodos de interpolación lineal devuelven valores únicos para diferentes percentiles. Ambos métodos de interpolación siguen sus métodos homólogos implementados por LibreOffice or Microsoft Excel.

La extensión y la resolución del ráster de salida se definen mediante un ráster de referencia. Las capas ráster de entrada que no coincidan con el tamaño de celda de la capa ráster de referencia se volverán a muestrear mediante el remuestreo del vecino más cercano. Los valores NoData en cualquiera de las capas de entrada darán como resultado una salida de celda NoData si el parámetro «Ignore NoData values» no está configurado. El tipo de datos ráster de salida siempre será `Float32`.

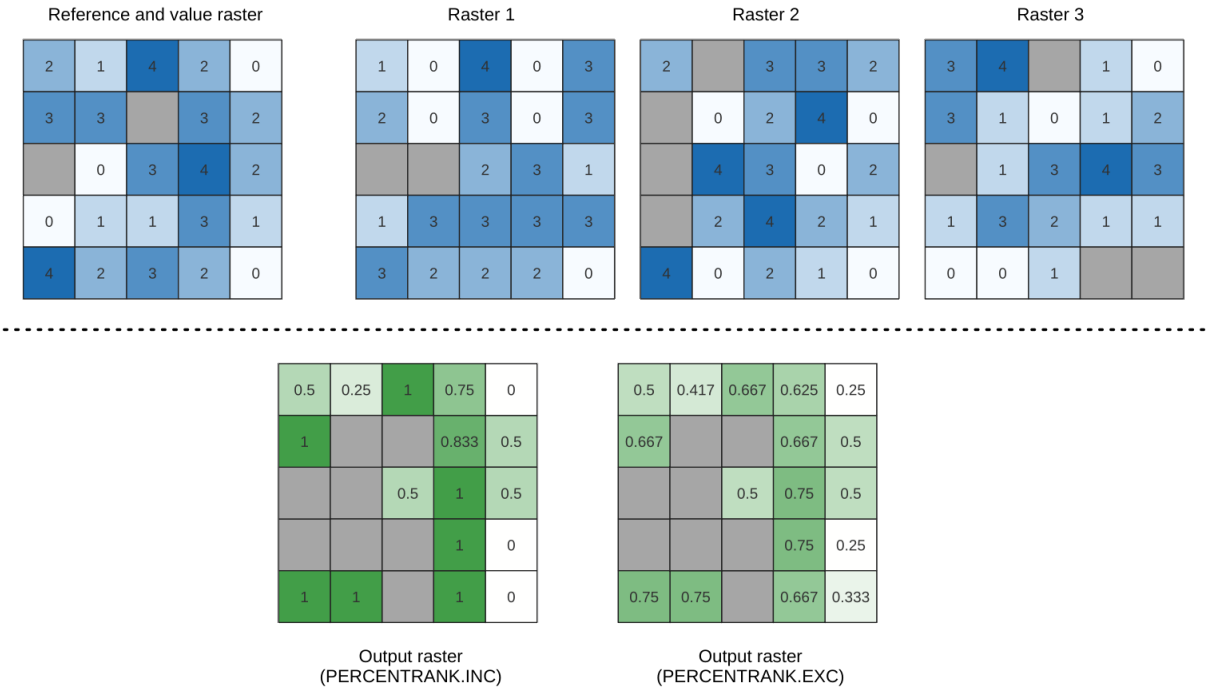


Figura 24.51: Clasificación de las celdas de la capa ráster de valor. Las celdas `NoData` (grises) se ignoran.

Ver también

percentil de pila de celdas, *Clasificación porcentual de la pila de celdas a partir del valor*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Capas ráster para evaluar. Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres.
Capa ráster de valor	`INPUT_VALUE_RA`	[ráster]	La capa para clasificar los valores entre la pila de todas las capas superpuestas
Banda ráster de valor	`VALUE_RASTER_B`	[numérico: entero] Predeterminado: 1	Banda de la «value raster layer» para comparar

continúe en la próxima página

Tabla 24.307 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Methodo	`METHOD`	[enumeración] Predeterminado: 0	Método para el cálculo del percentil: • 0 — Interpolación lineal inclusiva (PERCENTRANK.INC) • 1 — Interpolación lineal exclusiva (PERCENTRANK.EXC)
Ignorar valores NoData	`IGNORE_NODATA`	[booleano] Predeterminado: Verdadero	Si no está marcada, cualquier celda NoData en las capas de entrada dará como resultado una celda NoData en el ráster de salida
Capa de referencia	`REFERENCE_LAYE`	[ráster]	La capa de referencia para la creación de la capa de salida (extensión, SRC, dimensiones en píxeles)
Capa de salida	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData	`OUTPUT_NODATA_`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida

continúe en la próxima página

Tabla 24.309 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

Algoritmo ID: `native:cellstackpercentrankfromrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Estadísticas de celda

Calcula estadísticas por celda basadas en capas ráster de entrada y para cada celda escribe las estadísticas resultantes en un ráster de salida. En cada ubicación de celda, el valor de salida se define como una función de todos los valores de celda superpuestos de los rásteres de entrada.

De forma predeterminada, una celda NoData en CUALQUIERA de las capas de entrada dará como resultado una celda NoData en el ráster de salida. Si la opción *Ignorar valores NoData* está marcada, las entradas NoData se ignorarán en el cálculo estadístico. Esto puede resultar en una salida NoData para ubicaciones donde todas las celdas son NoData.

El parámetro *Capa de referencia* especifica una capa ráster existente para usar como referencia al crear el ráster de salida. El ráster de salida tendrá la misma extensión, SRC y dimensiones de píxeles que esta capa.

Detalles del cálculo: Las capas ráster de entrada que no coincidan con el tamaño de celda de la capa ráster de referencia se volverán a muestrear utilizando el `remuestreo del vecino más cercano`. El tipo de datos ráster de salida se establecerá en el tipo de datos más complejo presente en los conjuntos de datos de entrada, excepto cuando se utilizan las funciones `Media`, `Desviación estándar` y `Varianza` (el tipo de datos siempre es `Float32` o `Float64` dependiendo del tipo de flotador de entrada) o `Count` y `Variety` (el tipo de datos es siempre `Int32`).

- `Recuento`: la estadística de recuento siempre dará como resultado el número de celdas sin valores NoData en la ubicación de la celda actual.
- `Mediana`: si el número de capas de entrada es par, la mediana se calculará como la media aritmética de los dos valores medios de los valores de entrada de la celda ordenada.
- `Minoría/Mayoría`: Si no se pudo encontrar una minoría o mayoría única, el resultado es NoData, excepto que todos los valores de celda de entrada son iguales.

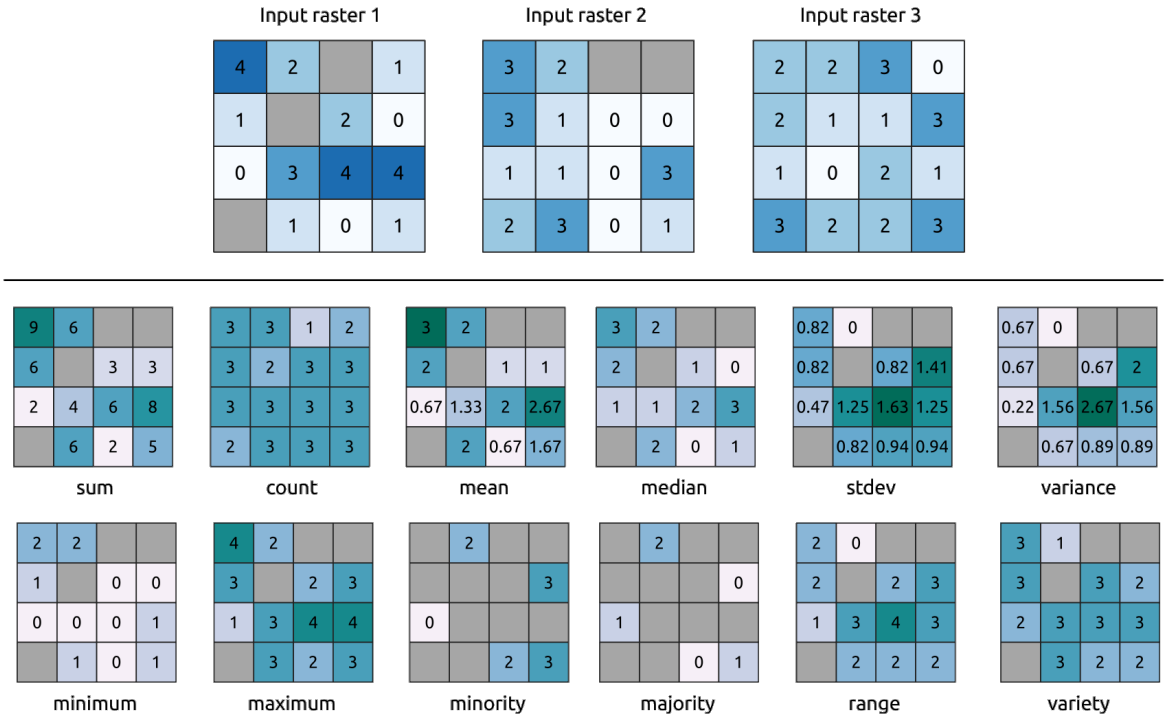


Figura 24.52: Ejemplo con todas las funciones estadísticas. Se tienen en cuenta las celdas `NoData` (grises).

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Capas ráster entrantes
Estadística	`STATISTIC`	[enumeración] Predeterminado: 0	Estadisticas disponibles. Opciones: • 0 — Suma • 1 — Recuento • 2 — Media • 3 — Mediana • 4 — Desviación Estándar • 5 — Varianza • 6 — Mínimo • 7 — Máximo • 8 — Minoría (valor menos común) • 9 — Mayoría (valor mas común) • 10 — Rango (máx - min) • 11 — Variedad (recuento de valores únicos)
Ignorar valores NoData	`IGNORE_NODATA`	[booleano] Predeterminado: Verdadero	Calcular estadísticas también para todas las pilas de celdas, ignorando la existencia de NoData.
Capa de referencia	`REF_LAYER`	[ráster]	La capa de referencia desde la que crear la capa de salida (extensión, SRC, dimensiones en píxeles)

continúe en la próxima página

Tabla 24.310 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData Opcional	`OUTPUT_NO_DATA`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Identificador de autoridad de SRC	`CRS_AUTHID`	[src]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Ráster de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida

Código Python

Algoritmo ID: `native:cellstatistics`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Frecuencia de igualdad

Evalúa celda por celda la frecuencia (cantidad de veces) que los valores de una pila de entrada de rásteres son iguales al valor de una capa de valor. La extensión y resolución del ráster de salida están definidas por la capa ráster de entrada y siempre es del tipo `Int32`.

Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres; use GDAL para usar otras bandas en el análisis. El valor de salida sin datos se puede configurar manualmente.

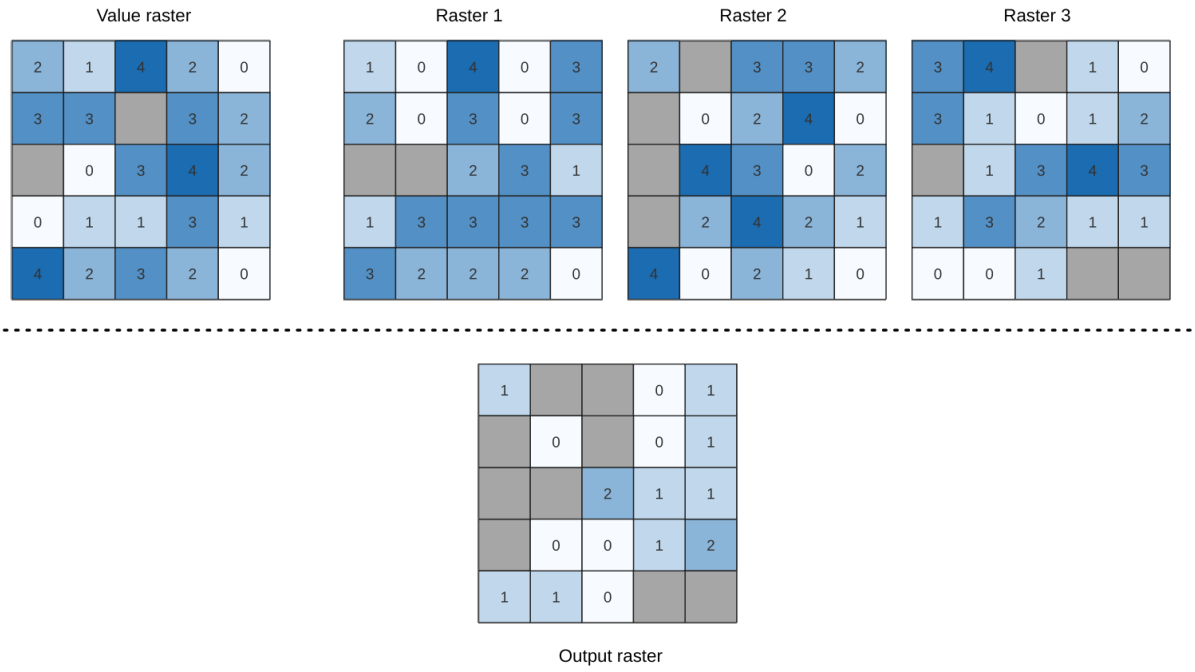


Figura 24.53: Para cada celda del ráster de salida, el valor representa la cantidad de veces que las celdas correspondientes en la lista de rásteres son iguales al ráster de valores. Se tienen en cuenta las celdas sin datos (grises).

Ver también

Frecuencia mayor que, *Frecuencia menor que*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Valor ráster de entrada	`INPUT_VALUE_RA`	[ráster]	La capa de valor de entrada sirve como capa de referencia para las capas de muestra
Banda ráster de valor	`INPUT_VALUE_RA`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Seleccione la banda que desea usar como muestra
Capas ráster entrante	`INPUT_RASTERS`	[ráster] [lista]	Capas ráster para evaluar. Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres.
Ignorar valores NoData	`IGNORE_NODATA`	[booleano] Predeterminado: Falso	Si no está marcada, cualquier celda sin datos en el ráster de valor o la pila de capas de datos dará como resultado una celda sin datos en el ráster de salida
Capa de salida	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData Opcional	`OUTPUT_NO_DATA`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Conteo de celdas con ocurrencias de valor igual	`FOUND_LOCATION`	[numérico: entero]	
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente
Frecuencia media en ubicaciones de celda válidas	`MEAN_FREQUENCY`	[numérico: doble]	
Conteo de ocurrencias del valor	`OCCURRENCE_COU`	[numérico: entero]	
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida

Código Python

Algoritmo ID: `native:equaltofrequency`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ráster Fuzzify (pertenencia gaussiana)

Transforma un ráster de entrada en un ráster difuso asignando un valor de pertenencia a cada píxel, utilizando una función de pertenencia gaussiana. Los valores de pertenencia van de 0 a 1. En el ráster difuso, un valor de 0 implica que no hay pertenencia al conjunto difuso definido, mientras que un valor de 1 significa pertenencia total. La función de pertenencia gaussiana se define como $\mu(x) = e^{-f1*(x-f2)^2}$, donde *f1* es la extensión y *f2* el punto medio.

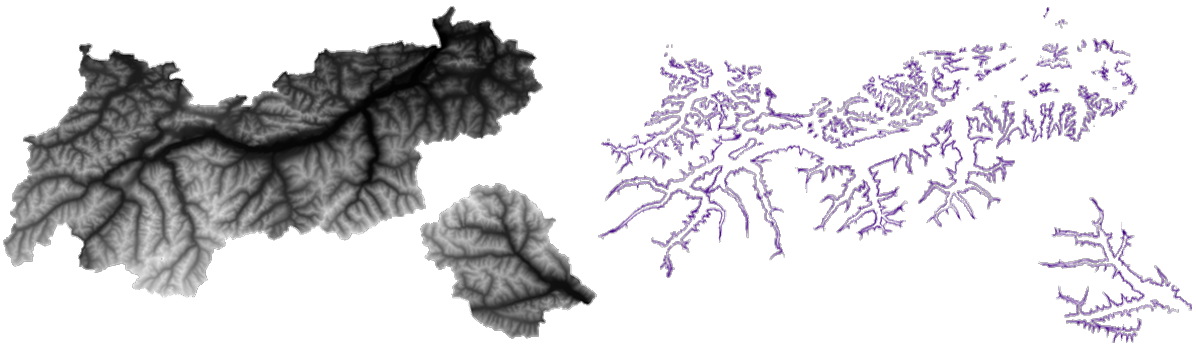


Figura 24.54: Ejemplo de ráster de Fuzzify. Fuente de ráster de entrada: Land Tirol - data.tirol.gv.at.

Ver también

Fuzzify ráster (gran número de miembros), *Ráster Fuzzify (membresía linear)*, *Fuzzify ráster (número de miembros cercanos)*, *Ráster Fuzzify(influencia de membresía)*, *Ràster Fuzzify (pequeña membresía)*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Ráster de Entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de Banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Si el ráster es multibanda, elija la banda a la que aplicar fuzzify.
Función Punto Medio	`FUZZYMIDPOINT`	[numérico: doble] Predeterminado: 10.0	Punto medio de la función gausiana
Distribución de funciones	`FUZZYSPREAD`	[numérico: doble] Predeterminado: 0.01	Distribución de la función gausiana
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[src]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

ID Algoritmo: `native:fuzzifyrastergaussianmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Fuzzify ráster (gran número de miembros)

Transforma un ráster de entrada en un ráster difuso asignando un valor de número de miembro a cada píxel, utilizando una función de membresía grande. Los valores de pertenencia van de 0 a 1. En el ráster difuso, un valor de 0 implica que no hay pertenencia al conjunto difuso definido, mientras que un valor de 1 significa pertenencia total. La función de gran número de miembros se define como $\mu(x) = \frac{1}{1+\left(\frac{x}{f2}\right)^{-f1}}$, donde $f1$ es la extensión y $f2$ el punto medio.

Ver también

Ráster Fuzzify (pertenencia gaussiana), *Ráster Fuzzify (membresía linear)*, *Fuzzify ráster (número de miembros cercanos)*, *Ráster Fuzzify(influencia de membresía)*, *Ràster Fuzzify (pequeña membresía)*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Ráster de Entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de Banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Si el ráster es multibanda, elija la banda a la que aplicar fuzzify.
Función Punto Medio	`FUZZYMIDPOINT`	[numérico: doble] Predeterminada: 50.0	Punto medio de una función larga
Distribución de funciones	`FUZZYSPREAD`	[numérico: doble] Preestablecido: 5.0	Distribución de una función larga
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[src]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

ID Algoritmo: `native:fuzzifyrasterlargemembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ráster Fuzzify (membresía linear)

Transforma un ráster de entrada en un ráster difuso asignando un valor de pertenencia a cada píxel, utilizando una función de pertenencia lineal. Los valores de pertenencia van de 0 a 1. En el ráster difuso, un valor de 0 implica que no hay pertenencia al conjunto difuso definido, mientras que un valor de 1 significa pertenencia total. La función lineal se define como $\mu(X)\begin{cases}0 & x \leq a \\ \frac{x-a}{b-a} & a < x < b \\ 1 & x \geq b\end{cases}$, donde a es el límite inferior y b el límite superior. Esta ecuación asigna valores de pertenencia mediante una transformación lineal para valores de píxeles entre los límites inferior y superior. Los valores de píxeles más pequeños que el límite bajo reciben una membresía 0, mientras que los valores de píxeles mayores que el límite alto reciben 1 membresía.

Ver también

Ráster Fuzzify (pertenencia gaussiana), *Fuzzify ráster (gran número de miembros)*, *Fuzzify ráster (número de miembros cercanos)*, *Ráster Fuzzify(influencia de membresía)*, *Ràster Fuzzify (pequeña membresía)*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Ráster de Entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de Banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Si el ráster es multibanda, elija la banda a la que aplicar fuzzify.
Límite de membresía difusa baja	`FUZZYLOWBOUND`	[numérico: doble] Predeterminado: 0.0	Límite inferior de la función lineal
Límite alto de membresía difusa	`FUZZYHIGHBOUND`	[numérico: doble] Predeterminado: 1.0	Límite Superior de la función lineal
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[src]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida

continúe en la próxima página

Tabla 24.324 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

Algoritmo ID: `native:fuzzifyrasterlinearmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Fuzzify ráster (número de miembros cercanos)

Transforma un ráster de entrada en un ráster difuso asignando un valor de pertenencia a cada píxel, utilizando una función de pertenencia cercana. Los valores de pertenencia van de 0 a 1. En el ráster difuso, un valor de 0 implica que no hay pertenencia al conjunto difuso definido, mientras que un valor de 1 significa pertenencia total. La función de membresía cercana se define como $\mu(x) = \frac{1}{1 + f1 * (x - f2)^2}$, donde $f1$ es la extensión y $f2$ el punto medio.

Ver también

Ráster Fuzzify (pertenencia gaussiana), *Fuzzify ráster (gran número de miembros)*, *Ráster Fuzzify (membresía linear)*, *Ráster Fuzzify(influencia de membresía)*, *Ràster Fuzzify (pequeña membresía)*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Ráster de Entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de Banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Si el ráster es multibanda, elija la banda a la que aplicar fuzzify.
Función Punto Medio	`FUZZYMIDPOINT`	[numérico: doble] Predeterminada: 50.0	Punto medio de la función vecino
Distribución de funciones	`FUZZYSPREAD`	[numérico: doble] Predeterminado: 0.01	Extensión de la función vecino

continúe en la próxima página

Tabla 24.325 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[src]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

Algoritmo ID: `native:fuzzifyrasternearmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ráster Fuzzify(influencia de membresía)

Transforma un ráster de entrada en un ráster difuso asignando un valor de membresía a cada píxel, utilizando una función de membresía de influencia. Los valores de pertenencia van de 0 a 1. En el ráster difuso, un valor de 0 implica que no hay pertenencia al conjunto difuso definido, mientras que un valor de 1 significa pertenencia total. La función de potencia se define como $\mu(x)\begin{cases}0 & x \le a\\ \left(\frac{x-a}{b-a}\right)^{f1} & a < x < b\\ 1 & x \ge b\end{cases}$, donde a es el límite inferior, b es el límite superior y $f1$ el exponente. Esta ecuación asigna valores de pertenencia utilizando la transformación de potencia para los valores de píxeles entre los límites inferior y superior. Los valores de píxeles más pequeños que el límite bajo reciben una membresía 0, mientras que los valores de píxeles mayores que el límite alto reciben 1 membresía.

Ver también

Ráster Fuzzify (pertenencia gaussiana), *Fuzzify ráster (gran número de miembros)*, *Ráster Fuzzify (membresía linear)*, *Fuzzify ráster (número de miembros cercanos)*, *Ràster Fuzzify (pequeña membresía)*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Ráster de Entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de Banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Si el ráster es multibanda, elija la banda a la que aplicar fuzzify.
Límite de membresía difusa baja	`FUZZYLOWBOUND`	[numérico: doble] Predeterminado: 0.0	Límite inferior de la función de influencia
Límite alto de membresía difusa	`FUZZYHIGHBOUND`	[numérico: doble] Predeterminado: 1.0	Límite superior de la función de influencia
Límite alto de membresía difusa	`FUZZYEXPONENT`	[numérico: doble] Preestablecido: 2.0	Exponente de la función de influencia
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[src]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

Algoritmo ID: `native:fuzzifyrasterpowermembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ràster Fuzzify (pequeña membresía)

Transforma un ráster de entrada en un ráster difuso asignando un valor de pertenencia a cada píxel, utilizando una función de pertenencia pequeña. Los valores de pertenencia van de 0 a 1. En el ráster difuso, un valor de 0 implica que no hay pertenencia al conjunto difuso definido, mientras que un valor de 1 significa pertenencia total. La función de pertenencia pequeña se define como |**fórmula_pequeña**|, donde *f1* es el margen y *f2* el punto medio.

Ver también

Ráster Fuzzify (pertenencia gaussiana), *Fuzzify ráster (gran número de miembros) Ráster Fuzzify (membresía linear)*, *Fuzzify ráster (número de miembros cercanos)*, *Ráster Fuzzify(influencia de membresía)*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Ráster de Entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de Banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Si el ráster es multibanda, elija la banda a la que aplicar fuzzify.
Función Punto Medio	`FUZZYMIDPOINT`	[numérico: doble] Predeterminada: 50.0	Punto medio de la función pequeña
Distribución de funciones	`FUZZYSPREAD`	[numérico: doble] Preestablecido: 5.0	Extensión de la función pequeña
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster Fuzzified	`OUTPUT`	[la misma que la entrada]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[src]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente

continúe en la próxima página

Tabla 24.333 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

Algoritmo ID: `native:fuzzifyrastersmallmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Frecuencia mayor que

Evalúa celda por celda la frecuencia (cantidad de veces) que los valores de una pila de rásteres de entrada son iguales al valor de un ráster de valores. La extensión y resolución del ráster de salida está definida por la capa ráster de entrada y siempre es del tipo `Int32`.

Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres; use GDAL para usar otras bandas en el análisis. El valor de salida sin datos se puede configurar manualmente.

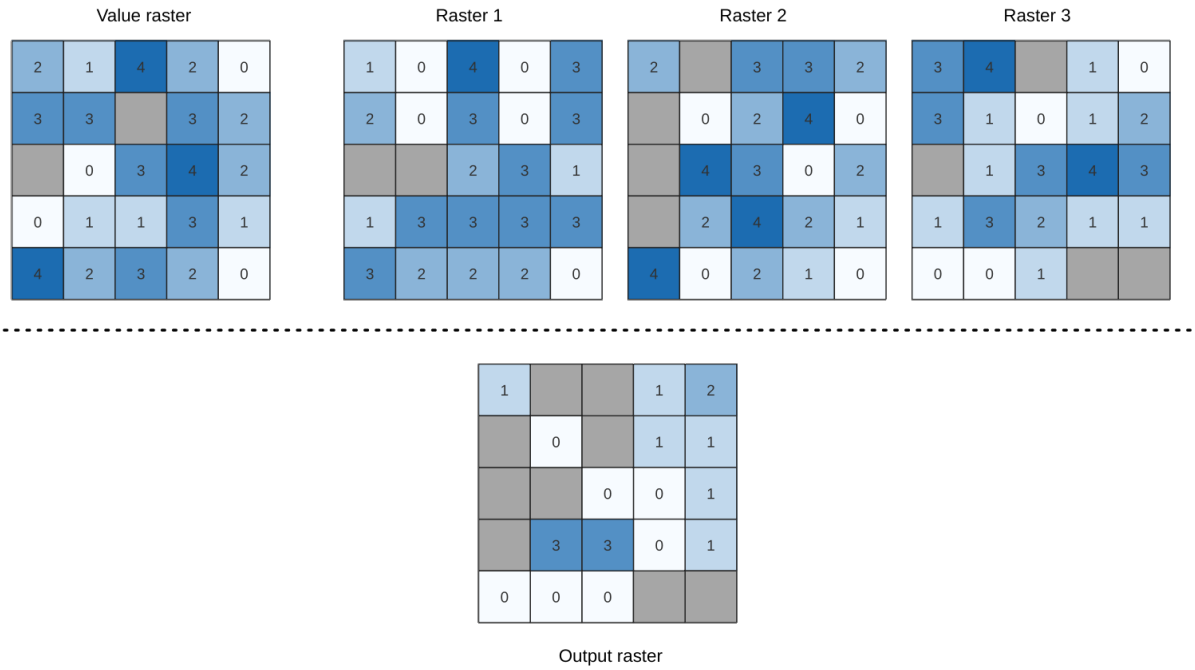


Figura 24.55: Para cada celda del ráster de salida, el valor representa la cantidad de veces que las celdas correspondientes en la lista de rásteres son mayores que el ráster de valor. Se tienen en cuenta las celdas `NoData` (grises).

Ver también

Frecuencia de igualdad, *Frecuencia menor que*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Valor ráster de entrada	`INPUT_VALUE_RA`	[ráster]	La capa de valor de entrada sirve como capa de referencia para las capas de muestra
Banda ráster de valor	`INPUT_VALUE_RA`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Seleccione la banda que desea usar como muestra
Capas ráster entrante	`INPUT_RASTERS`	[ráster] [lista]	Capas ráster para evaluar. Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres.
Ignorar valores NoData	`IGNORE_NODATA`	[booleano] Predeterminado: Falso	Si no está marcada, cualquier celda sin datos en el ráster de valor o la pila de capas de datos dará como resultado una celda sin datos en el ráster de salida
Capa de salida	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData Opcional	`OUTPUT_NO_DATA`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Conteo de celdas con ocurrencias de valor igual	`FOUND_LOCATION`	[numérico: entero]	
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente
Frecuencia media en ubicaciones de celda válidas	`MEAN_FREQUENCY`	[numérico: doble]	
Conteo de ocurrencias del valor	`OCCURRENCE_COU`	[numérico: entero]	
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida

Código Python

Algoritmo ID: `native:greaterthanfrequency`

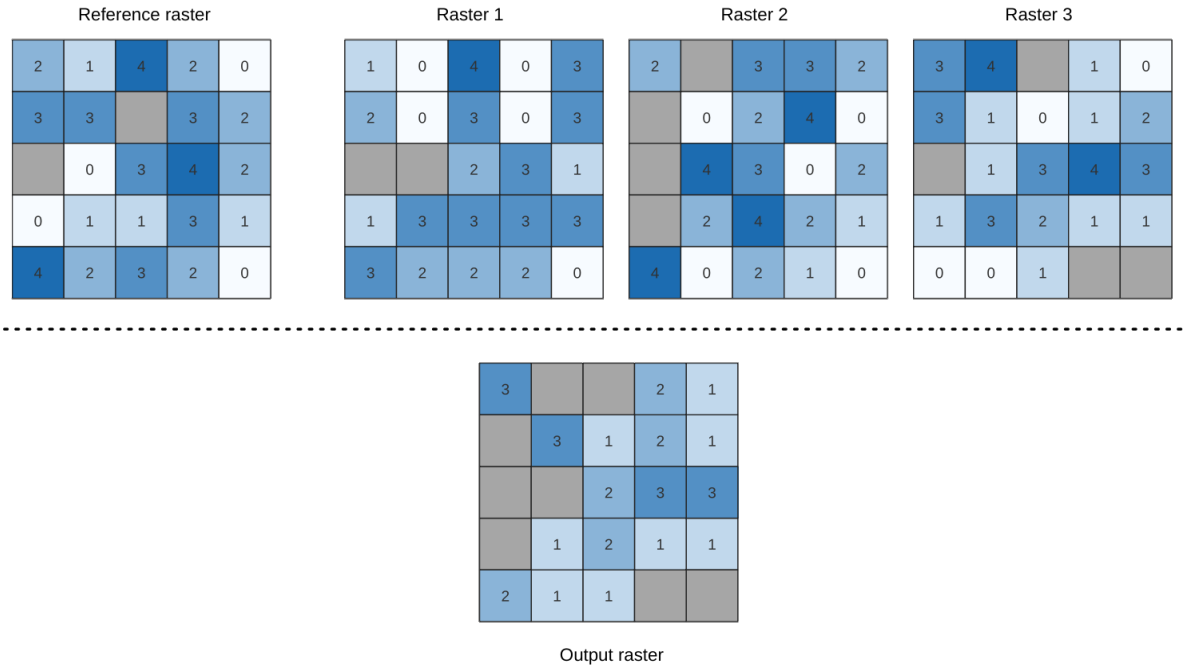
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Posición más alta en pila ráster

Evalúa celda por celda la posición del ráster con el valor más alto en una pila de rásteres. Los conteos de posición comienzan con 1 y van hasta el número total de rásteres de entrada. El orden de los rásteres de entrada es relevante para el algoritmo. Si múltiples rásteres presentan el valor más alto, el primer ráster se utilizará para el valor de posición.

Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres; use GDAL para usar otras bandas en el análisis. Cualquier celda sin datos en la pila de capas ráster dará como resultado una celda sin datos en el ráster de salida a menos que se marque el parámetro «ignore NoData». El valor de salida sin datos se puede configurar manualmente. La extensión y resolución de los rásteres de salida está definida por una capa ráster de referencia y siempre es del tipo `Int32`.



Ver también

Posición más baja en pila ráster

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas ráster entrante	`INPUT_RASTERS`	[ráster] [lista]	Lista de capas ráster para comparar
Capa de referencia	`REFERENCE_LAYE`	[ráster]	La capa de referencia para la creación de la capa de salida (extensión, SRC, dimensiones en píxeles)
Ignorar valores NoData	`IGNORE_NODATA`	[booleano] Predeterminado: Falso	Si no está marcada, cualquier celda sin datos en la pila de capas de datos dará como resultado una celda sin datos en el ráster de salida
Capa de salida	`OUTPUT`	[ráster] Predeterminado: `[Save to temporary file]`	Especificación de la trama de salida que contiene el resultado. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData	`OUTPUT_NODATA_`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

Algoritmo ID: `native:highestpositioninrasterstack`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Frecuencia menor que

Evalúa celda por celda la frecuencia (cantidad de veces) que los valores de una pila de rásteres de entrada son menores que el valor de un ráster de valores. La extensión y resolución del ráster de salida está definida por la capa ráster de entrada y siempre es del tipo `Int32`.

Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres; use GDAL para usar otras bandas en el análisis. El valor de salida sin datos se puede configurar manualmente.

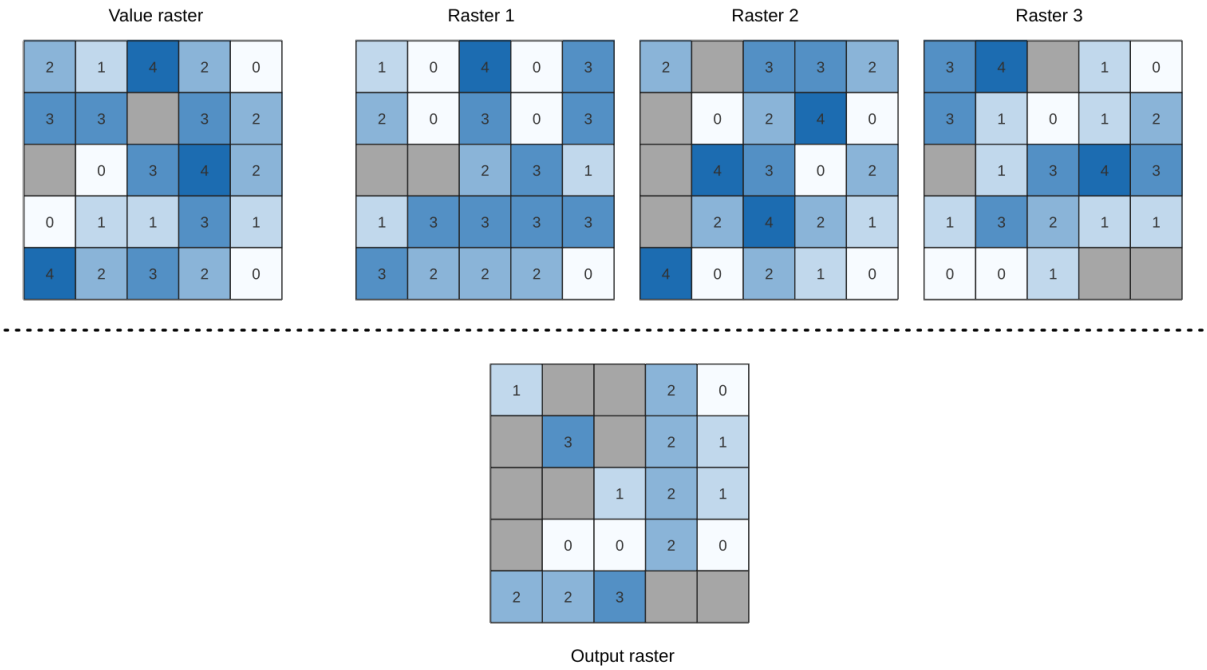


Figura 24.56: Para cada celda del ráster de salida, el valor representa la cantidad de veces que las celdas correspondientes en la lista de rásteres son menores que el ráster de valor. Se tienen en cuenta las celdas `NoData` (grises).

Ver también

Frecuencia de igualdad, *Frecuencia mayor que*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Valor ráster de entrada	`INPUT_VALUE_RA`	[ráster]	La capa de valor de entrada sirve como capa de referencia para las capas de muestra
Banda ráster de valor	`INPUT_VALUE_RA`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Seleccione la banda que desea usar como muestra

continúe en la próxima página

Tabla 24.340 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Capas ráster entrante	`INPUT_RASTERS`	[ráster] [lista]	Capas ráster para evaluar. Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres.
Ignorar valores NoData	`IGNORE_NODATA`	[booleano] Predeterminado: Falso	Si no está marcada, cualquier celda sin datos en el ráster de valor o la pila de capas de datos dará como resultado una celda sin datos en el ráster de salida
Capa de salida	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData Opcional	`OUTPUT_NO_DATA`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Conteo de celdas con ocurrencias de valor igual	`FOUND_LOCATION`	[numérico: entero]	
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

continúe en la próxima página

Tabla 24.342 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Frecuencia media en ubicaciones de celda válidas	`MEAN_FREQUENCY`	[numérico: doble]	
Conteo de ocurrencias del valor	`OCCURRENCE_COU`	[numérico: entero]	
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida

Código Python

Algoritmo ID: `native:lessthanfrequency`

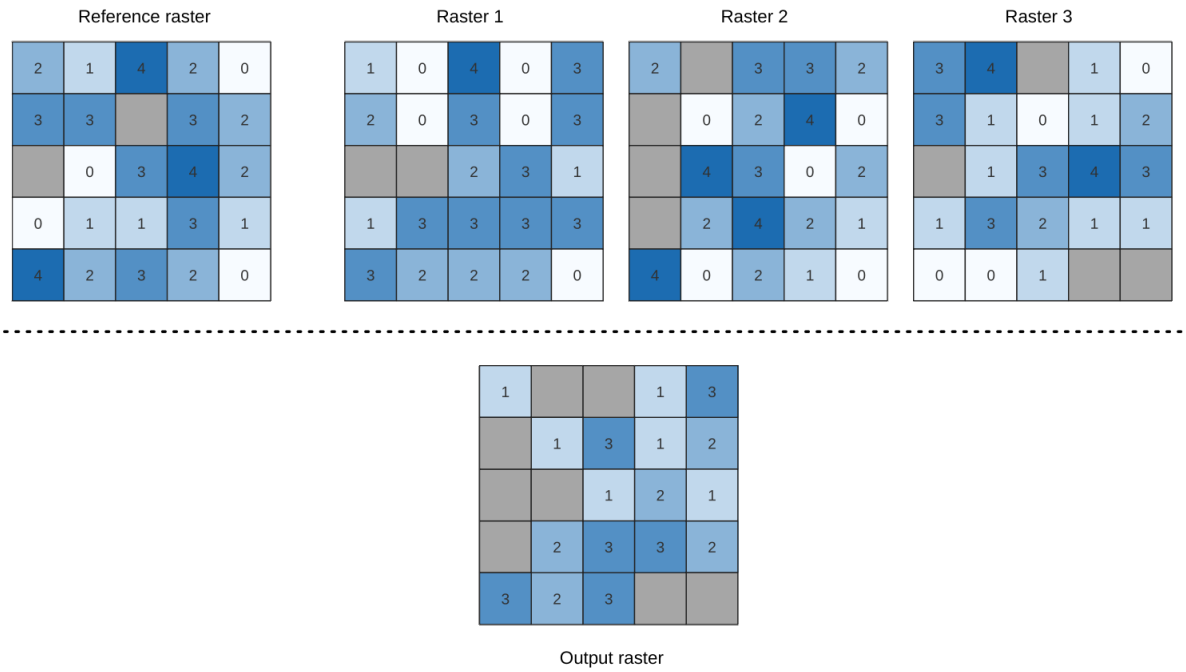
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Posición más baja en pila ráster

Evalúa celda por celda la posición del ráster con el valor más bajo en una pila de rásteres. Los conteos de posición comienzan con 1 y van hasta el número total de rásteres de entrada. El orden de los rásteres de entrada es relevante para el algoritmo. Si varios rásteres presentan el valor más bajo, el primer ráster se utilizará para el valor de posición.

Si se utilizan rásteres multibanda en la pila de ráster de datos, el algoritmo siempre realizará el análisis en la primera banda de los rásteres; use GDAL para usar otras bandas en el análisis. Cualquier celda sin datos en la pila de capas ráster dará como resultado una celda sin datos en el ráster de salida a menos que se marque el parámetro «ignore NoData». El valor de salida sin datos se puede configurar manualmente. La extensión y resolución de los rásteres de salida está definida por una capa ráster de referencia y siempre es del tipo `Int32`.



Ver también

Posición más alta en pila ráster

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas ráster entrante	`INPUT_RASTERS`	[ráster] [lista]	Lista de capas ráster para comparar
Capa de referencia	`REFERENCE_LAYE`	[ráster]	La capa de referencia para la creación de la capa de salida (extensión, SRC, dimensiones en píxeles)
Ignorar valores NoData	`IGNORE_NODATA`	[booleano] Predeterminado: Falso	Si no está marcada, cualquier celda sin datos en la pila de capas de datos dará como resultado una celda sin datos en el ráster de salida
Capa de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la trama de salida que contiene el resultado. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData	`OUTPUT_NODATA_`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado
Identificador de autoridad de SRC	`CRS_AUTHID`	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

Código Python

Algoritmo ID: `native:lowestpositioninrasterstack`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ráster Booleano AND

Calcula el valor booleano `AND` para un conjunto de rásteres de entrada. Si todos los rásteres de entrada tienen un valor distinto de cero para un píxel, ese píxel se establecerá en `1` en el ráster de salida. Si alguno de los rásteres de entrada tiene valores de `0` para el píxel, se establecerá en `0` en el ráster de salida.

El parámetro de la capa de referencia especifica una capa ráster existente para usar como referencia al crear el ráster de salida. El ráster de salida tendrá la misma extensión, SRC y dimensiones de píxeles que esta capa.

De forma predeterminada, un píxel NoData en CUALQUIERA de las capas de entrada resultará en un píxel NoData en el ráster de salida. Si la opción *Tratar los valores NoData como falsos* está marcada, las entradas NoData se tratarán igual que un valor de entrada `0`.

Ver también

Ráster Booleano OR

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Lista de capas ráster entrantes
Capa de referencia	`REF_LAYER`	[ráster]	La capa de referencia desde la que crear la capa de salida (extensión, SRC, dimensiones en píxeles)
Tratar los valores NoData como falsos	`NODATA_AS_FALS`	[booleano] Predeterminado: Falso	Tratar como 0 los valores NoData de los ficheros de entrada al realizar la operación.
Capa de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la trama de salida que contiene el resultado. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData	`NO_DATA`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Tipos de datos ráster salientes. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 4 — UInt32 (Entero de treinta y dos bit no firmado (quint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

continúe en la próxima página

Tabla 24.347 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Identificador de autoridad de SRC	`CRS_AUTHID`	[src]	El sistema de coordenadas de referencia de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente
Recuento de píxeles sin datos	`NODATA_PIXEL_C`	[numérico: entero]	Recuento de píxeles NoData en la capa ráster de salida
El recuento de pixels con valor True	`TRUE_PIXEL_COU`	[numérico: entero]	El recuento de pixels con valor verdadero (valor = 1) en la capa ráster saliente
Recuento de pixels con valor False	`FALSE_PIXEL_CO`	[numérico: entero]	El recuento de pixels con valor falso (valor = 0) en la capa ráster saliente
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado

Código Python

Algoritmo ID: `native:rasterbooleanand`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ráster Booleano OR

Calcula el valor booleano `O` para un conjunto de rásteres de entrada. Si todos los rásteres de entrada tienen un valor cero para un píxel, ese píxel se establecerá en `0` en el ráster de salida. Si alguno de los rásteres de entrada tiene valores de `1` para el píxel, se establecerá en `1` en el ráster de salida.

El parámetro de la capa de referencia especifica una capa ráster existente para usar como referencia al crear el ráster de salida. El ráster de salida tendrá la misma extensión, SRC y dimensiones de píxeles que esta capa.

De forma predeterminada, un píxel NoData en CUALQUIERA de las capas de entrada resultará en un píxel NoData en el ráster de salida. Si la opción *Tratar los valores NoData como falsos* está marcada, las entradas NoData se tratarán igual que un valor de entrada `0`.

Ver también

Ráster Booleano AND

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Lista de capas ráster entrantes
Capa de referencia	`REF_LAYER`	[ráster]	La capa de referencia desde la que crear la capa de salida (extensión, SRC, dimensiones en píxeles)
Tratar los valores NoData como falsos	`NODATA_AS_FALS`	[booleano] Predeterminado: Falso	Tratar como 0 los valores NoData de los ficheros de entrada al realizar la operación.
Capa de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la trama de salida que contiene el resultado. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData	`NO_DATA`	[numérico: doble] Predeterminado: -9999.0	Valor a utilizar para NoData en la capa de salida

continúe en la próxima página

Tabla 24.350 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Tipos de datos ráster salientes. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 4 — UInt32 (Entero de treinta y dos bit no firmado (quint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Identificador de autoridad de SRC	`CRS_AUTHID`	[src]	El sistema de coordenadas de referencia de la capa ráster saliente
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente
Recuento de píxeles sin datos	`NODATA_PIXEL_C`	[numérico: entero]	Recuento de píxeles NoData en la capa ráster de salida
El recuento de pixels con valor True	`TRUE_PIXEL_COU`	[numérico: entero]	El recuento de pixels con valor verdadero (valor = 1) en la capa ráster saliente

continúe en la próxima página

Tabla 24.351 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Recuento de pixels con valor False	`FALSE_PIXEL_CO`	[numérico: entero]	El recuento de pixels con valor falso (valor = 0) en la capa ráster saliente
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente contenedora del resultado

Código Python

Algoritmo ID: `native:rasterbooleanor`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Calculadora ráster

Ejecuta operaciones algebraicas usando capas ráster.

La capa resultante tendrá sus valores calculados de acuerdo con una expresión. La expresión puede contener valores numéricos, operadores y referencias a cualquiera de las capas del proyecto actual.

Ver también

Calculadora Ráster (virtual), *Calculadora ráster*, *Calculadora Ráster*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Lista de capas ráster entrantes
Expresión	`EXPRESSION`	[expresión]	*Expresión basada en ráster* que se utilizará para calcular la capa ráster de salida.

continúe en la próxima página

Tabla 24.352 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extensión de salida Opcional	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Si no se especifica la extensión, se utilizará la extensión mínima que cubra todas las capas de referencia seleccionadas. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Tamaño de la celda de salida (dejar vacío para que se ajuste automáticamente). Opcional	`CELLSIZE`	[numérico: doble]	Tamaño de celda de la capa ráster de salida. Si no se especifica el tamaño de celda, se utilizará el tamaño de celda mínimo de la capa(s) de referencia seleccionada. El tamaño de celda será el mismo para los ejes X e Y.
SRC saliente Opcional	`CRS`	[src]	SRC de la capa ráster de salida. Si no se especifica el SRC de salida, se utilizará el SRC de la primera capa de referencia.
Calculado	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Calculado	`OUTPUT`	[ráster]	Archivo ráster saliente con los valores calculados.

Código Python

ID Algoritmo: `native:rastercalc`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Calculadora Ráster (virtual)

Realiza operaciones algebraicas con capas ráster y genera resultados en memoria.

La capa resultante tendrá sus valores calculados de acuerdo con una expresión. La expresión puede contener valores numéricos, operadores y referencias a cualquiera de las capas del proyecto actual.

Una capa ráster virtual es una capa ráster definida por su URI y cuyos píxeles se calculan sobre la marcha. No se trata de un nuevo archivo en el disco; la capa ráster virtual sigue conectada a las capas ráster utilizadas en el cálculo, lo que significa que borrar o mover estas capas ráster la dañaría. Se puede proporcionar una etiqueta *Nombre de la capa*, de lo contrario la expresión del cálculo se utiliza como tal. Al eliminar la capa virtual del proyecto se borra, y puede hacerse persistente en un archivo utilizando el menú contextual de la capa *Exportar* ► *Guardar como…*.

Ver también

Calculadora ráster, *Calculadora ráster*, *Calculadora Ráster*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Lista de capas ráster entrantes
Expresión	`EXPRESSION`	[expresión]	*Expresión basada en ráster* que se utilizará para calcular la capa ráster de salida.

continúe en la próxima página

Tabla 24.355 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extensión de salida Opcional	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Si no se especifica la extensión, se utilizará la extensión mínima que cubra todas las capas de referencia seleccionadas. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Tamaño de la celda de salida (dejar vacío para que se ajuste automáticamente). Opcional	`CELLSIZE`	[numérico: doble]	Tamaño de celda de la capa ráster de salida. Si no se especifica el tamaño de celda, se utilizará el tamaño de celda mínimo de la capa(s) de referencia seleccionada. El tamaño de celda será el mismo para los ejes X e Y.
SRC saliente Opcional	`CRS`	[src]	SRC de la capa ráster de salida. Si no se especifica el SRC de salida, se utilizará el SRC de la primera capa de referencia.
Nombre de la capa de salida Opcional	`LAYER_NAME`	[cadena de texto]	El nombre a asignar a la capa generada. Si no se establece, se utiliza el texto de la expresión de cálculo.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Calculado	`OUTPUT`	[ráster]	Salida de la capa ráster virtual con los valores calculados.

Código Python

ID Algoritmo: `native:virtualrastercalc`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Propiedades de capa ráster

Devuelve las propiedades básicas de la capa ráster dada, incluyendo la extensión, el tamaño en píxeles y las dimensiones de los píxeles (en unidades de mapa), el número de bandas y el valor NoData.

Este algoritmo está diseñado para usarse como un medio para extraer estas propiedades útiles para usarlas como valores de entrada para otros algoritmos en un modelo, p. ej. para permitir pasar los tamaños de píxeles de un ráster existente a un algoritmo de ráster GDAL.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de banda Opcional	`BAND`	[banda ráster] Predeterminado: No establecido	Si devolver también las propiedades de una banda específica. Si se especifica una banda, también se devuelve el valor noData para la banda seleccionada.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Número de bandas en raster	`BAND_COUNT`	[numérico: entero]	El número de bandas en el ráster
Identificador de autoridad de SRC	`CRS_AUTHID`	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión de la capa ráster en el CRS
La banda tiene un conjunto de valores NoData	`HAS_NODATA_VAL`	[Booleano]	Indica si la capa ráster tiene un valor establecido para los píxeles NoData en la banda seleccionada.
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de columnas en la capa ráster
Valor de la banda NoData	`NODATA_VALUE`	[numérico: doble]	El valor (si está configurado) de los píxeles NoData en la banda seleccionada
Tamaño de píxel (altura) en unidades de mapa	`PIXEL_HEIGHT`	[numérico: entero]	Tamaño vertical en unidades de mapa del píxel

continúe en la próxima página

Tabla 24.358 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tamaño de píxel (ancho) en unidades de mapa	`PIXEL_WIDTH`	[numérico: entero]	Tamaño horizontal en unidades de mapa del píxel
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de filas en la capa ráster
Coordenada x máxima	`X_MAX`	[numérico: doble]	
Coordenada x mínima	`X_MIN`	[numérico: doble]	
Coordenada y máxima	`Y_MAX`	[numérico: doble]	
Coordenada y mínima	`Y_MIN`	[numérico: doble]	

Código Python

Algoritmo ID: `native:rasterlayerproperties`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Estadísticas de capa ráster

Calcula estadísticas básicas a partir de los valores de una banda dada de la capa ráster. La salida se carga en el menú *Procesos -> Visor de Resultados*.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa entrante	SI el ráster es multibanda, elija la banda de la que quiera obtener las estadísticas.
Estadísticas	`OUTPUT_HTML_FI`	[html] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Valor máximo	`MAX`	[numérico: doble]	
Valor promedio	`MEAN`	[numérico: doble]	
Valor mínimo	`MIN`	[numérico: doble]	
Estadísticas	`OUTPUT_HTML_FI`	[html]	El archivo de salida contiene la siguiente información: • Archivo analizado: ruta de la capa ráster • Valor mínimo: valor mínimo del ráster • Valor máximo: valor máximo del ráster • Rango: diferencia entre los valores máximo y mínimo • Sum: suma total de los valores • Valor medio: media de todos los valores • Desviación estándar: desviación estándar de los valores • Suma de cuadrados: suma de las diferencias al cuadrado de cada observación de la media general
Rango	`RANGE`	[numérico: doble]	
Desviación estándar	`STD_DEV`	[numérico: doble]	
Suma	`SUM`	[numérico: doble]	
Suma de cuadrados	`SUM_OF_SQUARES`	[numérico: doble]	

Código Python

Algorithm ID: `native:rasterlayerstatistics`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Informe de valores únicos de capa ráster

Devuelve el recuento y el área de cada valor único en una capa ráster dada. El cálculo del área se realiza en la unidad de área del SRC de la capa.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa entrante	SI el ráster es multibanda, elija la banda de la que quiera obtener las estadísticas.
Informe de valores únicos	`OUTPUT_HTML_FI`	[archivo] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…
Tabla de valores únicos	`OUTPUT_TABLE`	[vector: tabla] Preestablecido: `[Omitir salida]`	Especificación de la tabla para valores únicos: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Identificador de autoridad de SRC	`CRS_AUTHID`	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	`EXTENT`	[cadena de texto]	La extensión espaical de la capa ráster saliente
Altura en pixels	`HEIGHT_IN_PIXE`	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento de píxeles sin datos	`NODATA_PIXEL_C`	[numérico: entero]	Número de píxeles NoData en la capa ráster de salida
Recuento total de pixels	`TOTAL_PIXEL_CO`	[numérico: entero]	El recuento de pixels en la capa ráster saliente

continúe en la próxima página

Tabla 24.362 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Informe de valores únicos	`OUTPUT_HTML_FI`	[html]	El archivo HTML saliente contiene la siguiente información: • Archivo analizado: la ruta de la capa ráster • Extensión: coordenadas xmin, ymin, xmax, ymax de la extensión • Proyección: proyección de la capa • Anchura en pixels: número de columnas y tamaño de anchura de pixel • Altura en pixels: número de columans y tamaño de ancho de píxel • Recuento total de pixel: recuento de todos los pixels • Recuento de píxeles NoData: recuento de píxeles con valor NoData
Tabla de valores únicos	`OUTPUT_TABLE`	[vector: tabla]	Una tabla con tres columnas: • *valor*: valor de pixel • *recuento*: recuento de pixels con este valor • *m2* o *deg2* o *ft2* o … : área total de píxeles con este valor. El nombre de la columna depende de la unidad de superficie del SRC de la capa y el cálculo se realiza en esa unidad.
Anchura en pixels	`WIDTH_IN_PIXEL`	[numérico: entero]	El número de columnas en la capa ráster de salida

Código Python

Algoritmo ID: `native:rasterlayeruniquevaluesreport`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Estadísticas zonales de la capa ráster

Calcula estadísticas para los valores de una capa ráster, categorizados por zonas definidas en otra capa ráster.

Ver también

Estadísticas de zona

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[ráster]	Capa ráster de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Si el ráster es multibanda elige la banda sobre la que quieras calcular las estadísticas.
Capa de zonas	`ZONES`	[ráster]	Zonas de definición de capa ráster. Las zonas están dadas por píxeles contiguos que tienen el mismo valor de píxel.
Número de zonas de banda	`ZONES_BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Si el ráster es multibanda, elige la banda que define las zonas
Estadísticas	`OUTPUT_TABLE`	[vector: tabla] Predeterminado: `[Crear capa temporal]`	Especificación del informe de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Capa de referencia Opcional	REF_LAYER	[enumeración] Predeterminado: 0	Capa ráster utilizada para calcular los centroides que se utilizarán como referencia al determinar las zonas en la capa de salida. Uno de: • 0 — Input layer: las zonas se determinan muestreando el valor de la capa ráster de zona en el centroide de cada píxel de la capa ráster de origen • 1 — Zones layer: la capa ráster de entrada se muestreará en el centroide de cada píxel de la capa ráster de zonas

Salidas

Etiqueta	Nombre	Tipo	Descripción
Identificador de autoridad de SRC	CRS_AUTHID	[cadena de texto]	El sistema de coordenadas de referencia de la capa ráster saliente
Extensión	EXTENT	[cadena de texto]	La extensión espaical de la capa ráster saliente
Altura en pixels	HEIGHT_IN_PIXE	[numérico: entero]	El número de filas en la capa ráster de salida
Recuento de píxeles sin datos	NODATA_PIXEL_C	[numérico: entero]	Número de píxeles NoData en la capa ráster de salida
Estadísticas	OUTPUT_TABLE	[vector: tabla]	La capa de salida contiene la siguiente información **para cada zona**: • Área: el área en unidades ráster cuadradas en la zona; • Suma: la suma total de valores de pixel en la zona; • Recuento: el número de pixels en la zona; • Min: el valor mínimo de pixel en la zona; • Max: el valor máximo de pixel en la zona; • Media: la media de los valores de pixel en la zona;
Recuento total de pixels	TOTAL_PIXEL_CO	[numérico: entero]	El recuento de pixels en la capa ráster saliente
Anchura en pixels	WIDTH_IN_PIXEL	[numérico: entero]	El número de columnas en la capa ráster de salida

Código Python

Algoritmo ID: `native:rasterlayerzonalstats`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Raster minimum/maximum

Extracts the minimum and maximum pixel values from a specified band of a raster layer. If multiple pixels share the same minimum or maximum value, only one of them will be included in the output.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Raster layer from which to extract the minimum and maximum pixel values.
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	The band of the raster to analyze. If the raster is multiband, specify the band number (starting from 1).
Extract	`EXTREMA`	[enumeración] Default: 0 (Minimum and Maximum)	Choose which extrema to extract: • 0: Minimum and Maximum • 1: Minimum only • 2: Maximum only
Capa de salida	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Specification of the output layer. *One of*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[vector: punto]	Vector layer with point features at the locations of the minimum and/or maximum pixel values.

Código Python

Algorithm ID: `native:rasterminmax`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Raster rank

`NEW in 3.44`

Performs a cell-by-cell analysis in which output values match the rank of a sorted list of overlapping cell values from input layers. The output raster will be multi-band if multiple ranks are provided. If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters.

Ver también

Clasificación porcentual de la pila de celdas a partir del valor, *percentil de pila de celdas*, *Porcentaje de la pila de celdas a partir de la capa ráster*, *Estadísticas de celda*, *Calculadora ráster*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT_RASTERS`	[ráster] [lista]	Input raster layer(s) to calculate the rank from.
Rank	`RANKS`	[numeric: integer] [list] Default: [1]	List of ranks to calculate for each pixel in the sorted list of cells values from the input rasters. Cells whose values match the given rank(s) are kept in the output layer. The ranks are 1-based, meaning that the first rank is 1. Negative values are also supported.
NoData value handling	`NODATA_HANDLIN`	[enumeración] Predeterminado: 0	Choose how to handle NoData values in the input raster: • 0 — Exclude NoData from values lists: will not include NoData values in the list of pixel values when calculating ranks. • 1 — Presence of NoData in a values list results in NoData output cells: the output raster will have NoData values in the corresponding locations.

continúe en la próxima página

Tabla 24.368 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Ranked	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Specification of the output raster containing the rank values. *One of*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Extensión de salida Opcional	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Si no se especifica la extensión, se utilizará la extensión mínima que cubra todas las capas de referencia seleccionadas. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Output cell size Opcional	`CELL_SIZE`	[numérico: doble]	Tamaño de celda de la capa ráster de salida. Si no se especifica el tamaño de celda, se utilizará el tamaño de celda mínimo de la capa(s) de referencia seleccionada. El tamaño de celda será el mismo para los ejes X e Y.
SRC saliente Opcional	`CRS`	[src]	SRC de la capa ráster de salida. Si no se especifica el SRC de salida, se utilizará el SRC de la primera capa de referencia.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ranked	`OUTPUT`	[ráster]	Output raster layer containing the result.

Código Python

Algorithm ID: `native:rasterrank`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Volumen de la superficie ráster

Calcula el volumen debajo de una superficie ráster en relación con un nivel base determinado. Esto es principalmente útil para modelos digitales de elevación (MDE).

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de ENTRADA	`INPUT`	[ráster]	Ráster de entrada, representando una superficie
Número de banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Si el ráster es multibanda, elige la banda que debe definir la superficie.
Nivel Base	`LEVEL`	[numérico: doble] Predeterminado: 0.0	Defina un valor base o de referencia. Esta base se utiliza en el cálculo del volumen de acuerdo con el parámetro `Método` (ver más abajo).

continúe en la próxima página

Tabla 24.371 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Methodo	`METHOD`	[enumeración] Predeterminado: 0	Defina el método para el cálculo del volumen dado por la diferencia entre el valor del píxel de la trama y el `Nivel base`. Opciones: • 0 — Contar solo por encima del nivel base: solo los píxeles por encima del nivel base se sumarán al volumen. • 1 — Contar solo por debajo del nivel base: solo los píxeles por debajo del nivel base se sumarán al volumen. • 2 — Restar volúmenes por debajo del nivel base: los píxeles por encima del nivel base se sumarán al volumen, los píxeles por debajo del nivel base se restarán del volumen. • 3 — Agregar volúmenes por debajo del nivel base: agregue el volumen independientemente de si el píxel está por encima o por debajo del nivel base. Esto equivale a sumar los valores absolutos de la diferencia entre el valor de píxel y el nivel base.
Informe de volumen de superficie	`OUTPUT_HTML_FI`	[html] Predeterinado: `[Save to temporary file]`	Especificación del informe HTML de salida. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Tabla de volumen de superficie	`OUTPUT_TABLE`	[vector: tabla] Preestablecido: `[Omitir salida]`	Especificación de la tabla de salida. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Volumen	`VOLUME`	[numérico: doble]	El volumen calculado
Área	`AREA`	[numérico: doble]	El área en unidades cuadradas de mapa
Pixel_count	`PIXEL_COUNT`	[numérico: entero]	El número total de pixels que han sido analizados

continúe en la próxima página

Tabla 24.372 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Informe de volumen de superficie	`OUTPUT_HTML_FI`	[html]	El informe de salida (conteniendo volumen, área y recuento de pixel) en formato HTML
Tabla de volumen de superficie	`OUTPUT_TABLE`	[vector: tabla]	La tabla de salida (conteniendo volumen, área y recuento de pixels)

Código Python

Algoritmo ID: `native:rastersurfacevolume`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Reclasificar por capa

Reclasifica una banda ráster asignando nuevos valores de clase basados en los rangos especificados en una tabla de vectores.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa ráster	`INPUT_RASTER`	[ráster]	Capa ráster a reclasificar
Número de banda	`RASTER_BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	Si el ráster es multibanda, elige la banda que quieras reclasificar.
Capa que contiene rupturas de clase	`INPUT_TABLE`	[vectorial: cualquiera]	Capa vectorial que contiene los valores que se utilizarán para la clasificación.
Campo de valor mínimo de clase	`MIN_FIELD`	[campo de tabla: numérico]	Campo con el valor mínimo del rango para la clase. Utilice `-inf` para capturar el valor más bajo posible.
Campo de valor máximo de clase	`MAX_FIELD`	[campo de tabla: numérico]	Campo con el valor máximo del rango para la clase. Utilice `inf` para capturar el valor más alto posible.
Campo de valor de salida	`VALUE_FIELD`	[campo de tabla: numérico]	Campo con el valor que se asignará a los píxeles que caigan en la clase (entre los valores min y max correspondientes). Utilice `nan` para establecer el valor del rango en NoData.

continúe en la próxima página

Tabla 24.373 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Ráster reclasificado	`OUTPUT`	[ráster] Predeterminado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData	`NO_DATA`	[numérico: doble] Predeterminado: -9999.0	Valor a aplicar a los valores NoData.
Límites de rango	`RANGE_BOUNDARI`	[enumeración] Predeterminado: 0	Define reglas de comparación para la clasificación. Opciones: • 0 — min < valor <= max • 1 — min <= valor < max • 2 — min <= valor <= max • 3 — min < valor < max
Utilizar NoData cuando ningún rango coincide con el valor	`NODATA_FOR_MIS`	[booleano] Predeterminado: Falso	Aplica el valor NoData a los valores de banda que no pertenecen a ninguna clase. Si es False, se mantiene el valor original.
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el formato del fichero raster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 4 — UInt32 (Entero de treinta y dos bit no firmado (quint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

continúe en la próxima página

Tabla 24.374 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster reclasificado	`OUTPUT`	[ráster]	Capa ráster de salida con valores de banda reclasificados

Código Python

Algoritmo ID: `native:reclassifybylayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Reclasificar por tabla

Reclasifica una banda de ráster asignando nuevos valores de clase basados en los rangos especificados en una tabla fija.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa ráster	`INPUT_RASTER`	[ráster]	Capa ráster a reclasificar
Número de banda	`RASTER_BAND`	[banda ráster] Predeterminado: 1	Banda ráster para la cual quieres recalcular valores.

continúe en la próxima página

Tabla 24.376 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tabla de reclasificación	`TABLE`	[vector: tabla]	Una tabla de 3 columnas para rellenar con los valores para establecer los límites de cada clase (`Mínimo` y `Máximo`) y el nuevo `Valor` para asignar a los valores de banda que caen en la clase. El valor `-inf` se puede utilizar como mínimo, `inf` como máximo, y `nan` se puede utilizar para establecer el valor de salida a NoData.
Ráster reclasificado	`OUTPUT`	[ráster] Predeterminado: `[Save to temporary file]`	Especificación de la capa raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor de salida NoData	`NO_DATA`	[numérico: doble] Predeterminado: -9999.0	Valor a aplicar a los valores NoData.
Límites de rango	`RANGE_BOUNDARI`	[enumeración] Predeterminado: 0	Define reglas de comparación para la clasificación. Opciones: • 0 — min < valor <= max • 1 — min <= valor < max • 2 — min <= valor <= max • 3 — min < valor < max
Utilizar NoData cuando ningún rango coincide con el valor	`NODATA_FOR_MIS`	[booleano] Predeterminado: Falso	Aplica el valor NoData a los valores de banda que no pertenecen a ninguna clase. Si es False, se mantiene el valor original.

continúe en la próxima página

Tabla 24.377 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el formato del fichero raster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 4 — UInt32 (Entero de treinta y dos bit no firmado (quint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster reclasificado	`OUTPUT`	[ráster]	Capa ráster de salida con valores de banda reclasificados

Código Python

Algoritmo ID: `native:reclassifybytable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Reescalar ráster

Reescala una capa ráster con un nuevo rango de valores, mientras que preserva la forma (distribución) del histograma del ráster (valores de píxeles). Los valores de entrada son mapeados usando interpolación lineal los valores mínimo y máximo del ráster fuente hacia el rango de píxeles mínimo y máximo destino.

De forma predeterminada, el algoritmo conserva el valor original de NoData, pero existe una opción para anularlo.

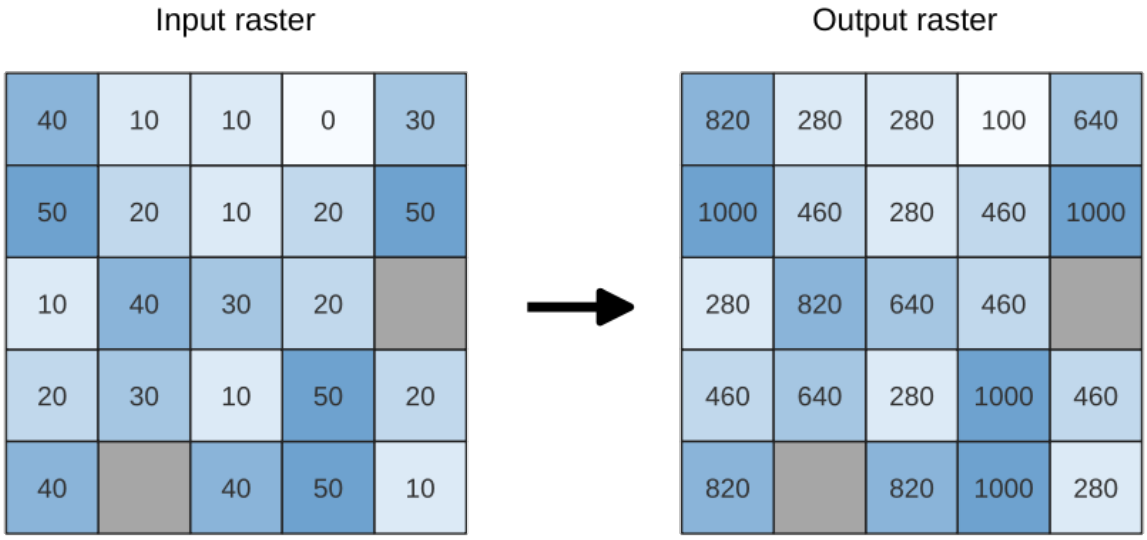


Figura 24.57: Cambiar la escala de los valores de una capa ráster de [0 - 50] a [100 - 1000]

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Ráster de Entrada	`INPUT`	[ráster]	Capa ráster que se usará para cambiar la escala
Número de banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa entrante	Si el ráster es multibanda, elija una banda.
Nuevo valor mínimo	`MINIMUM`	[numérico: doble] Valor predeterminado: 0.0	Valor mínimo de píxel para usar en la capa reescalada
Nuevo valor máximo	`MAXIMUM`	[numérico: doble] Valor predeterminado: 255.0	Valor máximo de píxel para usar en la capa reescalada
Nuevo valor NoData Opcional	`NODATA`	[numérico: doble] Predeterminado: No establecido	Valor a asignar a los píxeles NoData. Si no se establece, se conservan los valores NoData originales.
Reescalado	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la capa raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Reescalado	`OUTPUT`	[ráster]	Capa ráster de salida con valores de banda reescalados

Código Python

Algoritmo ID: `native:rescaleraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Redondear ráster

Redondea los valores de celda de un conjunto de datos ráster según el número especificado de decimales.

Alternativamente, se puede usar un número negativo de lugares decimales para redondear valores a potencias de una base n. Por ejemplo, con un valor base n de 10 y posiciones decimales de -1, el algoritmo redondea los valores de la celda a múltiplos de 10, -2 los redondea a múltiplos de 100, y así sucesivamente. Se pueden elegir valores base arbitrarios, el algoritmo aplica el mismo principio multiplicativo. El redondeo de los valores de la celda a múltiplos de una base n se puede usar para generalizar las capas ráster.

El algoritmo conserva el tipo de datos del ráster de entrada. Por lo tanto, los rásteres de tipo byte/entero solo se pueden redondear a múltiplos de una base n; de lo contrario, se genera un aviso y el ráster se copia como ráster de tipo byte/entero.

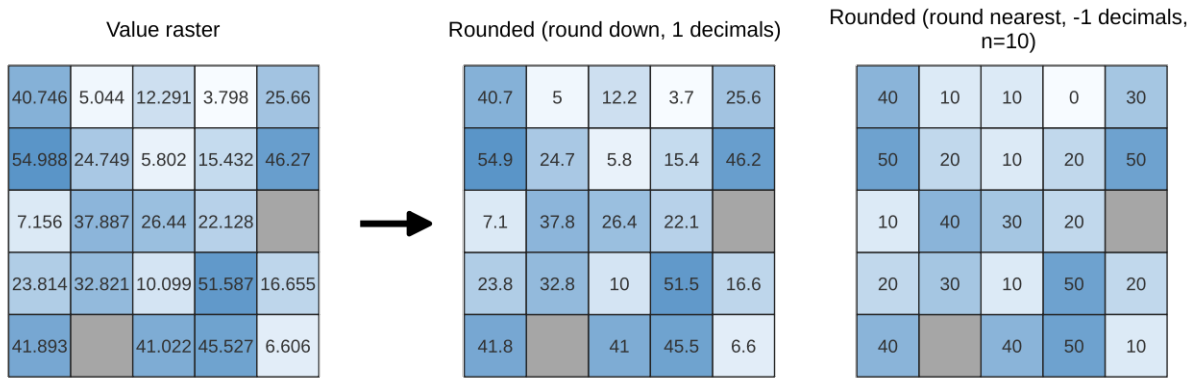


Figura 24.58: Redondeo de valores de un ráster

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Ráster de Entrada	`INPUT`	[ráster]	El ráster para procesar.
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	La banda del ráster
Dirección del redondeo	`ROUNDING_DIREC`	[lista] Predeterminado: 1	Cómo elegir el valor objetivo redondeado. Las opciones son: • 0 — Redondear hacia arriba • 1 — Redondear al más cercano • 2 — Redondear hacia abajo
Número de posiciones decimales	`DECIMAL_PLACES`	[numérico: entero] Preestablecido: 2	Número de decimales a redondear. Use valores negativos para redondear los valores de la celda a un múltiplo de una base n
Ráster de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Base n para redondear a múltiples de n	`BASE_N`	[numérico: entero] Predeterminado: 10	Cuando el parámetro `DECIMAL_PLACES` es negativo, los valores ráster se redondean a múltiplos del valor base n
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster]	La capa ráster de salida con valores redondeados para la banda seleccionada.

Código Python

Algoritmo ID: `native:roundrastervalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Valores de muestra ráster

Extrae valores ráster en las ubicaciones de los puntos. Si la capa ráster es multibanda, se muestrea cada banda.

La tabla de atributos de la capa resultante tendrá tantas columnas nuevas como el recuento de bandas de la capa ráster.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vector: punto]	Capa vectorial de puntos que se utilizará para el muestreo
Capa ráster	`RASTERCOPY`	[ráster]	Capa ráster a muestrear en las ubicaciones de los puntos dados.
Prefijo de columna saliente	`COLUMN_PREFIX`	[cadena de texto] Predeterminado: "SAMPLE_"	Prefijo para los nombres de las columnas agregadas.
Muestreada Opcional	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida que contiene los valores muestreados. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Muestreada	`OUTPUT`	[vector: punto]	La capa de salida que contiene los valores muestreados.

Código Python

Algoritmo ID: `native:rastersampling`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Histograma zonal

Agrega campos que representan recuentos de cada valor único de una capa ráster contenida en entidades poligonales.

La tabla de atributos de la capa de salida tendrá tantos campos como los valores únicos de la capa ráster que intersecta el polígono(s).

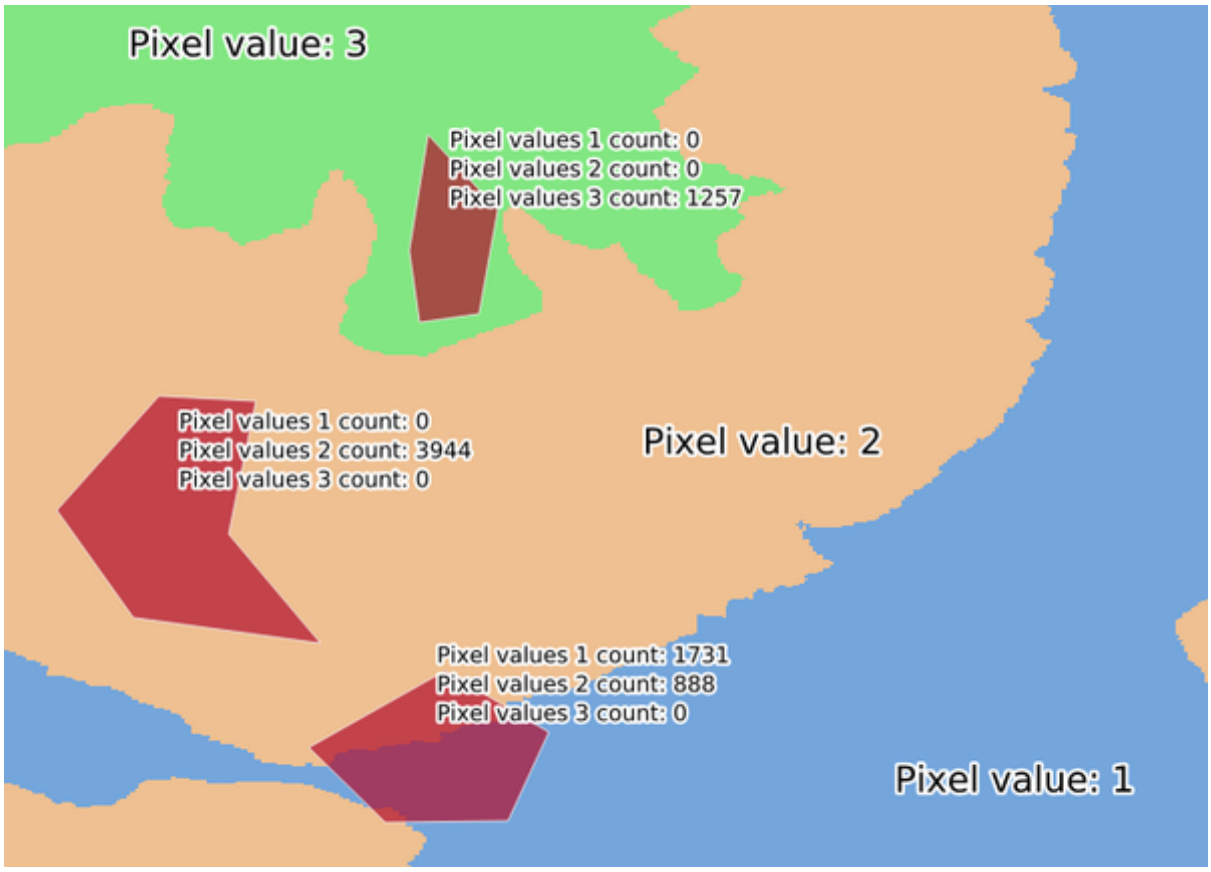


Figura 24.59: Ejemplo de histograma de capa ráster

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa ráster	`INPUT_RASTER`	[ráster]	Capa ráster de entrada.
Número de banda	`RASTER_BAND`	[banda ráster] Predeterminado: La primera banda de la capa entrante	Si el ráster es multibanda, elija una banda.
Capa de vector que contiene zonas	`INPUT_VECTOR`	[vectorial: poligonal]	Capa de polígono vectorial que define las zonas.
Prefijo de columna saliente	`COLUMN_PREFIX` Opcional	[cadena de texto] Predeterminado: "HISTO_"	Prefijo para los nombres de las columnas de salida.
Zonas salientes	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de polígonos del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Zonas salientes	`OUTPUT`	[vectorial: poligonal]	La capa de polígono de vector de salida.

Código Python

Algoritmo ID: `native:zonalhistogram`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Zonal Minimum/Maximum Point

Extracts point features corresponding to the minimum and maximum pixel values within polygon zones.

The output will contain one point feature for the minimum and one for the maximum raster value for every individual zonal feature from a polygon layer. The created point layer will be in the same spatial reference system as the selected raster layer.

Advertencia

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vectorial: poligonal]	Polygon layer defining the zones.
Capa ráster	`RASTER`	[ráster]	Raster layer to extract the minimum and maximum values from.
Raster Band	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa ráster	If the raster has multiple bands, select the band to process.
Zonal extrema	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Specification of the output layer. *One of*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Zonal extrema	`OUTPUT`	[vector: punto]	Output layer containing the minimum and maximum points for each zone.

Código Python

Algorithm ID: `native:zonalminmaxpoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Estadísticas de zona

Calcula las estadísticas de una capa ráster para cada entidad de una capa vectorial de polígono superpuesta.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa vectorial poligonal que contiene las zonas.
Capa ráster	`INPUT_RASTER`	[ráster]	Capa ráster de entrada.
Banda ráster	`RASTER_BAND`	[banda ráster] Predeterminado: La primera banda de la capa entrante	Si el ráster es multibanda, elija una banda para las estadísticas.
Prefijo de columna saliente	`COLUMN_PREFIX`	[cadena de texto] Predeterminado: "_"	Prefijo para los nombres de las columnas de salida.
Estadísticas a calcular	`STATISTICS`	[enumeración] [lista] Predeterminado: [0,1,2]	Lista de operador estadístico para la salida. Opciones: • 0 — Recuento • 1 — Suma • 2 — Media • 3 — Mediana • 4 — Desviación Estándar • 5 — Mínimo • 6 — Máximo • 7 — Rango • 8 — Minoría • 9 — Mayoría • 10 — Variedad • 11 — Varianza
Estadísticas de zona	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de polígonos del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Estadísticas de zona	`OUTPUT`	[vectorial: poligonal]	La capa vectorial zonal con estadísticas añadidas.

Código Python

Algoritmo ID: `native:zonalstatisticsfb`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.19 Creación de ráster

Crear capa ráster constante

Genera una capa ráster en la extensión dada y tamaño de celda rellenado con un valor específico

Adicionalmente se puede especificar un tipo de dato de salida. El algoritmo abortará si se ha sido introducido un valor que no puede ser representado por el tipo de dato de salida del ráster seleccionado.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión deseada	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC destino	`TARGET_CRS`	[src] Predeterminado: SRC del proyecto	SRC de la capa ráster de salida
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 1.0	Tamaño del píxel (X=Y) en unidades de mapa.

continúe en la próxima página

Tabla 24.389 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
**Valor constante*	NUMBER	[numérico: doble] Predeterminado: 1.0	Valor constante de píxel para la capa ráster de salida.
Constante	OUTPUT	[ráster] Predeterinado: [Save to temporary file]	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos del ráster de salida	OUTPUT_TYPE Predeterminado: 5	[enumeración]	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Byte • 1 — Entero16 • 2 — Entero sin signo16 • 3 — Entero32 • 4 — Entero sin signo32 • 5 — Flotante32 • 6 — Flotante64
Opciones de creación Opcional	CREATION_OPTIO (for QGIS <= 3.42, this was CREATE_OPTIONS	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Constante	OUTPUT	[ráster]	Ráster que cubre la extensión deseada con el tamaño y valor de píxel especificados.

Código Python

Algoritmo ID: `native:createconstantrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear una capa ráster aleatoria (distribución binomial)

Genera una capa ráster para una extensión dada y un tamaño de celda lleno con valores aleatorios distribuidos binomialmente.

Por omisión, los valores serán elegidos dado un N de 10 y una probabilidad de 0.5. Esto puede ser anulado usando el parámetro avanzado para N y probabilidad. El tipo de dato ráster se establece a tipos enteros (Entero16 por omisión). Los valores aleatorios de distribución binomial son definidos como números enteros positivos. Un ráster punto flotante representará una conversión de valores enteros a punto flotante.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión deseada	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC destino	`TARGET_CRS`	[src] Predeterminado: SRC del proyecto	SRC de la capa ráster de salida
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 1.0	Tamaño del píxel (X=Y) en unidades de mapa.

continúe en la próxima página

Tabla 24.392 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos del ráster de salida	`OUTPUT_TYPE` Predeterminado: 0	[enumeración]	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Entero16 • 1 — Entero sin signo16 • 2 — Entero32 • 3 — Entero sin signo32 • 4 — Flotante32 • 5 — Flotante64
N	`N`	[numérico: entero] Predeterminado: 10	
Probabilidad	`PROBABILITY`	[numérico: doble] Predeterminado: 0.5	
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: “”	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster]	Ráster que cubre la extensión deseada con el tamaño de celda lleno con valores aleatorios

Código Python

Algoritmo ID: `native:createrandombinomialrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear una capa ráster aleatoria (distribución exponencial)

Genera una capa ráster para una extensión dada y un tamaño de celda lleno de valores aleatorios distribuidos exponencialmente.

Por omisión, los valores serán elegidos dada una lambda de 1.0. Esto puede ser anulado usando el parámetro avanzado para lambda. El tipo de dato ráster se establece a Float32 por omisión ya que los valores aleatorios de distribución exponencial son números punto flotante.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión deseada	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC destino	`TARGET_CRS`	[src] Predeterminado: SRC del proyecto	SRC de la capa ráster de salida
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 1.0	Tamaño del píxel (X=Y) en unidades de mapa.

continúe en la próxima página

Tabla 24.395 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster] Predeterminado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos del ráster de salida	`OUTPUT_TYPE` Predeterminado: 0	[enumeración]	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Flotante32 • 1 — Flotante64
Lambda	`LAMBDA`	[numérico: doble] Predeterminado: 1.0	
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster]	Ráster que cubre la extensión deseada con el tamaño de celda lleno con valores aleatorios

Código Python

Algoritmo ID: `native:createrandomexponentialrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear una capa ráster aleatoria (distribución gamma)

Genera una capa ráster para una extensión dada y un tamaño de celda lleno con valores aleatorios distribuidos gamma.

Por omisión, los valores se elegirán con un valor alfa y beta de 1.0. Esto se puede anular utilizando el parámetro avanzado para alfa y beta. El tipo de datos ráster se establece en Flotante32 de forma predeterminada, ya que los valores aleatorios de la distribución gamma son números de punto flotante.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión deseada	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC destino	`TARGET_CRS`	[src] Predeterminado: SRC del proyecto	SRC de la capa ráster de salida
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 1.0	Tamaño del píxel (X=Y) en unidades de mapa.
Ráster de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos del ráster de salida	`OUTPUT_TYPE` Predeterminado: 0	[enumeración]	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Flotante32 • 1 — Flotante64
Alpha	`ALPHA`	[numérico: doble] Predeterminado: 1.0	
Beta	`BETA`	[numérico: doble] Predeterminado: 1.0	
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster]	Ráster que cubre la extensión deseada con el tamaño de celda lleno de valores distribuidos aleatoriamente

Código Python

Algoritmo ID: `native:createrandomgammarasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear una capa ráster aleatoria (distribución geométrica)

Genera una capa ráster para una extensión dada y un tamaño de celda lleno de valores aleatorios distribuidos geométricamente.

Por omisión, los valores se elegirán con una probabilidad de 0.5. Esto se puede anular utilizando el parámetro avanzado para el valor medio. El tipo de datos ráster se establece en tipos enteros (Entero16 por omisión). Los valores aleatorios de la distribución geométrica se definen como números enteros positivos. Un ráster de punto flotante representará una conversión de valores enteros a punto flotante.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión deseada	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC destino	`TARGET_CRS`	[src] Predeterminado: SRC del proyecto	SRC de la capa ráster de salida
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 1.0	Tamaño del píxel (X=Y) en unidades de mapa.
Ráster de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos del ráster de salida	`OUTPUT_TYPE` Predeterminado: 0	[enumeración]	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Entero16 • 1 — Entero sin signo16 • 2 — Entero32 • 3 — Entero sin signo32 • 4 — Flotante32 • 5 — Flotante64
Probabilidad	`PROBABILITY`	[numérico: doble] Predeterminado: 0.5	
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster]	Ráster que cubre la extensión deseada con el tamaño de celda lleno de valores distribuidos aleatoriamente

Código Python

Algoritmo ID: `native:createrandomgeometricrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear una capa ráster aleatoria (distribución binomial negativa)

Genera una capa ráster para una extensión dada y un tamaño de celda lleno con valores aleatorios distribuidos binomialmente negativos.

Por omisión, los valores se elegirán dado un parámetro de distribución k de 10.0 y una probabilidad de 0.5. Esto se puede anular utilizando los parámetros avanzados para k y probabilidad. El tipo de datos ráster se establece en tipo entero (Entero16 por omisión). Los valores aleatorios de distribución binomial negativa se definen como números enteros positivos. Un ráster de punto flotante representará una conversión de valores enteros a punto flotante.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión deseada	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC destino	`TARGET_CRS`	[src] Predeterminado: SRC del proyecto	SRC de la capa ráster de salida
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 1.0	Tamaño del píxel (X=Y) en unidades de mapa.
Ráster de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos del ráster de salida	`OUTPUT_TYPE` Predeterminado: 0	[enumeración]	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Entero16 • 1 — Entero sin signo16 • 2 — Entero32 • 3 — Entero sin signo32 • 4 — Flotante32 • 5 — Flotante64
Parámetro de distribución k	`K_PARAMETER`	[numérico: entero] Predeterminado: 10	
Probabilidad	`PROBABILITY`	[numérico: doble] Predeterminado: 0.5	
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster]	Ráster que cubre la extensión deseada con el tamaño de celda lleno de valores distribuidos aleatoriamente

Código Python

Algoritmo ID: `native:createrandomnegativebinomialrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear una capa ráster aleatoria (distribución normal)

Genera una capa ráster para una extensión dada y un tamaño de celda lleno con valores aleatorios distribuidos normalmente.

Por omisión, los valores se elegirán con una media de 0.0 y una desviación estándar de 1.0. Esto se puede anular utilizando los parámetros avanzados para el valor medio y de desviación estándar. El tipo de datos ráster se establece en Flotante32 de forma predeterminada, los valores aleatorios de distribución normal son números de punto flotante.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión deseada	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC destino	`TARGET_CRS`	[src] Predeterminado: SRC del proyecto	SRC de la capa ráster de salida
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 1.0	Tamaño del píxel (X=Y) en unidades de mapa.
Ráster de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos del ráster de salida	`OUTPUT_TYPE` Predeterminado: 0	[enumeración]	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Flotante32 • 1 — Flotante64
Media de la distribución normal	`MEAN`	[numérico: doble] Predeterminado: 0.0	
Desviación estándar de la distribución normal	`STDDEV`	[numérico: doble] Predeterminado: 1.0	
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster]	Ráster que cubre la extensión deseada con el tamaño de celda lleno de valores distribuidos aleatoriamente

Código Python

Algoritmo ID: `native:createrandomnormalrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear una capa ráster aleatoria (distribución de poisson)

Genera una capa ráster para una extensión dada y un tamaño de celda lleno con valores aleatorios distribuidos con poisson.

Por defecto, los valores se elegirán con una media de 1.0. Esto se puede anular utilizando el parámetro avanzado para el valor medio. El tipo de datos ráster se establece en tipo entero (Entero16 por defecto). Los valores aleatorios de la distribución de Poisson son números enteros positivos. Un ráster de punto flotante representará una conversión de valores enteros a punto flotante.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión deseada	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC destino	`TARGET_CRS`	[src] Predeterminado: SRC del proyecto	SRC de la capa ráster de salida
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 1.0	Tamaño del píxel (X=Y) en unidades de mapa.
Ráster de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos del ráster de salida	`OUTPUT_TYPE` Predeterminado: 0	[enumeración]	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Entero16 • 1 — Entero sin signo16 • 2 — Entero32 • 3 — Entero sin signo32 • 4 — Flotante32 • 5 — Flotante64
Media	`MEAN`	[numérico: doble] Predeterminado: 1.0	
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster]	Ráster que cubre la extensión deseada con el tamaño de celda lleno de valores distribuidos aleatoriamente

Código Python

Algoritmo ID: `native:createrandompoissonrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear una capa ráster aleatoria (distribución uniforme)

Genera una capa ráster para una extensión dada y un tamaño de celda lleno con valores aleatorios.

Por omisión, los valores variarán entra el valor mínimo y el máximo para el tipo ráster de salida especificado. Esto se puede invalidar usando los parámetros avanzados para valores límite inferior y superior. Si los límites tienen el mismo valor o ambos son cero (por omisión) el algoritmo creará valores aleatorios en el rango completo del tipo de dato ráster elegido. Al elegir límites fuera del rango aceptable del tipo ráster de salida se abortará el algoritmo.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión deseada	`EXTENT`	[extensión]	Especifique la extensión espacial de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC destino	`TARGET_CRS`	[src] Predeterminado: SRC del proyecto	SRC de la capa ráster de salida
Tamaño de pixel	`PIXEL_SIZE`	[numérico: doble] Predeterminado: 1.0	Tamaño del píxel (X=Y) en unidades de mapa.
Ráster de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos del ráster de salida	`OUTPUT_TYPE` Predeterminado: 5	[enumeración]	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Byte • 1 — Entero16 • 2 — Entero sin signo16 • 3 — Entero32 • 4 — Entero sin signo32 • 5 — Flotante32 • 6 — Flotante64
Límite inferior para el rango de números aleatorios	`LOWER_BOUND`	[numérico: doble] Predeterminado: 0.0	
Límite superior para el rango de números aleatorios	`UPPER_BOUND`	[numérico: doble] Predeterminado: 0.0	
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster]	Ráster que cubre la extensión deseada con el tamaño de celda lleno de valores distribuidos aleatoriamente

Código Python

Algoritmo ID: `native:createrandomuniformrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.20 Análisis de terreno ráster

Aspecto

Calcula el aspecto del modelo de terreno digital en la entrada. La capa ráster de aspecto final contiene valores de 0 a 360 que expresan la dirección de la pendiente, comenzando desde el norte (0 °) y continuando en el sentido de las agujas del reloj.

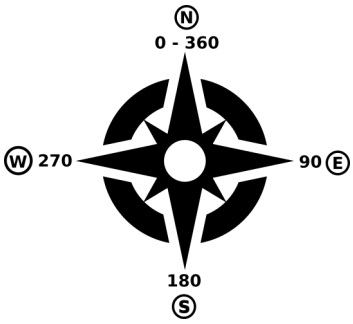


Figura 24.60: Valores de aspecto

La siguiente imagen muestra el aspecto de la capa reclasificada con una rampa de color:

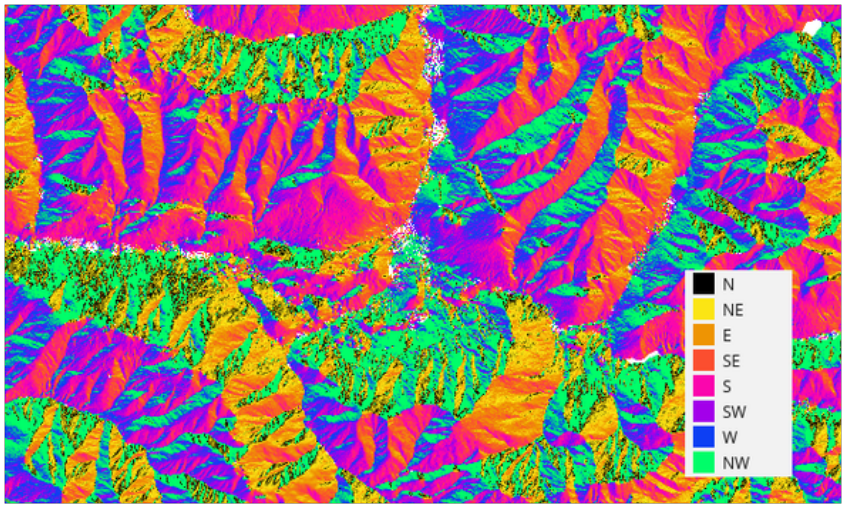


Figura 24.61: Aspecto de capa reclasificada

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de elevación	`INPUT`	[ráster]	capa ráster del Modelo Digital de Elevaciones
Factor Z	`Z_FACTOR`	[numérico: doble] Predeterminado: 1.0	Exageración vertical. Este parámetro es útil cuando las unidades Z difieren de las unidades X e Y, por ejemplo, pies y metros. Puede utilizar este parámetro para ajustarlo. El valor predeterminado es 1 (sin exagerar).

continúe en la próxima página

Tabla 24.416 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Aspecto	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de aspecto de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Aspecto	`OUTPUT`	[ráster]	La capa ráster de aspecto de salida

Código Python

Algorithm ID: `native:aspect`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Filtro MDT (basado en la pendiente)

Puede utilizarse para filtrar un modelo digital de elevación con el fin de clasificar sus celdas en celdas de suelo y de objeto (no de suelo).

La herramienta utiliza conceptos descritos por Vosselman (2000) y se basa en el supuesto de que es poco probable que una gran diferencia de altura entre dos celdas cercanas se deba a una pendiente pronunciada del terreno. La probabilidad de que la celda más alta no sea terreno aumenta cuando disminuye la distancia entre las dos celdas. Por lo tanto, el filtro define una diferencia de altura máxima (`dz_max`) entre dos celdas en función de la distancia (`d`) entre las celdas (`dz_max( d ) = d`). Una celda se clasifica como terreno si no hay ninguna celda dentro del radio del núcleo cuya diferencia de altura sea mayor que la diferencia de altura máxima permitida a la distancia entre estas dos celdas.

El parámetro de pendiente aproximada del terreno (`s`) se utiliza para modificar la función de filtro de forma que se ajuste a la pendiente general de la zona de estudio (`dz_max( d ) = d * s`). Se puede utilizar un intervalo de confianza del 5 % (`ci = 1,65 * sqrt( 2 * stddev )`) para modificar aún más la función de filtrado relajando (`dz_max( d ) = d * s + ci`) o amplificando (`dz_max( d ) = d * s - ci`) el criterio de filtrado.

Referencias: Vosselman, G. (2000): Slope based filtering of laser altimetry data. IAPRS, Vol. XXXIII, Part B3, Amsterdam, Países Bajos, 935-942.

Ver también

This tool is a port of the SAGA DTM Filter (slope-based)

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	capa ráster del Modelo Digital de Elevaciones
Número de banda	`BAND`	[banda ráster]	La banda del MDE a tener en cuenta
Radio del núcleo (píxeles)	`RADIUS`	[numérico: entero] Predeterminado: 5	El radio del núcleo del filtro (en píxeles). Debe ser lo suficientemente grande como para alcanzar las celdas de tierra junto a objetos no terrestres.
Pendiente del terreno (%, tamaño del píxel/unidades verticales).	`TERRAIN_SLOPE`	[numérico: doble] Predeterminado: 30,0	La pendiente aproximada del terreno en `%`. La pendiente del terreno debe ajustarse para tener en cuenta la relación entre las unidades de altura y las dimensiones en píxeles de la trama. Se utiliza para relajar el criterio de filtrado en terrenos más escarpados.
Modificación del filtro	`FILTER_MODIFIC`	[lista] Predeterminado: 0	Elija si desea aplicar el núcleo de filtro sin modificaciones o utilizar un intervalo de confianza para relajar o ampliar el criterio de altura. • 0 - Ninguna • 1 - Filtro de relajación • 2 - Amplificar
Desviación estándar	`STANDARD_DEVIA`	[numérico: doble] Predeterminado: 0.1	La desviación típica utilizada para calcular un intervalo de confianza del 5% aplicado al umbral de altura.
Capa de salida (suelo) Opcional	`OUTPUT_GROUND`	[ráster] Predeterinado: `[Save to temporary file]`	Especifica el MDE filtrado que contiene sólo las celdas clasificadas como suelo. *Uno de*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…
Capa de salida (objetos no terrestres) Opcional	`OUTPUT_NONGROU`	[ráster] Preestablecido: `[Omitir salida]`	Especifica los objetos no terrestres eliminados por el filtro. *Uno de*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida (suelo)	`OUTPUT_GROUND`	[ráster]	El MDE filtrado que contiene sólo las celdas clasificadas como suelo.
Capa de salida (objetos no terrestres)	`OUTPUT_NONGROU`	[ráster]	Los objetos no terrestres eliminados por el filtro.

Código Python

ID Algoritmo: `native:dtmslopebasedfilter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Fill sinks (Wang & Liu)

`NEW in 3.44`

Uses a method proposed by Wang & Liu to identify and fill surface depressions in digital elevation models.

The method was enhanced to allow the creation of hydrologically sound elevation models, i.e. not only to fill the depression(s) but also to preserve a downward slope along the flow path. If desired, this is accomplished by preserving a minimum slope gradient (and thus elevation difference) between cells.

References: Wang, L. & H. Liu (2006): An efficient method for identifying and filling surface depressions in digital elevation models for hydrologic analysis and modelling. International Journal of Geographical Information Science, Vol. 20, No. 2: 193-213.

Ver también

This tool is a port of the SAGA Fill Sinks (Wang & Liu) tool.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	capa ráster del Modelo Digital de Elevaciones
Número de banda	`BAND`	[banda ráster] Default: 1	La banda del MDE a tener en cuenta
Minimum slope (degrees)	`MIN_SLOPE`	[numérico: doble] Default: 0.10	Minimum slope gradient to preserve from cell to cell; with a value of zero, sinks are filled up to the spill elevation (which results in flat areas).
Output layer (filled DEM) Opcional	`OUTPUT_FILLED_`	[ráster] Predeterinado: `[Save to temporary file]`	Specify the ouput raster corresponding to the depression-free DEM. *One of*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…
Output layer (flow directions) Opcional	`OUTPUT_FLOW_DI`	[ráster] Preestablecido: `[Omitir salida]`	Specify the output raster with computed flow directions. *One of*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…
Output layer (watershed basins) Opcional	`OUTPUT_WATERSH`	[ráster] Preestablecido: `[Omitir salida]`	Specify the output raster corresponding to the delineated watershed basins. *One of*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Output layer (filled DEM)	`OUTPUT_FILLED_`	[ráster]	Ouput raster corresponding to the depression-free digital elevation model.
Output layer (flow directions)	`OUTPUT_FLOW_DI`	[ráster]	Output raster with computed flow directions; 0=N, 1=NE, 2=E, … 7=NW.
Output layer (watershed basins)	`OUTPUT_WATERSH`	[ráster]	Output raster corresponding to the delineated watershed basins.

Código Python

Algorithm ID: `native:fillsinkswangliu`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Sombreado

Calcula el sobreado de capa ráster dado un Modelo Digital del Terreno de entrada.

El sombreado de la capa se calcula de acuerdo con la posición del sol: tiene las opciones para cambiar tanto el ángulo horizontal (acimut) como el ángulo vertical (elevación del sol) del sol.

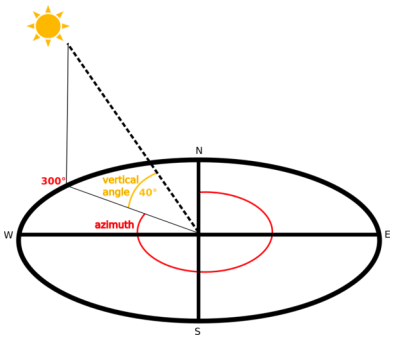


Figura 24.62: Azimut y ángulo vertical

La capa sombreada contiene valores desde 0 (completamente oscura) a 255 (completamente soleada). El sombreado se usa normalmente para entender mejor el relieve del área.

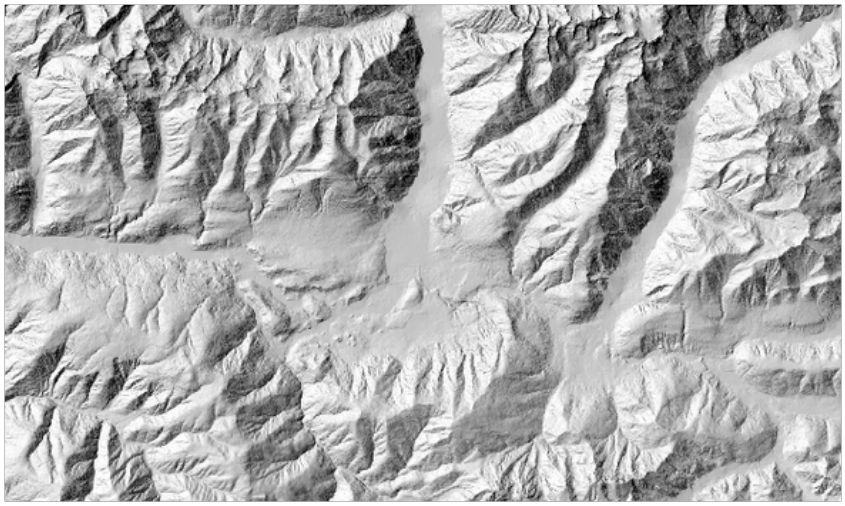


Figura 24.63: Capa de sobreado con azimut 300 y ángulo vertical de 45

Particularmente interesante es dar a la capa sombreada un valor de transparencia y superponerla con el ráster de elevación:

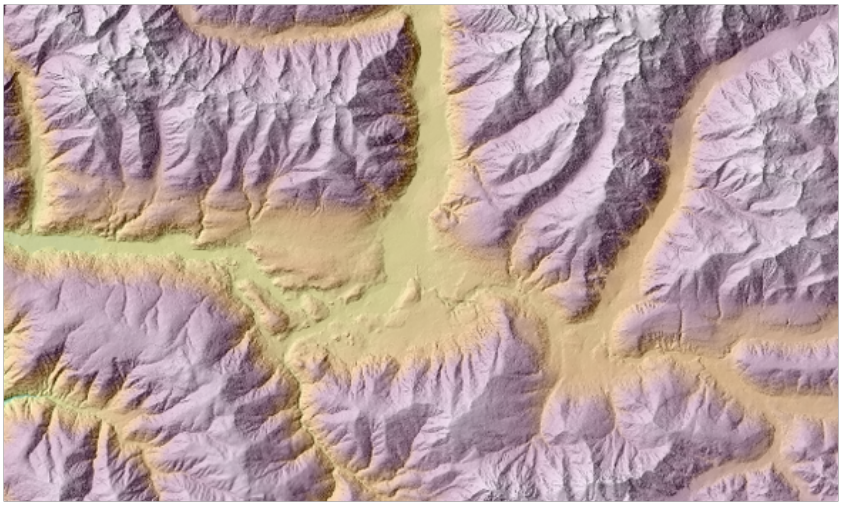


Figura 24.64: Superponer el sombreado con la capa de elevación

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de elevación	`INPUT`	[ráster]	capa ráster del Modelo Digital de Elevaciones
Factor Z	`Z_FACTOR`	[numérico: doble] Predeterminado: 1.0	Exageración vertical. Este parámetro es útil cuando las unidades Z difieren de las unidades X e Y, por ejemplo, pies y metros. Puede utilizar este parámetro para ajustarlo. Incrementar el valor de este parámetro exagerará el resultado final (lo hará parecer más «montañoso»). El valor predeterminado es 1 (sin exagerar).

continúe en la próxima página

Tabla 24.424 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Azimut (ángulo horizontal)	`AZIMUTH`	[numérico: doble] Predeterminado: 300.0	Establece el ángulo horizontal (en grados) del sol (sentido de las agujas del reloj). Rango: 0 a 360. 0 es el norte.
Ángulo vertical	`V_ANGLE`	[numérico: doble] Predeterminado: 40.0	Establece el ángulo vertical (en grados) del sol, que es la altura del sol. Los valores pueden ir de 0 (elevación mínima) a 90 (elevación máxima).
Sombreado	`OUTPUT`	[ráster] Predeterminado: `Guardar en archivo temporal`	Especifique la capa de trama de relieve sombreado de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Sombreado	`OUTPUT`	[ráster]	La capa ráster sombreada saliente

Código Python

Algorithm ID: `native:hillshade`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Curvas hipsométricas

Calcula curvas hipsométricas para un modelo de elevación digital de entrada. Las curvas se generan como archivos CSV en una carpeta de salida especificada por el usuario.

Una curva hipsométrica es un histograma acumulativo de valores de elevación en un área geográfica.

Puede utilizar curvas hipsométricas para detectar diferencias en el paisaje debido a la geomorfología del territorio.

Parametros

Etiqueta	Nombre	Tipo	Descripción
MDE a analizar	`INPUT_DEM`	[ráster]	Capa ráster del Modelo Digital del Terreno a usar para calcular altitudes
Capa límite	`BOUNDARY_LAYER`	[vectorial: poligonal]	Capa vectorial polígonal con límites de áreas utilizadas para calcular curvas hipsométricas

continúe en la próxima página

Tabla 24.426 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Paso	`STEP`	[numérico: doble] Predeterminado: 100.0	Distancia vertical entre curvas
Usar % del área en lugar de valor absoluto	`USE_PERCENTAGE`	[booleano] Predeterminado: Falso	Escribe el porcentaje de área en el campo «Área» del archivo CSV en lugar del área absoluta
Curvas hipsométricas	`OUTPUT_DIRECTO`	[carpeta]	Especifique la carpeta de salida de las curvas hipsométricas. *Una de*: • Guardar en Directorio Temporal • Guardar en Directorio

Salidas

Etiqueta	Nombre	Tipo	Descripción
Curvas hipsométricas	`OUTPUT_DIRECTO`	[carpeta]	Directorio que contiene los archivos con las curvas hipsométricas. Para cada entidad de la capa de vector de entrada, se creará un archivo CSV con valores de área y altitud. Los nombres de los archivos comienzan con `histogram_`, seguidos del nombre de la capa y el ID de la entidad.

	A	B
1	Area	Elevation
2	177475194.383	307
3	233206029.24	407
4	295553735.793	507
5	394718815.615	607
6	501801102.615	707
7	624399019.792	807
8	828877274.39	907
9	1042693465.68	1007
10	1277373021.81	1107
11	1556443975.41	1207
12	1888617494.27	1307
13	2248520437.31	1407
14	2627916813.17	1507
15	3010880212.04	1607
16	3411087555.34	1707

Código Python

Algoritmo ID: `qgis:hypsometriccurves`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Relieve

Crea una capa de relieve sombreada a partir de datos de elevación digitales. Puede especificar el color de relieve manualmente o puede dejar que el algoritmo elija automáticamente todas las clases de relieve.

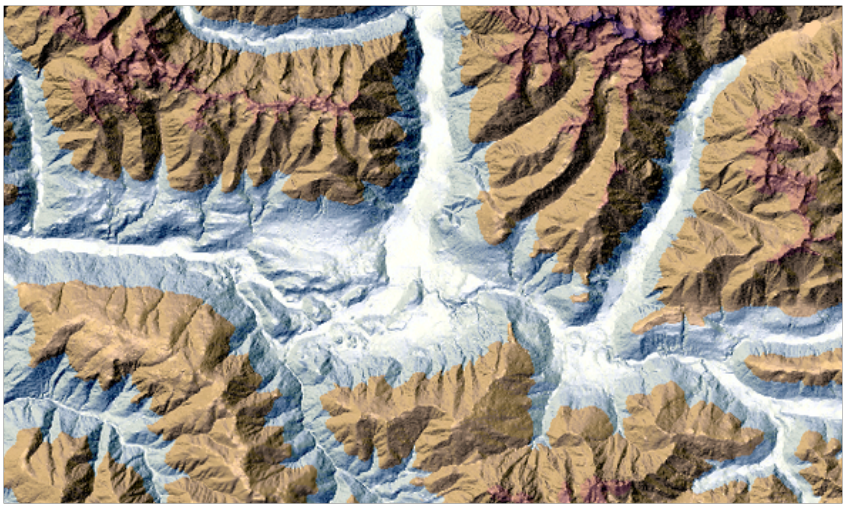


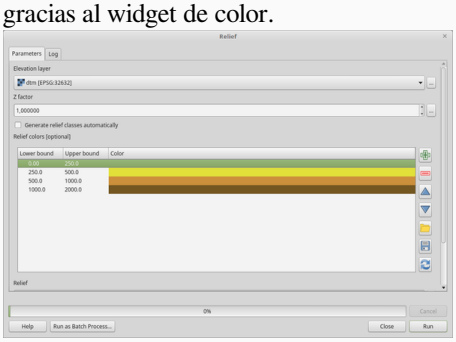
Figura 24.65: Capa de relieve

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de elevación	`INPUT`	[ráster]	capa ráster del Modelo Digital de Elevaciones
Factor Z	`Z_FACTOR`	[numérico: doble] Predeterminado: 1.0	Exageración vertical. Este parámetro es útil cuando las unidades Z difieren de las unidades X e Y, por ejemplo, pies y metros. Puede utilizar este parámetro para ajustarlo. Incrementar el valor de este parámetro exagerará el resultado final (lo hará parecer más «montañoso»). El valor predeterminado es 1 (sin exagerar).
Generar clases de relieve automáticamente	`AUTO_COLORS`	[booleano] Predeterminado: Falso	Si marcas esta opción el algoritmo creará todas las clases de color automáticamente

continúe en la próxima página

Tabla 24.428 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Colores de relieve Opcional	`COLORS`	[table widget]	Utilice el widget de tabla si desea elegir los colores de relieve manualmente. Puede agregar tantas clases de color como desee: para cada clase puede elegir el límite inferior y superior y, finalmente, haciendo clic en la fila de colores, puede elegir el color gracias al widget de color.  Figura 24.66: Ajuste manual de las clases de color del relieve Los botones en el panel lateral derecho le dan la oportunidad de: agregar o quitar clases de color, cambiar el orden de las clases de color ya definidas, abrir un archivo existente con clases de color y guardar las clases actuales como archivo.
Relieve	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa raster de relieve de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…
Distribución de frecuencias Opcional	`FREQUENCY_DIST`	[vector: tabla] Preestablecido: `[Omitir salida]`	Especifique la tabla CSV para la distribución de frecuencias de salida. *Una de*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Relieve	`OUTPUT`	[ráster]	La capa ráster saliente de relieve
Distribución de frecuencias	`OUTPUT`	[vector: tabla]	La distribución de frecuencias saliente

Código Python

Algoritmo ID: `qgis:relief`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Índice de aspereza

Calcula la medición cuantitativa de la heterogeneidad del terreno descrita por Riley et al. (1999). Se calcula para cada ubicación, resumiendo el cambio de elevación dentro de la cuadrícula de 3x3 píxeles.

Cada pixel contiene la diferencia en elevación desde una celda central y sus 8 celdas de alrededor.

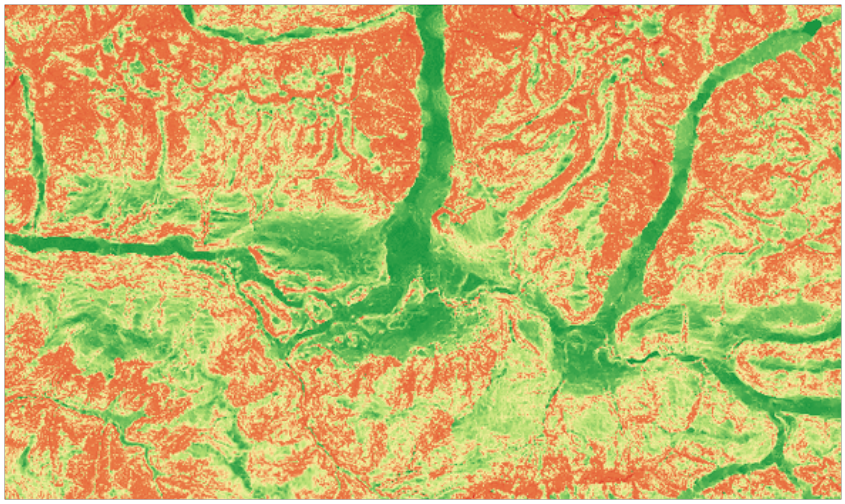


Figura 24.67: Capa de aspereza de valores bajos (rojo) a valores altos (verde)

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de elevación	`INPUT`	[ráster]	capa ráster del Modelo Digital de Elevaciones
Factor Z	`Z_FACTOR`	[numérico: doble] Predeterminado: 1.0	Exageración vertical. Este parámetro es útil cuando las unidades Z difieren de las unidades X e Y, por ejemplo, pies y metros. Puede utilizar este parámetro para ajustarlo. Incrementar el valor de este parámetro exagerará el resultado final (haciéndolo parecer más resistente). El valor predeterminado es 1 (sin exagerar).

continúe en la próxima página

Tabla 24.430 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Aspereza	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de rugosidad de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Aspereza	`OUTPUT`	[ráster]	La capa ráster saliente de aspereza

Código Python

Algorithm ID: `native:ruggednessindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Pendiente

Calcula la pendiente a partir de una capa ráster de entrada. La pendiente es el ángulo de inclinación del terreno y se expresa en **grados**.

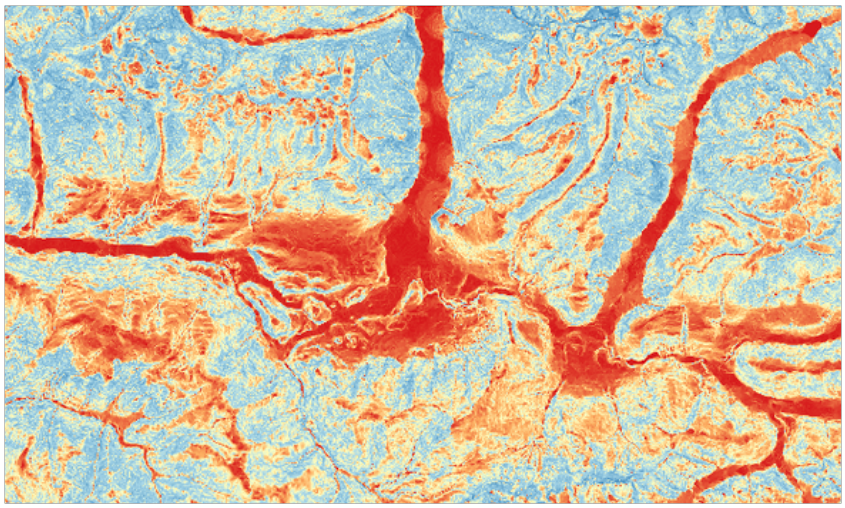


Figura 24.68: Áreas planas en rojo, áreas empinadas en azul

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de elevación	`INPUT`	[ráster]	capa ráster del Modelo Digital de Elevaciones
Factor Z	`Z_FACTOR`	[numérico: doble] Predeterminado: 1.0	Exageración vertical. Este parámetro es útil cuando las unidades Z difieren de las unidades X e Y, por ejemplo, pies y metros. Puede utilizar este parámetro para ajustarlo. Incrementar el valor de este parámetro exagerará el resultado final (haciéndolo más empinado). El valor predeterminado es 1 (sin exagerar).
Pendiente	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa raster de pendiente de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Pendiente	`OUTPUT`	[ráster]	La capa ráster saliente de pendientes

Código Python

Algorithm ID: `native:slope`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.21 Herramientas de ráster

Alinear ráster

Alinea el ráster remuestreándolo al mismo tamaño de celda y reproyectándolo al mismo SRC que un ráster de referencia.

> **Advertencia**
>
> **Este algoritmo SOLO está disponible en el contexto** *Model Designer*. Para otros contextos, utilice en su lugar *Alinear rásters*.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de entrada para alinear
Método de remuestreo	`RESAMPLING_MET`	[enumeración] Predeterminado: 0	Method to use for input layer resampling. `Available options` are: • 0 — Vecino más próximo • 1 — Bilineal (núcleo 2x2) • 2 — Cúbico (núcleo 4x4) • 3 — B-Spline cúbica (núcleo 4x4) • 4 — Lanczos (núcleo 6x6) • 5 — Promedio • 6 — Modo • 7 — Máximo • 8 — Mínimo • 9 — Mediana • 10 — Primer cuartil (Q1) • 11 — Tercer cuartil (Q3)
Reduzca la escala de los valores según el tamaño de la celda.	`RESCALE`	[booleano] Predeterminado: No	
Capa de referencia	`REFERENCE_LAYE`	[ráster]	Una capa ráster que se utilizará para obtener la extensión, el tamaño de celda y el SRC que se aplicarán a las capas de entrada.
Anular referencia SRC Opcional	`CRS`	[src]	SRC que se utilizará en lugar de la capa de referencia
Anular tamaño de celda de referencia X Opcional	`CELL_SIZE_X`	[numérico: doble]	Tamaño de celda en dirección X que se utilizará en lugar del de la capa de referencia
Anular tamaño de celda de referencia Y Opcional	`CELL_SIZE_Y`	[numérico: doble]	Tamaño de celda en dirección Y que se utilizará en lugar del de la capa de referencia
Desplazamiento de la cuadrícula de referencia X Opcional	`GRID_OFFSET_X`	[numérico: doble]	Desplazamiento en dirección X para aplicar a la cuadrícula de celdas
Anular desplazamiento de cuadrícula de referencia Y Opcional	`GRID_OFFSET_Y`	[numérico: doble]	Desplazamiento en dirección Y para aplicar a la cuadrícula de celdas

continúe en la próxima página

Tabla 24.434 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Cortar por extensión Opcional	`EXTENT`	[extensión]	Especifique la extensión de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Ráster alineado	`OUTPUT`	[ráster]	ráster de salida con píxeles remuestreados

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster alineado	`OUTPUT`	[ráster]	ráster de salida con píxeles remuestreados

Código Python

ID Algoritmo: `native:alignsingleraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Alinear rásters

Alinea los rásteres remuestreándolos al mismo tamaño de celda y reproyectándolos al mismo SRC que un ráster de referencia.

> **Advertencia**
>
> **Este algoritmo NO está disponible en el contexto** *Diseñador de modelo*. Utilice en su lugar *Alinear ráster*.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Lista de capas ráster de entrada con opciones de remuestreo asociadas (rellenada como un elemento `QgsProcessingParameterAlignRasterLayers` — hecho en GUI pulsando el botón *Configurar Ráster…* para cada capa seleccionada): **Capa de entrada [cadena] (`archivo de entrada`)** Ruta completa de la capa de entrada a alinear **Archivo de salida [cadena] (`archivo de salida`)** Ruta completa de la capa de salida alineada correspondiente **Método de remuestreo [enumeración] (`resampleMethod`)** Method to use for input layer resampling. `Available options` are: • 0 — Vecino más próximo • 1 — Bilineal (núcleo 2x2) • 2 — Cúbico (núcleo 4x4) • 3 — B-Spline cúbica (núcleo 4x4) • 4 — Lanczos (núcleo 6x6) • 5 — Promedio • 6 — Modo • 7 — Máximo • 8 — Mínimo • 9 — Mediana • 10 — Primer cuartil (Q1) • 11 — Tercer cuartil (Q3) **Redimensionar los valores según el tamaño de la celda** [booleano] (`rescale`)
Capa de referencia	`REFERENCE_LAYE`	[ráster]	Una capa ráster que se utilizará para obtener la extensión, el tamaño de celda y el SRC que se aplicarán a las capas de entrada.
Anular referencia SRC Opcional	`CRS`	[src]	SRC que se utilizará en lugar de la capa de referencia

continúe en la próxima página

Tabla 24.436 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Anular tamaño de celda de referencia X Opcional	`CELL_SIZE_X`	[numérico: doble]	Tamaño de celda en dirección X que se utilizará en lugar del de la capa de referencia
Anular tamaño de celda de referencia Y Opcional	`CELL_SIZE_Y`	[numérico: doble]	Tamaño de celda en dirección Y que se utilizará en lugar del de la capa de referencia
Desplazamiento de la cuadrícula de referencia X Opcional	`GRID_OFFSET_X`	[numérico: doble]	Desplazamiento en dirección X para aplicar a la cuadrícula de celdas
Anular desplazamiento de cuadrícula de referencia Y Opcional	`GRID_OFFSET_Y`	[numérico: doble]	Desplazamiento en dirección Y para aplicar a la cuadrícula de celdas
Cortar por extensión Opcional	`EXTENT`	[extensión]	Especifique la extensión de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`

Salidas

Etiqueta	Nombre	Tipo	Descripción
Rásters alineados	`OUTPUT_LAYERS`	[ráster] [lista]	rásters de salida con píxeles remuestreados

Código Python

ID Algoritmo: `native:alignrasters`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Convertir mapa a ráster

Crea una imagen ráster con el contenido del mapa del lienzo.

Un *tema de mapa* se puede seleccionar para renderizar un conjunto predeterminado de capas con un estilo definido para cada capa.

Alternativamente, se puede seleccionar una sola capa si no se establece ningún tema de mapa.

Si no se configura ni el tema ni la capa del mapa, se representará el contenido del mapa actual. La extensión mínima ingresada se extenderá internamente para ser un múltiplo del tamaño del mosaico.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Extensión mínima a renderizar (xmin, xmax, ymin, ymax)	`EXTENT`	[extensión]	Especifique la extensión de la capa ráster de salida. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Tamaño de tesela	`TILE_SIZE`	[numérico: entero] Predeterminado: 1024	Tamaño de tesela de la capa ráster de salida. Valor mínimo: 64.
Unidades de mapa por pixel	`MAP_UNITS_PER_`	[numérico: doble] Predeterminado: 100.0	Tamaño de pixel (en unidades de mapa). Valor mínimo: 0.0

continúe en la próxima página

Tabla 24.438 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Hacer fondo transparente	`MAKE_BACKGROUN`	[booleano] Predeterminado: Falso	Permite exportar el mapa con fondo transparente. Emite una imagen RGBA (en lugar de RGB) si se establece en `True`.
Tema de mapa a representar Opcional	`MAP_THEME`	[enumeración]	Usar un *tema de mapa* existente para la representación.
Capa simple a representar Opcional	`LAYER`	[enumeración]	Elige una capa simple para la representación
Capa de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[ráster]	Capa ráster saliente

Código Python

ID de Algoritmo: `native:rasterize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Rellenar celdas sin datos

Restablece los valores sin datos en el ráster de entrada a un valor elegido, lo que da como resultado un conjunto de datos ráster con píxeles sin datos.

El algoritmo respeta el tipo de datos ráster de entrada, por ejemplo un valor de relleno de punto flotante se truncará cuando se aplique a un ráster entero.

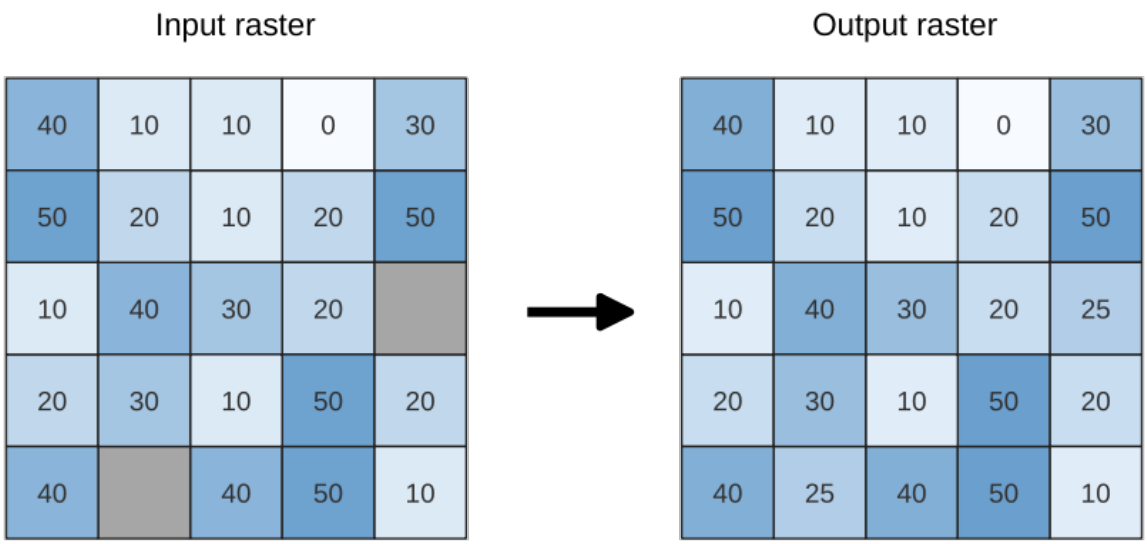


Figura 24.69: Relleno de valores sin datos (en gris) de un ráster

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Ráster de Entrada	`INPUT`	[ráster]	El ráster para procesar.
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	La banda del ráster
Valor de relleno	`FILL_VALUE`	[numérico: doble] Predeterminado: 1.0	Establezca el valor a usar para los píxeles sin datos
Ráster de salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ráster de salida	`OUTPUT`	[ráster]	La capa ráster de salida con celdas de datos rellenas.

Código Python

Algoritmo ID: `native:fillnodata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Generar teselas XYZ (Directorio)

Genera teselas ráster “XYZ” usando el proyecto actual de QGIS como imágenes individuales en una estructura de directorio.

Opcionalmente, se puede crear un archivo de salida Leaflet HTML utilizando los mosaicos generados como capa cartográfica.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extesión (xmin, xmax, ymin, ymax)	`EXTENT`	[extensión]	Especifique la extensión de las teselas. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Zoom mínimo	`ZOOM_MIN`	[numérico: entero] Predeterminado: 12	Mínimo 0, máximo 25.
Zoom máximo	`ZOOM_MAX`	[numérico: entero] Predeterminado: 12	Mínimo 0, máximo 25.
PPP	`DPI`	[numérico: entero] Predeterminado: 96	Mínimo 48, máximo 600.
Color de fondo Opcional	`BACKGROUND_COL`	[color] Predeterminado: QColor(0, 0, 0, 0)	Elige el color de fondo para las teselas
Habilitar antialiasing	`ANTIALIAS`	[booleano] Predeterminado: Verdadero	Determina si se debe activar el antialiasing
Formato de tesela	`TILE_FORMAT`	[enumeración] Predeterminado: 0	Una de: • 0 — PNG • 1 — JPG
Calidad (solo JPG) Opcional	`QUALITY`	[numérico: entero] Predeterminado: 75	Mínimo 1, máximo 100.
Tamaño de Metatesela Opcional	`METATILESIZE`	[numérico: entero] Predeterminado: 4	Especifique un tamaño de metatile personalizado al generar mosaicos XYZ. Los valores más grandes pueden acelerar la representación de mosaicos y proporcionar un mejor etiquetado (menos espacios sin etiquetas) a expensas de utilizar más memoria. Mínimo 1, máximo 20.
Anchura de tesela Opcional	`TILE_WIDTH`	[numérico: entero] Predeterminado: 256	Mínimo 1, máximo 4096.
Altura de tesela Opcional	`TILE_HEIGHT`	[numérico: entero] Predeterminado: 256	Mínimo 1, máximo 4096.

continúe en la próxima página

Tabla 24.443 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Usar eje Y de tesela invertido (convención TMS)	`TMS_CONVENTION`	[booleano] Predeterminado: Falso	
Directorio de salida	`OUTPUT_DIRECTO`	[carpeta] Predeterminado: `[Guardar en directorio temporal]`	Especificación del directorio de salida (para los mosaicos). *Uno de*: • Saltar salida • Guardar en Directorio Temporal • Guardar en Directorio
Salida html (folleto) Opcional	`OUTPUT_HTML`	[html] Predeterinado: `[Save to temporary file]`	Especificación del archivo HTML de salida. *Uno de*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Título de la salida HTML de Leaflet Opcional	`HTML_TITLE`	[cadena de texto] Predeterminado: No establecido	HTML <title>-etiqueta utilizada para el archivo de salida HTML de Leaflet.
Atribución de la salida HTML de Leaflet Opcional	`HTML_ATTRIBUTI`	[cadena de texto] Predeterminado: No establecido	Atribución de mapa personalizada utilizada para el archivo de salida HTML de Leaflet. Son posibles los enlaces HTML.
Incluir el mapa base de OpenStreetMap en la salida HTML de Leaflet	`HTML_OSM`	[booleano] Predeterminado: Falso	Se incluye una capa de mapa base de OpenStreetMap (fuente: https://tile.openstreetmap.org) en el archivo de salida HTML de Leaflet. La atribución correcta del mapa se añade automáticamente.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Directorio de salida	`OUTPUT_DIRECTO`	[carpeta]	Directorio de salida (para las teselas)
Salida html (folleto)	`OUTPUT_HTML`	[html]	El archivo saliente HTML (folleto)

Código Python

ID Algoritmo: `native:tilesxyzdirectory`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Generar teselas XYZ (MBTiles)

Genera mosaicos ráster "XYZ" utilizando el proyecto actual de QGIS como un solo archivo en el formato "MBTiles".

Parametros

Etiqueta	Nombre	Tipo	Descripción
Extesión (xmin, xmax, ymin, ymax)	`EXTENT`	[extensión]	Especifique la extensión de las teselas. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Zoom mínimo	`ZOOM_MIN`	[numérico: entero] Predeterminado: 12	Mínimo 0, máximo 25.
Zoom máximo	`ZOOM_MAX`	[numérico: entero] Predeterminado: 12	Mínimo 0, máximo 25.
PPP	`DPI`	[numérico: entero] Predeterminado: 96	Mínimo 48, máximo 600.
Color de fondo Opcional	`BACKGROUND_COL`	[color] Predeterminado: QColor(0, 0, 0, 0)	Elige el color de fondo para las teselas
Habilitar antialiasing	`ANTIALIAS`	[booleano] Predeterminado: Verdadero	Determina si se debe activar el antialiasing

continúe en la próxima página

Tabla 24.446 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Formato de tesela	`TILE_FORMAT`	[enumeración] Predeterminado: 0	Una de: • 0 — PNG • 1 — JPG
Calidad (solo JPG) Opcional	`QUALITY`	[numérico: entero] Predeterminado: 75	Mínimo 1, máximo 100.
Tamaño de Metatesela Opcional	`METATILESIZE`	[numérico: entero] Predeterminado: 4	Especifique un tamaño de metatile personalizado al generar mosaicos XYZ. Los valores más grandes pueden acelerar la representación de mosaicos y proporcionar un mejor etiquetado (menos espacios sin etiquetas) a expensas de utilizar más memoria. Mínimo 1, máximo 20.
Archivo saliente (para MBTiles)	`OUTPUT_FILE`	[archivo] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida. *Uno de*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo saliente (para MBTiles)	`OUTPUT_FILE`	[archivo]	El archivo saliente.

Código Python

ID Algoritmo: `native:tilesxyzmbtiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.22 Análisis vectorial

Estadísticas básicas para campos

Genera estadísticas básicas para un campo de la tabla de atributos de una capa vectorial. Se admiten campos numéricos, de fecha, de hora y de cadena. Las estadísticas devueltas dependerán del tipo de campo.

Las estadísticas se pueden generar en forma de tabla o archivo HTML y están disponibles en *Procesamiento* ► *Visor de resultados*.

Menú predeterminado: *Vectorial* ► *Herramientas de análisis*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Vectorial de entrada	`INPUT_LAYER`	[vectorial: cualquiera]	Capa vectorial sobre la cuál calcular las estadíticas
Campo para calcular las estadísticas	`FIELD_NAME`	[campo de tabla: cualquiera]	Cualquier campo de la tabla soportado para calcular las estadísticas
Estadísticas Opcional	`OUTPUT`	[vector: tabla] Predeterminado: `[Crear capa temporal]`	Especifique la tabla de salida para las estadísticas generadas. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Informe estadístico Opcional	`OUTPUT_HTML_FI`	[html] Predeterinado: `[Save to temporary file]`	Especificación del archivo para las estadísticas calculadas. *Uno de*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Estadísticas	`OUTPUT`	[vector: tabla]	Tabla que contiene las estadísticas calculadas
Informe estadístico	`OUTPUT_HTML_FI`	[html]	Archivo HTML con las estadísticas calculadas
Recuento	`COUNT`	[numérico: entero]	
Número de valores únicos	`UNIQUE`	[numérico: entero]	
Número de valores vacíos (nulos)	`EMPTY`	[numérico: entero]	
Número de valores no vacíos	`FILLED`	[numérico: entero]	
Valor mínimo	`MIN`	[la misma que la entrada]	
Valor máximo	`MAX`	[la misma que la entrada]	
Longitud mínima	`MIN_LENGTH`	[numérico: entero]	

continúe en la próxima página

Tabla 24.449 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Longitud máxima	`MAX_LENGTH`	[numérico: entero]	
Longitud media	`MEAN_LENGTH`	[numérico: doble]	
Coeficiente de variación	`CV`	[numérico: doble]	
Suma	`SUM`	[numérico: doble]	
Valor promedio	`MEAN`	[numérico: doble]	
Desviación estándar	`STD_DEV`	[numérico: doble]	
Rango	`RANGE`	[numérico: doble]	
Mediana	`MEDIAN`	[numérico: doble]	
Minoría (valor de aparición mas rara)	`MINORITY`	[la misma que la entrada]	
Mayoría (valor que sucede mas frecuentemente)	`MAJORITY`	[la misma que la entrada]	
Primer Cuartil	`FIRSTQUARTILE`	[numérico: doble]	
Tercer cuartil	`THIRDQUARTILE`	[numérico: doble]	
Rango Intercuartil (IQR)	`IQR`	[numérico: doble]	

Código Python

Algorithm ID: `native:basicstatisticsforfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ascenso a lo largo de la línea

Calcula el ascenso y descenso total a lo largo de geometrías lineales. La capa de entrada debe tener valores Z presentes. Si los valores Z no están disponibles, el algoritmo *Drapeado (establecer el valor Z del ráster)* se puede usar para agregar valores Z de una capa MDE.

La capa de salida es una copia de la capa de entrada con campos adicionales que contienen el ascenso total (`climb`), el descenso total (`descent`), la elevación mínima (`minelev`) y la elevación máxima (`maxelev`) para cada geometría de línea. Si la capa de entrada contiene campos con los mismos nombres que estos campos agregados, se les cambiará el nombre (los nombres de los campos se modificarán a «name_2», «name_3», etc., encontrando el primer nombre no duplicado).

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa lineal	`INPUT`	[vectorial: lineal]	Capa lineal sobre la cual calcular el ascenso. Debe tener valores Z

continúe en la próxima página

Tabla 24.450 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Capa de ascenso	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especificación de la capa de salida (línea). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de ascenso	`OUTPUT`	[vectorial: lineal]	Capa lineal contenedora de nuevos atributos con los resultados para los cálculos de ascenso.
Ascenso total	`TOTALCLIMB`	[numérico: doble]	La suma del ascenso para todas las geometrías lineales en la capa de entrada
*Descenso total**	`TOTALDESCENT`	[numérico: doble]	La suma del descenso para todas las geometrías lineales en la capa de entrada
Elevación mínima	`MINELEVATION`	[numérico: doble]	La elevación mínima para las geometrías en la capa
Elevación máxima	`MAXELEVATION`	[numérico: doble]	La elevación máxima para las geometrías de la capa

Código Python

Algorithm ID: `native:climbalongline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Recuento de puntos en polígono

Toma una capa de puntos y una poligonal y cuenta el número de puntos de la capa de puntos en cada uno de los polígonos de la capa poligonal.

Se genera una nueva capa poligonal, con exactamente el mismo contenido que la capa poligonal entrante, pero conteniendo un camp adicional con el recuento de puntos que corresponden a cada polígono.

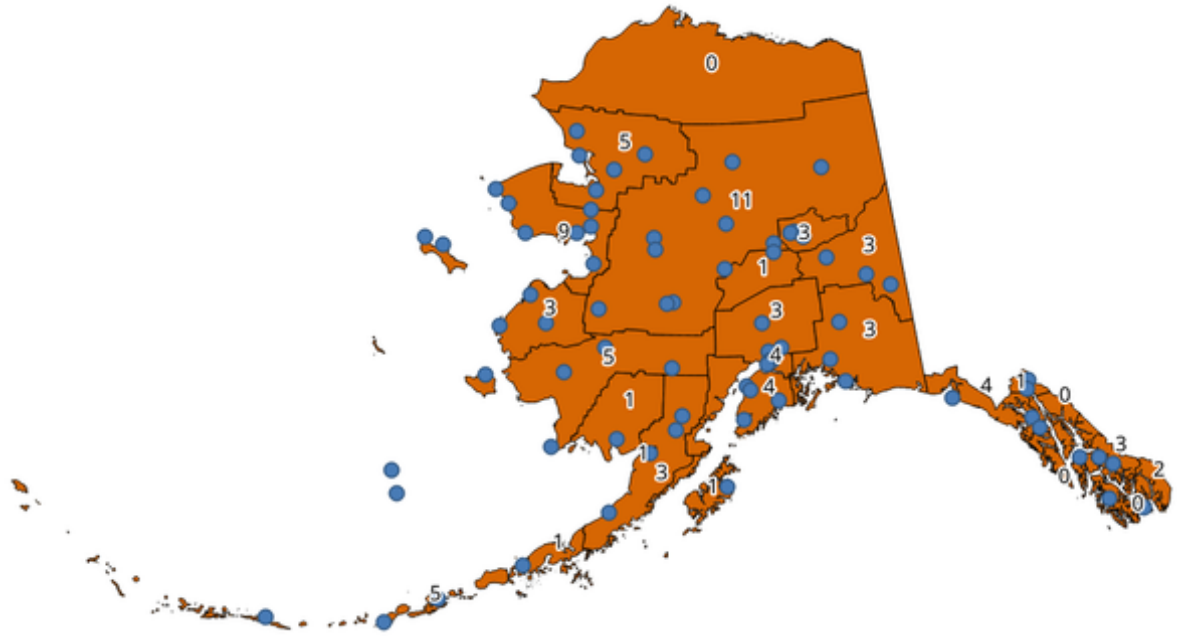


Figura 24.70: Las etiquetas en los polígonos muestran el recuento de puntos

Se puede utilizar un campo de peso opcional para asignar pesos a cada punto. Alternativamente, se puede especificar un campo de clase único. Si se utilizan ambas opciones, el campo de ponderación tendrá prioridad y el campo de clase única se ignorará.

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

`Menú predeterminado`: *Vectorial* ► *Herramientas de Análisis*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Polígonos	`POLYGONS`	[vectorial: poligonal]	Capa poligonal cuyas entidades son asociadas con el número de puntos que contienen
Puntos	`POINTS`	[vector: punto]	Capa de puntos con objetos a contar
Campo de peso Opcional	`WEIGHT`	[campo de tabla: cualquiera]	Un campo de la capa de puntos. El recuento generado será la suma del campo de pesos de los puntos que contiene el polígono. Si el campo de peso no es numérico, el recuento será `0`.
Campo clase Opcional	`CLASSFIELD`	[campo de tabla: cualquiera]	Los puntos se clasifican según el atributo seleccionado y si varios puntos con el mismo valor de atributo están dentro del polígono, solo se cuenta uno de ellos. El conteo final de los puntos en un polígono es, por lo tanto, el conteo de las diferentes clases que se encuentran en él.
Nombre del campo Recuento	`FIELD`	[cadena de texto] Predeterminado: "NUMPOINTS"	El nombre del campo donde almacenar el número de puntos

continúe en la próxima página

Tabla 24.452 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Recuento	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especificación de la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Recuento	`OUTPUT`	[vectorial: poligonal]	Capa resultante con la tabla de atributos conteniendo la nueva columna con el recuento de puntos

Código Python

Algoritmo ID: `native:countpointsinpolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Clustering o agrupamiento DBSCAN

Los clústeres señalan características basadas en una implementación 2D del algoritmo de agrupamiento espacial de aplicaciones con ruido (DBSCAN) basado en densidad.

El algoritmo requiere dos parámetros, un tamaño mínimo de grupo o cluster, y la distancia máxima entre puntos agrupados.

Ver también

Agrupación ST-DBSCAN, *Agrupación de K-medias*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	Capa a analizar
Tamaño mínimo de clúster	`MIN_SIZE`	[numérico: entero] Predeterminado: 5	Número mínimo de objetos para generar un grupo
Distancia máxima entre puntos agrupados	`EPS`	[numérico: doble] Predeterminado: 1.0	Distancia más allá de la cual dos entidades no pueden pertenecer al mismo grupo (eps)
Clusters	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial para el resultado de la agrupación. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tratar los puntos fronterizos como ruido (DBSCAN*)	`DBSCAN*`	[booleano] Predeterminado: Falso	Si se marca, los puntos en el borde de un grupo se tratan como puntos no agrupados, y solo los puntos en el interior de un grupo se etiquetan como agrupados.
Nombre de campo de grupo	`FIELD_NAME`	[cadena de texto] Predeterminado: “CLUSTER_ID”	Nombre del campo donde se almacenará el número de grupo asociado
Nombre de campo de tamaño de clúster	`SIZE_FIELD_NAM`	[cadena de texto] Predeterminado: “CLUSTER_SIZE”	Nombre del campo con el recuento de entidades en el mismo clúster

Salidas

Etiqueta	Nombre	Tipo	Descripción
Clusters	`OUTPUT`	[vector: punto]	Capa vectorial que contiene las entidades originales con un campo que establece el grupo al que pertenecen
Número de grupos	`NUM_CLUSTERS`	[numérico: entero]	El número de grupos descubiertos

Código Python

Algoritmo ID: `native:linesubstring`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Matriz distancia

Calcula distancias de entidades puntuales a las entidades más cercanas en la misma capa o en otra capa.

Menú predeterminado: *Vectorial ► Herramientas de análisis*

Ver también

Unir atributos por proximidad

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada de puntos	`INPUT`	[vector: punto]	Capa de puntos para la cuál se calcula la matriz de distancia (**desde** puntos)
Campo único ID de entrada	`INPUT_FIELD`	[campo de tabla: cualquiera]	Campo que se utilizará para identificar de únicamente las entidades de la capa de entrada. Usado en la tabla de atributos de salida.
Capa de puntos objetivo	`TARGET`	[vector: punto]	Capa de puntos contenedora del punto(s) mas cercano a buscar(**a** puntos)
Campo único ID objetivo	`TARGET_FIELD`	[campo de tabla: cualquiera]	Campo a usar para identificar únicamente objetos de la capa objetivo. Usado en la tabla de atributos.
Tipo de matriz saliente	`MATRIX_TYPE`	[enumeración] Predeterminado: 0	Diferentes tipos de cálculo están disponibles: • 0 — Matriz de distancia lineal (N * * k * x 3): para cada punto de entrada, informa la distancia a cada uno de los *k* puntos objetivo más cercanos. La matriz de salida consta de hasta *k* filas por punto de entrada, y cada fila tiene tres columnas: *InputID*, *TargetID* y *Distance*. • 1 — Matriz de distancia estándar (N x T) • 2 — Matriz de resumen de distancias (media, desv. Estándar, mín., máx.): Para cada punto de entrada, informa estadísticas sobre las distancias a sus puntos objetivo.

continúe en la próxima página

Tabla 24.457 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Usar solo los puntos objetivos (k) mas cercanos	`NEAREST_POINTS`	[numérico: entero] Predeterminado: 0	Puede elegir calcular la distancia a todos los puntos en la capa de destino (*0*) o limitar a un número (*k*) de entidades más cercanas.
Matriz de distancia	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especificación de la capa vectorial de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Matriz de distancia	`OUTPUT`	[vector: punto]	Punto (o multipunto para el caso «Lineal (N * * k * x 3)») capa vectorial que contiene el cálculo de la distancia para cada entidad de entrada. Sus entidades y tabla de atributos dependen del tipo de matriz de salida seleccionado.

Código Python

Algoritmo ID: `qgis:distancematrix`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Distancia al centro más cercano (línea a centro)

Crea líneas que unen cada entidad de un vector de entrada a la entidad más cercana en una capa de destino. Las distancias se calculan en base al *centro* de cada entidad.

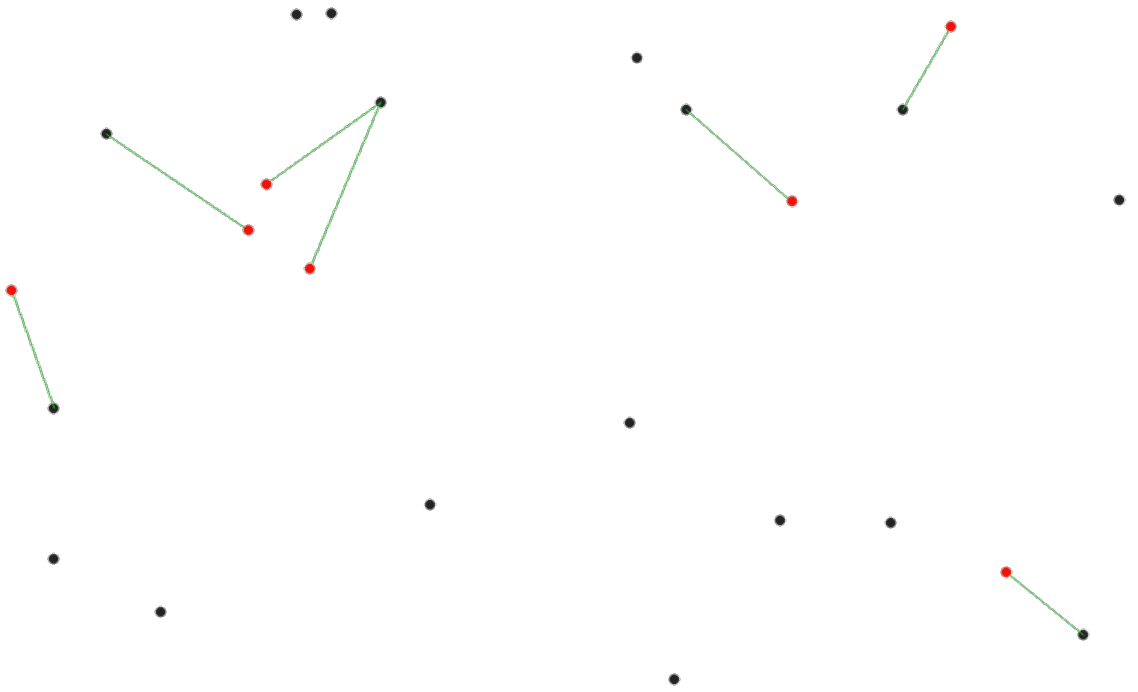


Figura 24.71: Mostrar el centro mas cercano para las entidades rojas entrantes

Ver también

Distancia al centro mas cercano (puntos), *Unir atributos por proximidad*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de puntos fuente	`INPUT`	[vector: geometría]	Capa vectorial para la cual se busca el objeto mas cercano
Capa destino de centros	`HUBS`	[vector: geometría]	Capa vectorial que contiene las entidades para buscar
Nombre de atributo de capa de centro	`FIELD`	[campo de tabla: cualquiera]	Campo a usar para identificar unívocamente entidades de la capa de destino. Usado en la tabla de atributos saliente
Unidad de medida	`UNIT`	[enumeración] Predeterminado: 0	Unidades en las cuales informar de la distancia a la entidad mas cercana: • 0 — Metros • 1 — Pie • 2 — Millas • 3 — Kilometros • 4 — Unidades de capa

continúe en la próxima página

Tabla 24.459 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Distancia al centro	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de línea de salida que conecta los puntos coincidentes. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Distancia al centro	`OUTPUT`	[vectorial: lineal]	Capa vectorial lineal con los atributos de las entidades enteras, el identificador de su entidad mas cercana y la distancia calculada.

Código Python

Algoritmo ID: `qgis:distancetonearesthublinetohub`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Distancia al centro mas cercano (puntos)

Crea una capa de puntos representando el *centro* de las entidades entrantes con dos campos añadidos conteniendo el identificador del objeto mas cercano (basado en su ponto central) y la distancia entre los puntos.

Ver también

Distancia al centro más cercano (línea a centro), *Unir atributos por proximidad*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de puntos fuente	`INPUT`	[vectorial: cualquiera]	Capa vectorial para la cual se busca el objeto mas cercano
Capa destino de centros	`HUBS`	[vectorial: cualquiera]	Capa vectorial que contiene las entidades para buscar
Nombre de atributo de capa de centro	`FIELD`	[campo de tabla: cualquiera]	Campo a usar para identificar unívocamente entidades de la capa de destino. Usado en la tabla de atributos saliente
Unidad de medida	`UNIT`	[enumeración] Predeterminado: 0	Unidades en las cuales informar de la distancia a la entidad mas cercana: • 0 — Metros • 1 — Pie • 2 — Millas • 3 — Kilometros • 4 — Unidades de capa
Distancia al centro	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa de vectores de puntos de salida con el centro más cercano. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Distancia al centro	`OUTPUT`	[vector: punto]	Capa vectorial lineal con los atributos de las entidades enteras, el identificador de su entidad mas cercana y la distancia calculada.

Código Python

Algoritmo ID: `qgis:distancetonearesthubpoints`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Unión por líneas (líneas de centro)

Crea diagramas de centro y radios conectando líneas desde puntos en la capa Spoke a puntos coincidentes en la capa Hub.

La determinación de qué centro va con cada punto se basa en una coincidencia entre el campo de ID de centro en los puntos de centro y el campo de ID de radio en los puntos de radio.

Si las capas de entrada no son capas de puntos, se tomará un punto en la superficie de las geometrías como ubicación de conexión.

Opcionalmente, se pueden crear líneas geodésicas, que representan la ruta más corta en la superficie de un elipsoide. Cuando se utiliza el modo geodésico, es posible dividir las líneas creadas en el antimeridiano (± 180 grados de longitud), lo que puede mejorar la representación de las líneas. Además, se puede especificar la distancia entre vértices. Una distancia menor da como resultado una línea más densa y precisa.

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

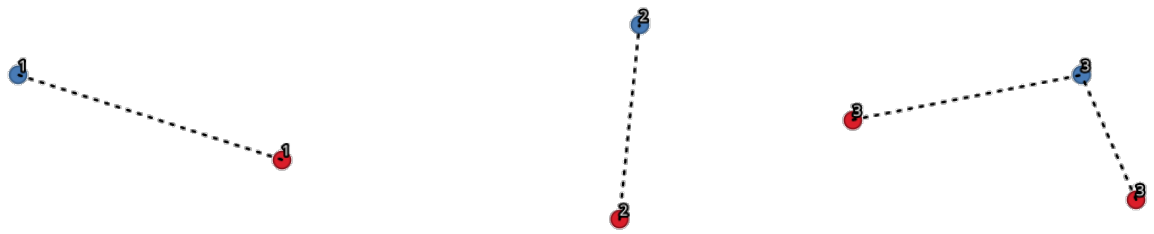


Figura 24.72: Unir puntos basados en un campo / atributo común

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa centro	`HUBS`	[vector: geometría]	Capa entrante
Campo ID centro	`HUB_FIELD`	[campo de tabla: cualquiera]	Campo de la capa de centros con la ID a unir
**Campos de la capa centro a copiar (dejar vacío para copiar todos los campos)* Opcional	`HUB_FIELDS`	[campo de tabla: cualquiera] [lista]	El campo(s) de la capa central que se copiará. Si no se elige ningún campo(s), se toman todos los campos.
Capa radios	`SPOKES`	[vector: geometría]	Capa de puntos adicional de radios
Campo ID radio	`SPOKE_FIELD`	[campo de tabla: cualquiera]	Campo de la capa de radios con ID a unir

continúe en la próxima página

Tabla 24.463 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Campos de la capa de radios a copiar (dejar vaciá para copiar todos los campos) Opcional	`SPOKE_FIELDS`	[campo de tabla: cualquiera] [lista]	Campo(s) de la capa de radios a copiar. Si no hay campos escogidos se toman todos los campos.
Crear líneas geodésicas	`GEODESIC`	[booleano] Predeterminado: Falso	Crear líneas geodésicas (la ruta mas corta en la superficie de un elipsoide)
Líneas de centro	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de vectores de la línea del hub de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Distancia entre vértices (solo líneas geodésicas)	`GEODESIC_DISTA`	[numérico: doble] Predeterminado: 1000.0 (kilometros)	Distancia entre vértices consecutivos (en kilómetros). Una distancia menor se traduce en una línea mas densa, mas precisa
Cortar líneas al antimeridiano (±180 grados de longitud)	`ANTIMERIDIAN_S`	[booleano] Predeterminado: Falso	Cortar líneas a longitud ±180 grados (para mejorar la representación de las líneas)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Líneas de centro	`OUTPUT`	[vectorial: lineal]	La capa de línea resultante que conecta los puntos coincidentes en las capas de entrada

Código Python

Algoritmo ID: `native:mergelines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Agrupación de K-medias

Calcula el número de clúster de k-medias basado en la distancia 2D para cada entidad de entrada.

La agrupación en clústeres de K-medias tiene como objetivo dividir las entidades en k grupos en los que cada entidad pertenece al grupo con la media más cercana. El punto medio está representado por el baricentro de las entidades agrupadas.

Si las geometrías de entrada son líneas o polígonos, la agrupación se basa en el centroide de la entidad.

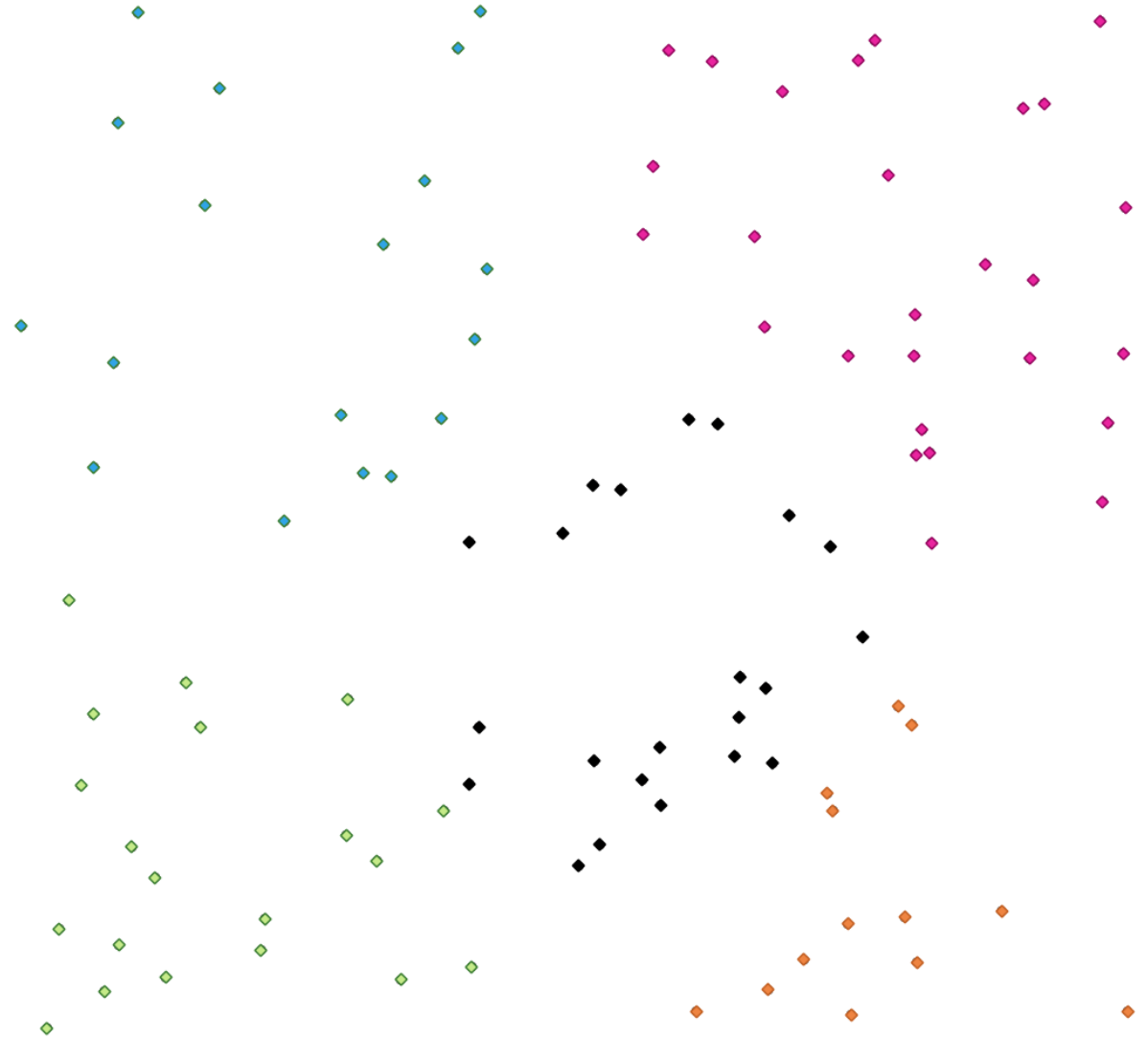


Figura 24.73: Un grupo de cinco puntos de clase

Ver también

Clustering o agrupamiento DBSCAN, *Agrupación ST-DBSCAN*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa a analizar
Número de grupos	`CLUSTERS`	[numérico: entero] Predeterminado: 5	Número de grupos a crear con los objetos.
Methodo	`METHOD`	[enumeración] Predeterminado: 0	Initial centers selection method. Possible values are: • 0 — Farthest points: the initial centers are selected as the farthest points from each other. • 1 — K-means++: The first center is selected uniformly at random from the set of input features. Subsequent centers are chosen based on a weighted probability distribution that favors features farther from existing centers. This ensures that the initial centers are well-dispersed, reducing the likelihood of poor clustering and improving the algorithm's performance. *References: Arthur, David & Vassilvitskii, Sergei. (2007). K-Means++: The Advantages of Careful Seeding. Proc. of the Annu. ACM-SIAM Symp. on Discrete Algorithms. 8.*
Clusters	`OUTPUT`	[vectorial: igual que la entrada] Predeterminado: `[Create temporary layer]`	Especifique la capa vectorial de salida para los clústeres generados. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Nombre de campo de grupo	`FIELD_NAME`	[cadena de texto] Predeterminado: "CLUSTER_ID"	Nombre del campo donde se almacenará el número de grupo asociado
Nombre de campo de tamaño de clúster	`SIZE_FIELD_NAM`	[cadena de texto] Predeterminado: "CLUSTER_SIZE"	Nombre del campo con el recuento de entidades en el mismo clúster

Salidas

Etiqueta	Nombre	Tipo	Descripción
Clusters	`OUTPUT`	[vectorial: igual que la entrada]	Capa vectorial que contiene las entidades originales con un campo que establece el grupo al que pertenecen

Código Python

Algoritmo ID: `native:meshrasterize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Lista de valores únicos

Lista de valores únicos de un campod e la tabla de atributos y recuento de sus números.

Menú predeterminado: *Vectorial ► Herramientas de análisis*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa a analizar
Campo(s) destino	`FIELDS`	[campo de tabla: cualquiera] [lista]	Campo(s) a analizar
Valores únicos Opcional	`OUTPUT`	[vector: tabla] Predeterminado: `[Create temporary layer]`	Especifique la capa de tabla resumen con valores únicos. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Informe HTML Opcional	`OUTPUT_HTML_FI`	[html] Predeterminado: `[Save to temporary file]`	Informe HTML de valores únicos en *Procesamiento ► Visor de resultados*. *Uno de*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Valores únicos	`OUTPUT`	[vector: tabla]	Resumen de tabla de capa con valores únicos
Informe HTML	`OUTPUT_HTML_FI`	[html]	Informe HTML de valores únicos. Puede ser abierto desde *Procesos* ► *Visor de Resultados*
Total de valores únicos	`TOTAL_VALUES`	[numérico: entero]	Número de valores únicos en el campo de entrada
Valores únicos concatenados	`UNIQUE_VALUES`	[cadena de texto]	Una cadena con la lista separada por comas de valores únicos que se encuentran en el campo de entrada

Código Python

Algorithm ID: `native:listuniquevalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Coordenada(s) Media

Calcula una capa de puntos con el centro de masa de geometrías en una capa de entrada.

Se puede especificar que un atributo contenga pesos que se aplicarán a cada entidad al calcular el centro de masa.

Si se selecciona un atributo en el parámetro, las entidades se agruparán según los valores de este campo. En lugar de un solo punto con el centro de masa de toda la capa, la capa de salida contendrá un centro de masa para las entidades en cada categoría.

Menú predeterminado: *Vectorial* ► *Herramientas de análisis*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Campo de peso Opcional	`WEIGHT`	[campo de tabla: numérico]	Campo a usar si quieres realizar una media ponderada
Campo de ID único	`UID`	[campo de tabla: cualquiera]	Campo único sobre el que se realizará el cálculo de la media

continúe en la próxima página

Tabla 24.471 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Coordenadas medias	`OUTPUT`	[vector: punto] Predeterminado: `[Create temporary layer]`	Especifique la capa (vector de puntos) para el resultado. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Coordenadas medias	`OUTPUT`	[vector: punto]	Capa de punto(s) resultante

Código Python

Algoritmo ID: `native:mergelines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Análisis de vecino mas próximo

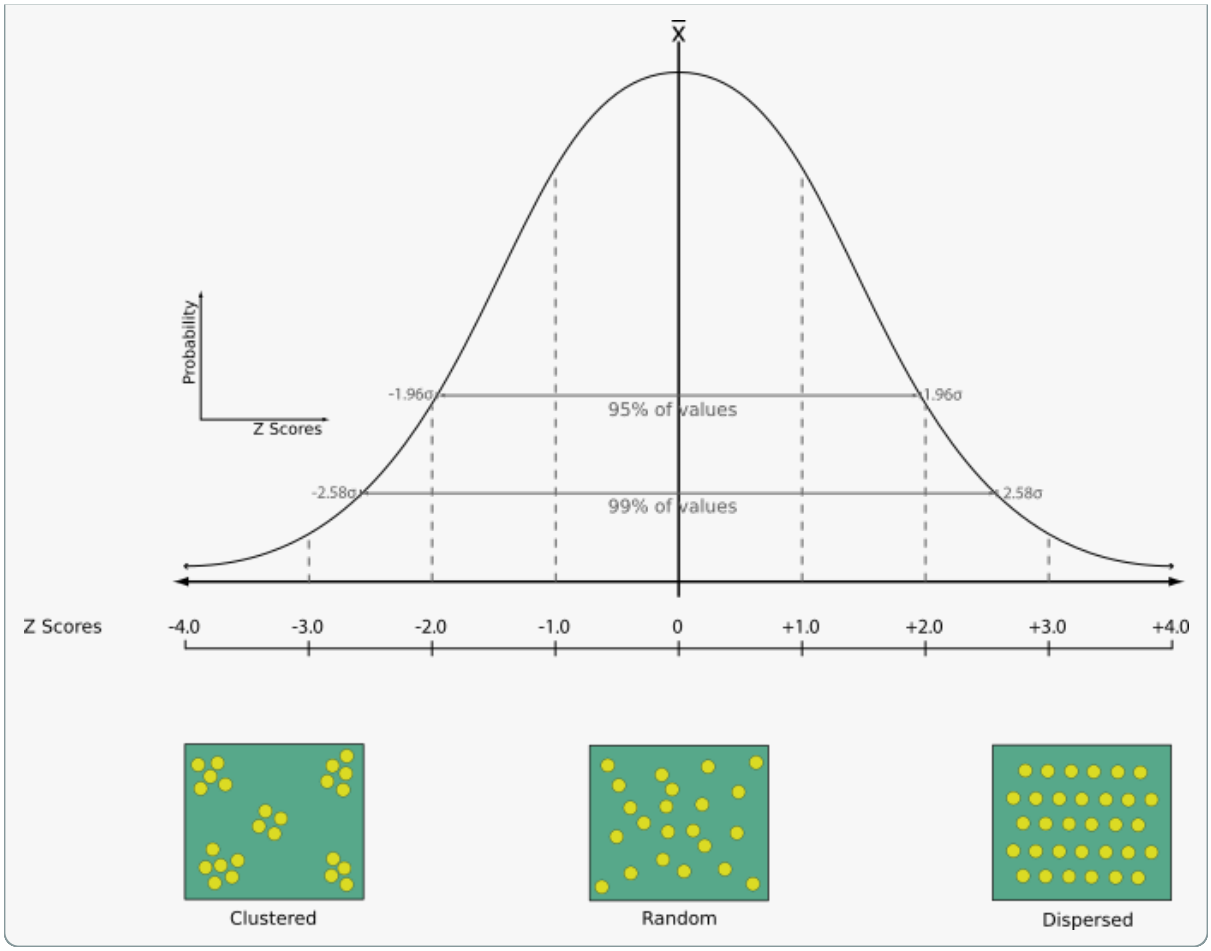
Realiza análisis de vecinos más cercanos para una capa de puntos. La salida le dice cómo se distribuyen sus datos (agrupados, aleatoriamente o distribuidos).

La salida se genera como un archivo HTML con los valores estadísticos calculados:

- Distancia media observada
- Distancia media esperada
- Índice de vecino mas próximo
- Número de puntos
- Z-Score: comparar el Z-Score con la distribución normal le indica cómo se distribuyen sus datos. Un Z-Score bajo significa que es poco probable que los datos sean el resultado de un proceso espacialmente aleatorio, mientras que un Z-Score alto significa que es probable que sus datos sean el resultado de un proceso espacialmente aleatorio.

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.



Menú predeterminado: *Vectorial* ► *Herramientas de análisis*

Ver también

Unir atributos por proximidad

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	Capa vectorial de puntos para calcular las estadísticas
Vecino mas próximo Opcional	`OUTPUT_HTML_FI`	[html] Predeterminado: `[Save to temporary file]`	Especificación del archivo HTML para las estadísticas calculadas. *Uno de*: • Saltar salida • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Vecino mas próximo	`OUTPUT_HTML_FI`	[html]	Archivo HTML con las estadísticas calculadas
Distancia media observada	`OBSERVED_MD`	[numérico: doble]	Distancia media observada
Distancia media esperada	`EXPECTED_MD`	[numérico: doble]	Distancia media esperada
Índice de vecino más cercano	`NN_INDEX`	[numérico: entero]	Índice de vecino mas próximo
Número de puntos	`POINT_COUNT`	[numérico: entero]	Número de puntos
Z-Score	`Z_SCORE`	[numérico: doble]	Z-Score

Código Python

Algoritmo ID: `native:deleteholes`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Análisis de superposición

Calcula el área y el porcentaje de cobertura mediante el cual las entidades de una capa de entrada se superponen con las entidades de una selección de capas superpuestas.

Se agregan nuevos atributos a la capa de salida que informan el área total de superposición y el porcentaje de la entidad de entrada superpuesta por cada una de las capas de superposición seleccionadas.

> **Nota**
>
> Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	La capa entrante.
Capas superpuestas	`LAYERS`	[vector: polygon] [list]	Las capas superpuestas.

continúe en la próxima página

Tabla 24.475 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Superposición	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tamaño de la cuadrícula Opcional	`GRID_SIZE`	[numérico: doble] Predeterminado: No establecido	If provided, the input geometries are snapped to a grid of the given size, and the result vertices are computed on that same grid.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Superposición	`OUTPUT`	[la misma que la entrada]	La capa de salida con campos adicionales que informan la superposición (en unidades de mapa y porcentaje) de la entidad de entrada superpuesta por cada una de las capas seleccionadas.

Código Python

ID Algoritmo: `native:splitvectorlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Línea más corta entre objetos espaciales

Crea una capa de línea como la línea más corta entre la capa de origen y la de destino. Por defecto, sólo se tiene en cuenta el primer objeto espacial más cercano de la capa de destino. Se puede especificar el número de objetos espaciales vecinos más cercanos. Si se especifica una distancia máxima, sólo se tendrán en cuenta los elementos que estén más cerca que esa distancia.

Los objetos espaciales de salida contendrán todos los atributos de la capa de origen, todos los atributos del objeto espacial n-más cercano y el campo adicional de la distancia.

Importante

Este algoritmo utiliza cálculos puramente cartesianos para la distancia, y no tiene en cuenta las propiedades geodésicas o elipsoidales a la hora de determinar la proximidad de las características. El sistema de coordenadas de medición y salida se basa en el sistema de coordenadas de la capa de origen.

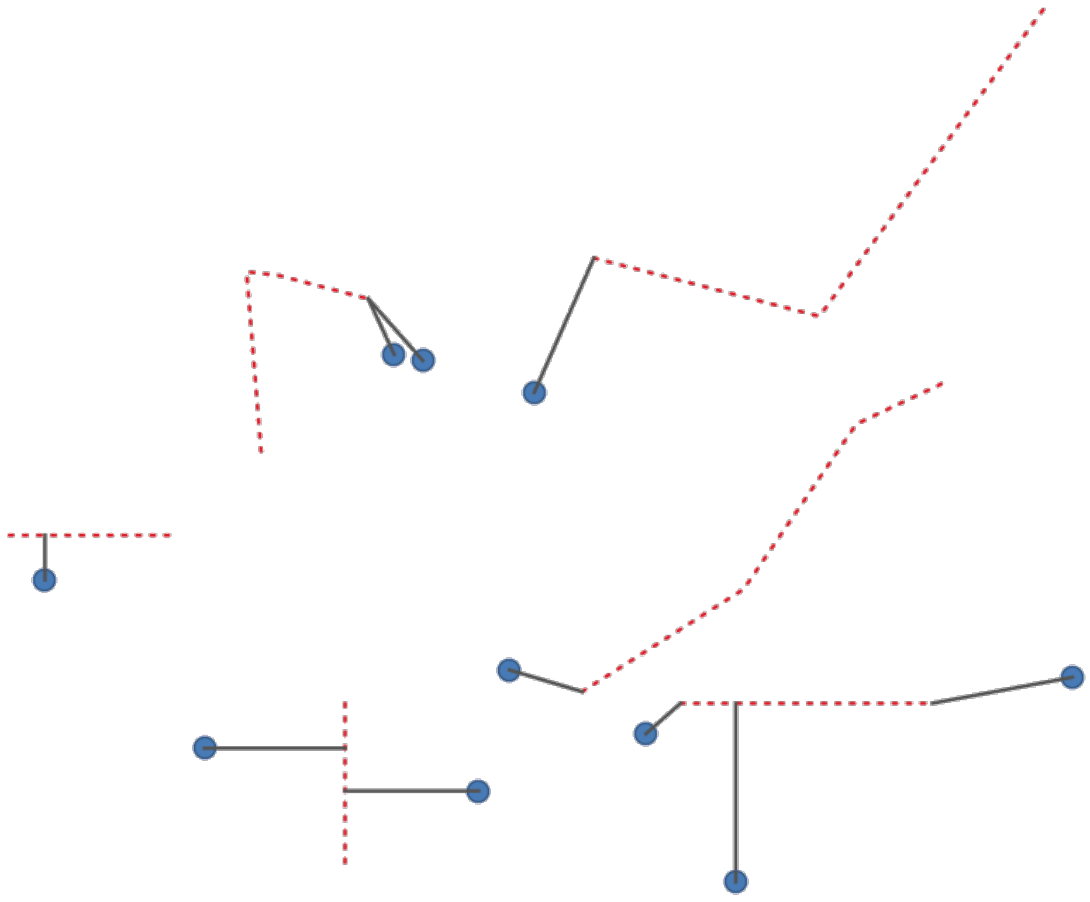


Figura 24.74: Línea más corta entre objetos espaciales punto y líneas

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de origen	`SOURCE`	[vector: geometría]	Capa de origen para la que buscar vecinos más cercanos
Capa de destino	`DESTINATION`	[vector: geometría]	Capa objetivo en la que buscar vecinos más cercanos
Methodo	`METHOD`	[enumeración] Predeterminado: 0	Método de cálculo de la distancia más corta Los valores posibles son: • 0 — Distancia al punto más cercano del elemento • 1 — Distancia al centroide del objeto espacial
Número máximo de vecinos	`NEIGHBORS`	[numérico: entero] Predeterminado: 1	Número máximo de vecinos a buscar
Distancia máxima Opcional	`DISTANCE`	[numérico: doble]	Sólo se tendrán en cuenta los objetos espaciales de destino que estén más cerca que esta distancia.
Líneas más cortas	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[vectorial: lineal]	Capa de vectores lineales que une los elementos de origen con su(s) vecino(s) más cercano(s) en la capa de destino. Contiene todos los atributos de los elementos de origen y destino, así como la distancia calculada.

Código Python

ID del algoritmo: `native:shortestline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Agrupación ST-DBSCAN

Agrupa entidades puntuales basadas en una implementación 2D del algoritmo de agrupamiento de aplicaciones con ruido (ST-DBSCAN) basado en la densidad espaciotemporal.

Ver también

Clustering o agrupamiento DBSCAN, *Agrupación de K-medias*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	Capa a analizar
Campo de fecha/hora	`DATETIME_FIELD`	[campo de tabla: fecha]	Campo que contiene la información temporal
Tamaño mínimo de clúster	`MIN_SIZE`	[numérico: entero] Predeterminado: 5	Número mínimo de objetos para generar un grupo
Distancia máxima entre puntos agrupados	`EPS`	[numérico: doble] Predeterminado: 1.0	Distancia más allá de la cual dos entidades no pueden pertenecer al mismo grupo (eps)
Duración máxima de tiempo entre puntos agrupados	`EPS2`	[numérico: doble] Predeterminado: 0.0 (días)	Tiempo de duración más allá del cual dos entidades no pueden pertenecer al mismo clúster (eps2). Las unidades de tiempo disponibles son milisegundos, segundos, minutos, horas, días y semanas.
Clusters	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial para el resultado de la agrupación. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tratar los puntos fronterizos como ruido (DBSCAN*)	`DBSCAN*`	[booleano] Predeterminado: Falso	Si se marca, los puntos en el borde de un grupo se tratan como puntos no agrupados, y solo los puntos en el interior de un grupo se etiquetan como agrupados.
Nombre de campo de grupo	`FIELD_NAME`	[cadena de texto] Predeterminado: “CLUSTER_ID”	Nombre del campo donde se almacenará el número de grupo asociado
Nombre de campo de tamaño de clúster	`SIZE_FIELD_NAM`	[cadena de texto] Predeterminado: “CLUSTER_SIZE”	Nombre del campo con el recuento de entidades en el mismo clúster

Salidas

Etiqueta	Nombre	Tipo	Descripción
Clusters	`OUTPUT`	[vector: punto]	Capa vectorial que contiene las entidades originales con un campo que establece el grupo al que pertenecen
Número de grupos	`NUM_CLUSTERS`	[numérico: entero]	El número de grupos descubiertos

Código Python

Algoritmo ID: `native:linesubstring`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Estadísticas por categorías

Calcula las estadísticas de un campo en función de una clase principal. La clase principal es una combinación de valores de otros campos.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa vectorial entrante	`INPUT`	[vectorial: cualquiera]	Capa vectorial de entrada con clases y valores únicos
Campo para calcular estadísticas (si está vacío, solo se calcula el recuento) Opcional	`VALUES_FIELD_N`	[campo de tabla: cualquiera]	Si está vacío solo se calcula el recuento
Campo(s) con categorías	`CATEGORIES_FIE`	[campo de tabla: cualquiera] [lista]	Los campos que (combinados) definen las categorías
Estadísticas por categoría	`OUTPUT`	[vector: tabla] Predeterminado: `[Crear capa temporal]`	Especifique la tabla de salida para las estadísticas generadas. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Estadísticas por categoría	`OUTPUT`	[vector: tabla]	Tabla contenedora de las estadísticas

Dependiendo del tipo del campo siendo analizado, las siguientes estadísticas son devueltas para cada valor de grupo:

Estadísticas	Cadena	Numeric	Fecha
Recuento (`COUNT`)	✓	✓	✓
Valores únicos (`UNIQUE`)	✓		✓
Valores vacíos (nulos) (`EMPTY`)	✓		✓
Valores no vacíos (`FILLED`)	✓		✓
Valor mínimo (`MIN`)	✓	✓	✓
Valor máximo (`MAX`)	✓	✓	✓
Rango (`RANGE`)		✓	
Suma (`SUM`)		✓	
Valor medio(`MEAN`)		✓	
Valor mediana (`MEDIAN`)		✓	
Desviación Estándar (`STD_DEV`)		✓	
Coeficiente de variación (`CV`)		✓	
Minoría (valor suceso mas raro - `MINORITY`)		✓	
Mayoría (valor suceso mas frecuente - `MAJORITY`)		✓	
Primer cuartil (`FIRSTQUARTILE`)		✓	
Tercer cuartil (`THIRDQUARTILE`)		✓	
Rango Inter Cuartil (`IQR`)		✓	
Longitud Mínima (`MIN_LENGTH`)	✓		
Longitud Media (`MEAN_LENGTH`)	✓		
Longitud Máxima (`MAX_LENGTH`)	✓		

Código Python

Algoritmo ID: `qgis:statisticsbycategories`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Suma de la longitud de las líneas

Toma una capa poligonal y una capa lineal y mide la longitud total de líneas y el número total de ellas que cruzan cada polígono.

La capa resultante tiene los mismos objetos que la capa poligonal entrante, pero con dos atributos adicionales conteniendo la longitud y el recuento de las líneas que cruzan cada polígono.

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

 Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Menú predeterminado: *Vectorial ► Herramientas de análisis*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Líneas	`LINES`	[vectorial: lineal]	Capa vectorial lineal entrante
Polígonos	`POLYGONS`	[vectorial: poligonal]	Capa vectorial poligonal
Nombre de campo de la longitud de líneas	`LEN_FIELD`	[cadena de texto] Predeterminado: “LENGTH”	Nombre del campo para la longitud de las líneas
Nombre de campo del recuento de líneas	`COUNT_FIELD`	[cadena de texto] Predeterminado: “COUNT”	Nombre del campo para el recuento de líneas
Longitud de Línea	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de polígonos de salida con estadísticas generadas. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Longitud de Línea	`OUTPUT`	[vectorial: poligonal]	Capa poligonal saliente con campos de longitud y recuento de líneas

Código Python

Algoritmo ID: `native:splitlinesbylength`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.23 Cobertura vectorial

Disolver la cobertura

Opera sobre una cobertura (representada como un conjunto de características poligonales con geometría de bordes que coinciden exactamente) para disolver (unir) las geometrías. Ofrece un enfoque muy optimizado para unir estas características en comparación con las herramientas de disolución estándar.

Ver también

Disolver

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa vectorial de polígonos entrante
Disuelto	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Disuelto	`OUTPUT`	[vectorial: poligonal]	La capa vectorial de polígonos de salida con geometrías disueltas.

Código Python

ID del algoritmo: `native:coverageunion`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Simplificar la cobertura

Opera sobre una cobertura (representada como un conjunto de objetos espaciales polígono con una geometría de aristas que coincide exactamente) para aplicar una simplificación Visvalingam-Whyatt a las aristas, reduciendo la complejidad en proporción con la tolerancia proporcionada, al tiempo que se conserva una cobertura válida (es decir, ninguna arista se cruzará o tocará después de la simplificación). Las geometrías nunca se eliminan, pero pueden simplificarse hasta reducirse a un triángulo. Además, algunas geometrías (como los polígonos que tienen muy pocos puntos no repetidos) se devolverán sin cambios.

Si el conjunto de datos de entrada no es una cobertura válida debido a solapamientos, seguirá simplificándose, pero la topología no válida, como los bordes que se cruzan, seguirá siendo inválida.

Atención

Running this algorithm requires QGIS installed with GEOS >= 3.12 (see *Help* ► *About* menu).

Ver también

Simplificar, *Suavizar*

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa vectorial de polígonos entrante
Tolerancia	`TOLERANCE`	[numérico: doble] Predeterminado: 0.0	La distancia máxima (en unidad de elección) entre dos vértices consecutivos a fusionar.
Preservar el límite	`PRESERVE_BOUND`	[booleano] Predeterminado: False	Si esta opción está activada, los bordes exteriores de la cobertura se conservarán sin simplificación.
Simplificado	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Simplificado	`OUTPUT`	[vectorial: poligonal]	La capa vectorial de polígonos de salida, con un número menor de vértices.

Código Python

ID del algoritmo: `native:coveragesimplify`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Validar la cobertura

Analiza una cobertura (representada como un conjunto de objetos espaciales polígono con geometría de aristas exactamente coincidentes) para encontrar lugares en los que no se cumple el supuesto de aristas exactamente coincidentes. La invalidez incluye polígonos que se solapan o que tienen huecos menores que la anchura de hueco especificada.

Atención

Running this algorithm requires QGIS installed with GEOS >= 3.12 (see *Help* ► *About* menu).

Ver también

Comprobar validez, *Small gaps*, *Overlaps*

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa vectorial de polígonos entrante
Ancho de separación	`GAP_WIDTH`	[numérico: doble] Predeterminado: 0.0	Anchura máxima del espacio a detectar

continúe en la próxima página

Tabla 24.492 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Bordes no válidos	`INVALID_EDGES`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Bordes no válidos	`OUTPUT`	[vectorial: lineal]	La capa vectorial de líneas de salida que muestra las aristas de error de cada polígono no válido.
Resultado de validez	`IS_VALID`	[booleano]	Devuelve si la cobertura es válida o no.

Código Python

ID del algoritmo: `native:coveragevalidate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.24 Creación vectorial

Colección de lineas desplazadas (paralelas)

Crea copias de entidades de línea en una capa mediante la creación de varias versiones desplazadas de cada entidad. Cada nueva versión se desplaza gradualmente en una distancia especificada.

Una distancia positiva desplazará las líneas hacia la izquierda, mientras que las negativas las desplazarán hacia la derecha.

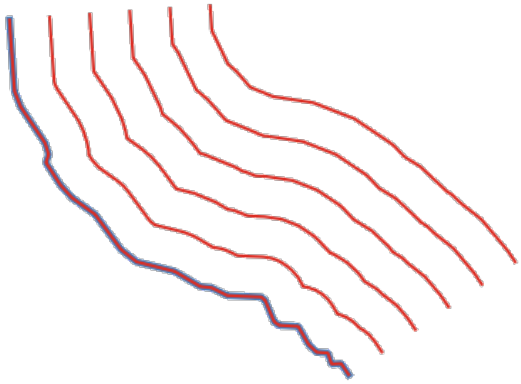


Figura 24.75: En azul la capa fuente, en rojo la compensada

Permite *modificación in situ* de objetos espaciales línea

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

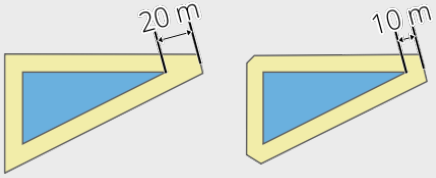
Compensado de líneas, *Colección de objetos trasladados*

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante a usar para los desplazamientos.
Número de entidades a crear	`COUNT`	[numérico: entero] Predeterminado: 10	Número de copias desplazadas a generar para cada objeto
Distancia de paso de desplazamiento	`OFFSET`	[numérico: doble] Predeterminado: 1.0	Distancia entre dos copias consecutivas desplazadas
Líneas desplazadas	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de línea de salida con características de desplazamiento. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Segmentos	`SEGMENTS`	[numérico: doble] Preestablecido: 8	Número de segmentos lineales a usar para aproximar un cuarto de círculo al crear compensaciones redondeadas
Unir estilo	`JOIN_STYLE`	[enumeración] Predeterminado: 0	Especifique si se deben utilizar uniones redondas, a inglete o biseladas al compensar esquinas en una línea. Uno de: • 0 — Redondo • 1 — Inglete • 2 — Bisel  Figura 24.76: Estilos de unión redonda, a inglete y en bisel
Límite de inglete	`MITER_LIMIT`	[numérico: doble] Preestablecido: 2.0	Establezca la distancia máxima desde la geometría de desplazamiento que se utilizará al crear una unión a inglete como factor de la distancia de desplazamiento (sólo aplicable a los estilos de unión a inglete). Mínimo: 1.0  Figura 24.77: Un área de influencia (búfer) de 10 m con un límite de 2 y un área de influencia de 10 m con un límite de 1

Salidas

Etiqueta	Nombre	Tipo	Descripción
Líneas desplazadas	`OUTPUT`	[vectorial: lineal]	Capa de línea de salida con las características originales y desplazadas. Los atributos se copiarán a las salidas correspondientes. Se añadirán los siguientes atributos relacionados con los parámetros de entrada: `instance` que indexa las líneas desplazadas `COUNT` y `offset` que designa la distancia de desplazamiento desde la línea original.

Código Python

Algoritmo ID: `native:arrayoffsetlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Colección de objetos trasladados

Crea copias de entidades en una capa creando múltiples versiones trasladadas de cada una. Cada copia se desplaza incrementalmente por una cantidad preestablecida en los ejes X, Y y/o Z.

Valores M presentes en la geometría pueden también ser trasladadas.

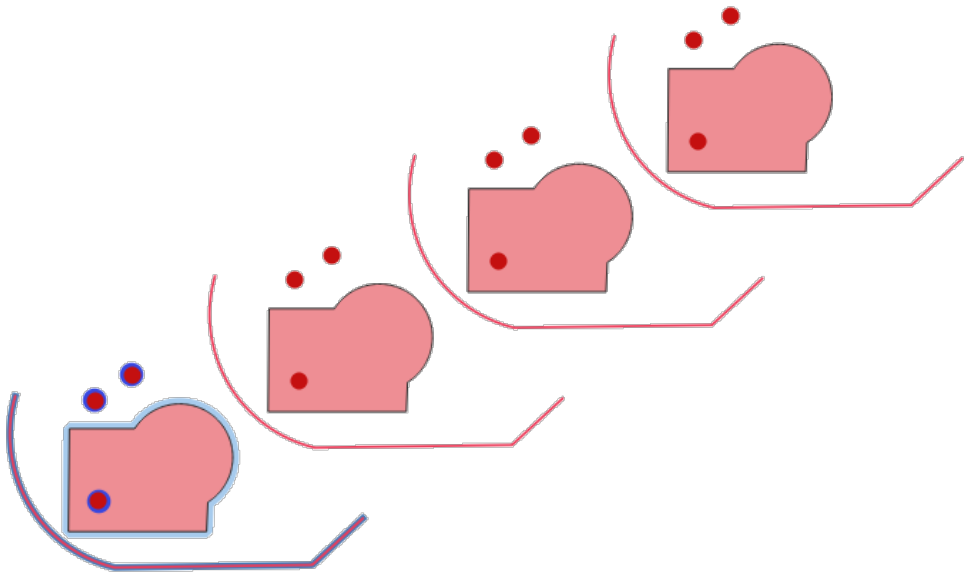


Figura 24.78: Capas entrantes en tono azul, capas salientes con objetos trasladados en tonos rojos

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Traslado, *Colección de lineas desplazadas (paralelas)*

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa vectorial entrante a trasladar
Número de entidades a crear	`COUNT`	[numérico: entero]  Predeterminado: 10	Número de copias a generar para cada objeto
Distancia de paso (eje-x)	`DELTA_X`	[numérico: doble]  Predeterminado: 0.0	Desplazamiento a aplicar en el eje X
Distancia de paso (eje-y)	`DELTA_Y`	[numérico: doble]  Predeterminado: 0.0	Desplazamiento a aplicar en el eje Y
Distancia de paso (eje-z)	`DELTA_Z`	[numérico: doble]  Predeterminado: 0.0	Desplazamiento a aplicar en el eje Z
Distancia de paso (valores M)	`DELTA_M`	[numérico: doble]  Predeterminado: 0.0	Desplazamiento a aplicar en M
Trasladado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Capa vectorial de salida con copias traducidas (movidas) de las características. Las características originales también se copian. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Trasladado	`OUTPUT`	[la misma que la entrada]	Capa de vector de salida con copias trasladadas (movidas) de las entidades. También se copian las entidades originales.

Código Python

Algoritmo ID: `native:arraytranslatedfeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Crear cuadrícula

Crea una capa vectorial con una cuadrícula cubriendo una extensión dada. La celdas de rejilla pueden tener diferentes formas:

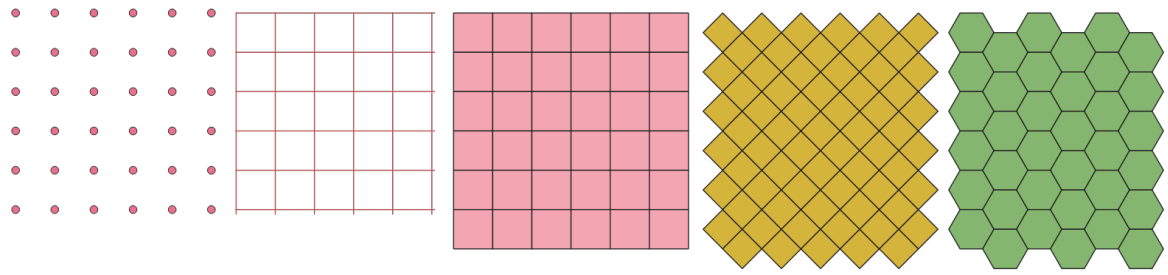


Figura 24.79: Diferentes formas de celda de grid aplicadas a la misma extensión, sin solapamientos

El tamaño y/o la colocación de cada elemento en la cuadrícula se definen mediante un espaciado horizontal y vertical. Debe definirse el SRC de la capa de salida. La extensión de la cuadrícula y los valores de espaciado deben expresarse en las coordenadas y unidades de este SRC.

menú predeterminado: *Vectorial ► Herramientas de Investigación*

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Tipo de cuadrícula	`TYPE`	[enumeración] Predeterminado: 0	Forma de la cuadrícula. Una de: • 0 — Punto • 1 — Lineal • 2 — Rectangular (poligonal) • 3 — Diamantada (poligonal) • 4 — Hexagonal (poligonal)

continúe en la próxima página

Tabla 24.499 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extensión de cuadrícula	`EXTENT`	[extensión]	Extensión de la cuadrícula Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Espaciado horizontal	`HSPACING`	[numérico: doble] Predeterminado: 1.0	Tamaño de celda de cuadrícula en el eje-X
Espaciado vertical	`VSPACING`	[numérico: doble] Predeterminado: 1.0	Tamaño de una celda de cuadrícula en el eje-Y
Superposición horizontal	`HOVERLAY`	[numérico: doble] Predeterminado: 0.0	Distancia de superposición entre dos celdas de cuadrícula consecutivas en el eje X
Superposición vertical	`VOVERLAY`	[numérico: doble] Predeterminado: 0.0	Distancia de superposición entre dos celdas de cuadrícula consecutivas en el eje Y
SRC de Cuadrícula	`CRS`	[src] Predeterminado: *SRC de Proyecto*	Sistema de referencia de coordenadas a aplicar a la cuadrícula
Cuadrícula	`OUTPUT`	[vector: geometría] Predeterminado: `[Crear capa temporal]`	Capa de cuadrícula vectorial resultante. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cuadrícula	`OUTPUT`	[vector: geometría]	Capa de malla vectorial resultante. El tipo de geometría de salida (punto, línea o polígono) depende de *Tipo de cuadrícula*. Los objetos espaciales se crean de arriba a abajo, de izquierda a derecha. La tabla de atributos se rellena con: • un `id` • coordenadas de los lados `izquierdo`, `derecho`, `superior` e `inferior`. • y su ubicación en la cuadrícula: `row_index` y `column_index` (disponibles para los tipos de rejilla punto, rectángulo y hexágono)

Código Python

Algoritmo ID: `native:creategrid`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Crea una capa desde extensión

Crea una nueva capa vectorial que contiene una única entidad con geometría que coincide con la extensión de la capa de entrada.

Se puede usar en modelos para convertir una extensión literal (formato `` xmin “”, `` xmax “”, `` ymin “”, `` ymax “”) en una capa que se puede usar para otros algoritmos que requieren un entrada basada en capas.

Ver también

Crear capas desde puntos

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Extesión (xmin, xmax, ymin, ymax)	`INPUT`	[extensión]	Extensión de entrada Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Extensión	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extensión	`OUTPUT`	[vectorial: poligonal]	La capa vectorial saliente (extensión)

Código Python

Algoritmo ID: `native:extenttolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Crear capas desde puntos

Crea una nueva capa vectorial que contiene una única entidad con geometría que coincide con un parámetro de punto. Se puede usar en modelos para convertir un punto en una capa de puntos para algoritmos que requieren una entrada basada en capas.

Ver también

Crea una capa desde extensión

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Punto	`INPUT`	[coordenadas]	Entrade de puntos, incluyendo información del CRS (ejemplo: `397254,6214446 [EPSG:32632]`). Si el CRS no es proporcionado, se usará el CRS del Proyecto. El punto puede ser especificado clickando en el canvas del mapa.
Punto	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifica la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Punto	`OUTPUT`	[vector: punto]	La capa vectorial de puntos saliente conteniendo los puntos de entrada.

Código Python

Algoritmo ID: `native:pointtolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Crear capa de puntos desde tabla

Crea capas de puntos a partir de una tabal con columnas que contienen campos de coordenadas.

Además de las coordenadas X e Y, también puede especificar campos Z y M.

> **Advertencia**
>
> Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial entrante o una tabla.
Campo X	`XFIELD`	[campo de tabla: cualquiera]	Campo contenedor de la coordenada X
Campo Y	`YFIELD`	[campo de tabla: cualquiera]	Campo contenedor de la coordenada Y
Campo Z Opcional	`ZFIELD`	[campo de tabla: cualquiera]	Campo contenedor de la coordenada Z
Campo M Opcional	`MFIELD`	[campo de tabla: cualquiera]	Campo contenedor del valor M
SRC destino	`TARGET_CRS`	[src] Predeterminado: `EPSG:4326`	Sistema de referencia de coordenadas que se utilizará para la capa. Se supone que las coordenadas proporcionadas son compatibles.
Puntos desde tabla	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa de puntos resultante. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos desde tabla	`OUTPUT`	[vector: punto]	La capa de puntos resultante

Código Python

Algoritmo ID: `native:createpointslayerfromtable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Genera puntos (centroides de pixel) a lo largo de la línea

Genera una capa vectorial de punto desde un ráster entrante y una capa lineal.

Los puntos corresponden al centroide de pixel que interseca la capa lineal.

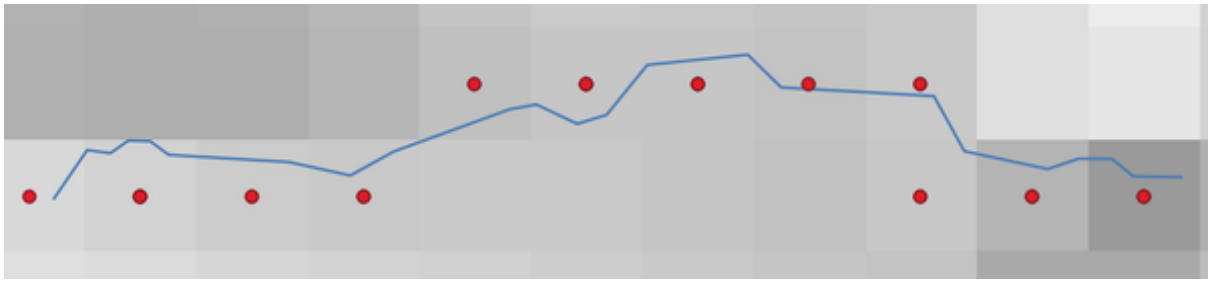


Figura 24.80: Puntos de los centroides de pixel

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa ráster	`INPUT_RASTER`	[ráster]	Capa ráster de entrada
Capa vectorial	`INPUT_VECTOR`	[vectorial: lineal]	Capa vectorial entrante de lineas
Puntos a lo largo de línea	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Capa de puntos resultante con centroides de píxeles. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos a lo largo de línea	`OUTPUT`	[vector: punto]	Capa de puntos resultante con centroides de pixel

Código Python

Algoritmo ID: `qgis:generatepointspixelcentroidsalongline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Genera puntos (centroides de pixel) dentro del polígono

Genera una capa vectorial de puntos a partir de un ráster entrante y una capa poligonal.

Los puntos corresponden a los centroides de pixel que intersecan la capa poligonal.

> **Advertencia**
>
> Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

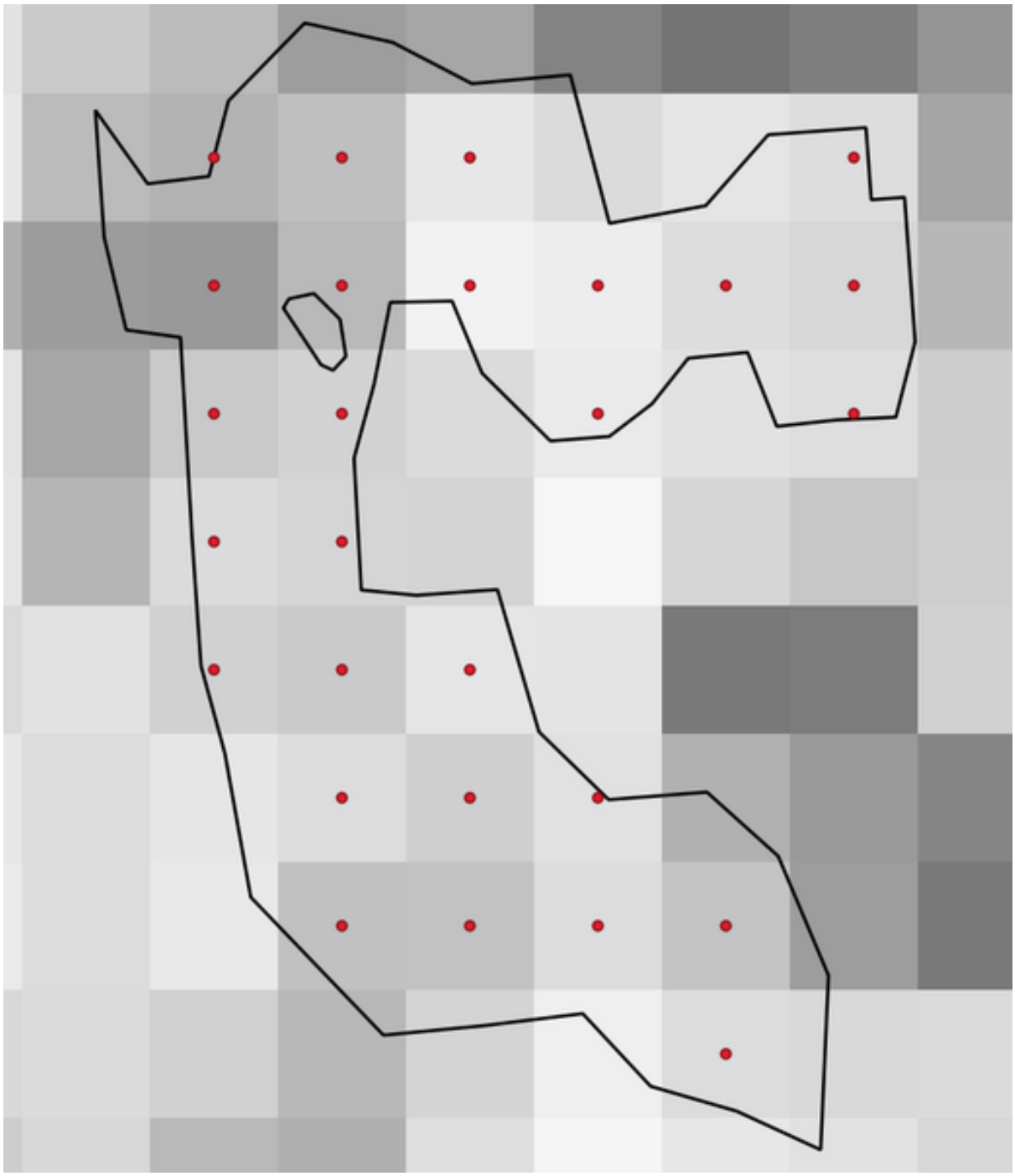


Figura 24.81: Puntos de los centroides de pixel

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa ráster	`INPUT_RASTER`	[ráster]	Capa ráster de entrada
Capa vectorial	`INPUT_VECTOR`	[vectorial: poligonal]	Capa vectorial de polígonos entrante
Puntos dentro de polígonos	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Capa de puntos resultante de los centroides de píxeles. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos dentro de polígonos	`OUTPUT`	[vector: punto]	Capa de puntos resultante de centroides de píxeles

Código Python

Algoritmo ID: `native:generatepointspixelcentroidsinsidepolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Importar fotos geoetiquetadas

Crea una capa de puntos correspondiente a las ubicaciones geoetiquetadas a partir de imágenes JPEG o HEIC/HEIF de una carpeta de origen.

La capa de puntos contendrá una única entidad PointZ por archivo de entrada desde la que se pueden leer las etiquetas geográficas. Cualquier información de altitud de las geoetiquetas se utilizará para establecer el valor Z del punto.

Además de la longitud y la latitud, la información de altitud, dirección y marca de tiempo, si está presente en la foto, se agregará al punto como atributos.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Carpeta de entrada	`FOLDER`	[carpeta]	Ruta a la carpeta fuente contenedora de las fotos geoetiquetadas
Escanear recursivamente	`RECURSIVE`	[booleano] Predeterminado: False	Si está marcado, la carpeta y sus subcarpetas serán escaneadas
Fotos Opcional	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa de vectores de puntos para las fotos geolocalizadas. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Tabla inválida de fotos Opcional	`INVALID`	[vector: tabla] Preestablecido: `[Omitir salida]`	Especifique la tabla de fotos ilegibles o sin etiquetas geográficas. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Fotos	`OUTPUT`	[vector: punto]	Punto de capa vectorial con fotos geoetiquetadas. La forma de la capa se llena automáticamente con rutas y configuraciones de vista previa de fotos.
Tabla inválida de fotos Opcional	`INVALID`	[vector: tabla]	También se puede crear una tabla de fotos ilegibles o no geoetiquetadas.

Código Python

Algoritmo ID: `native:importphotos`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Puntos a ruta

Convierte una capa de puntos en una capa de líneas uniendo puntos en un orden definido por una expresión o un campo en la capa de puntos de entrada.

Los puntos se pueden agrupar por un campo o una expresión para distinguir las características de línea.

Además de la capa de vector de línea, se genera un archivo de texto que describe la línea resultante como un punto de inicio y una secuencia de rumbos/direcciones (en relación con el acimut) y distancias.

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada de puntos	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Crea caminos cerrados	`CERRAR_ITINERA`	[booleano] Predeterminado: False	Si está marcado, se conectarán el primer y el último punto de la línea y se cerrará el trayecto generado
Expresión de orden Opcional	`ORDER_EXPRESSI`	[expresión]	Campo o expresión que proporciona el orden para conectar los puntos de la ruta. Si no se establece, se utiliza el ID de función (`$id`).
Ordenar texto con números de forma natural	`NATURAL_SORT`	[booleano] Predeterminado: False	Si está marcado, ordena naturalmente las características según la expresión proporcionada (es decir, “a9” < “a10”).
Expresión de grupo de ruta Opcional	`GROUP_EXPRESSI`	[expresión]	Las entidades puntuales del mismo valor en el campo o expresión se agruparán en la misma línea. Si no se establece, se dibuja una sola ruta con todos los puntos de entrada.
Rutas	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de vectores de línea de la ruta. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Directorio para salida de texto Opcional	`OUTPUT_TEXT_DI`	[carpeta] Preestablecido: `[Omitir salida]`	Especifique el directorio que contendrá los archivos de descripción de puntos y rutas. *Uno de*: • Saltar salida • Guardar en Directorio Temporal • Guardar en Directorio

Salidas

Etiqueta	Nombre	Tipo	Descripción
Rutas	`OUTPUT`	[vectorial: lineal]	Capa vectorial lineal de la ruta
Directorio para salida de texto	`OUTPUT_TEXT_DI`	[carpeta]	Directorio conteniendo los archivos de descripción de puntos y rutas

Código Python

Algoritmo ID: `native:pointstopath`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Puntos aleatorios a lo largo de la línea

Crea una nueva capa de puntos, con puntos situados sobre las líneas de otra capa.

Para cada línea de la capa de entrada, se añade un número determinado de puntos a la capa resultante. El procedimiento para añadir un punto es:

1. seleccionar al azar una entidad de línea de la capa de entrada
2. si la entidad es multi-parte, seleccionar al azar una de sus partes
3. seleccionar al azar un segmento de esa línea
4. seleccionar al azar una posición sobre ese segmento

El procedimiento implica que las partes curvas de las líneas (con segmentos relativamente cortos) recibirán más puntos que las partes rectas (con segmentos relativamente largos), como queda representado en la ilustración de abajo, en la que la salida del algoritmo *Puntos aleatorios a lo largo de líneas* puede compararse con la de *Puntos aleatorios sobre líneas* (que produce, en promedio, puntos con una distribución más regular a lo largo de las líneas).

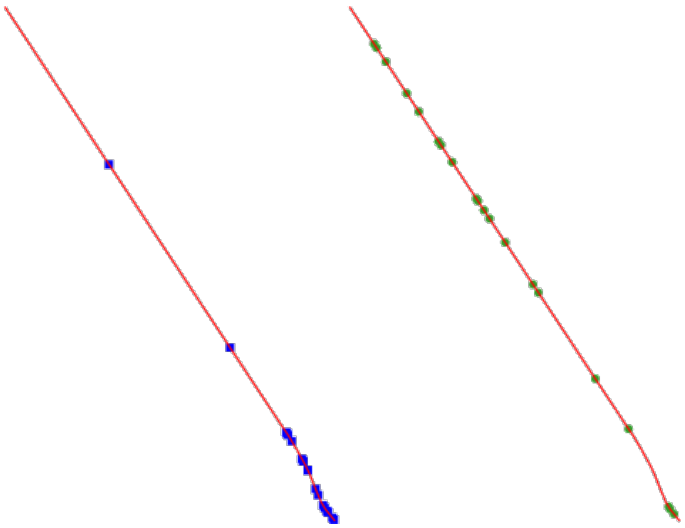


Figura 24.82: Ejemplo de salida de algoritmo. Izquierda: *Puntos aleatorios a lo largo de línea*, derecha: *Puntos aleatorios sobre líneas*

Una distancia mínima puede ser especificada, para evitar que los puntos estén muy cerca unos de otros.

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Puntos aleatorios sobre líneas

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada de puntos	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Número de puntos	`POINTS_NUMBER`	[numérico: entero] Predeterminado: 1	Número de puntos a crear
Distancia mínima entre puntos	`MIN_DISTANCE`	[numérico: doble] Predeterminado: 0.0	La distancia mínima entre puntos
Puntos aleatorios	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Los puntos aleatorios de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos aleatorios	`OUTPUT`	[vector: punto]	La capa de puntos aleatorios saliente.

Código Python

Algoritmo ID: `qgis:qgisrandompointsalongline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Puntos aleatorios en extensión

Crea una nueva capa de puntos con un número dado de puntos aleatorios, todos ellos dentro de una extensión dada.

Se puede especificar un factor de distancia, para evitar que los puntos queden demasiado cerca unos de otros. Si la distancia mínima entre puntos hace imposible crear nuevos puntos, se puede disminuir la distancia o aumentar el número de intentos.

menú predeterminado: *Vectorial ► Herramientas de Investigación*

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Extensión de entrada	`EXTENT`	[extensión]	Extensión de mapa para los puntos aleatorios Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Número de puntos	`POINTS_NUMBER`	[numérico: entero] Predeterminado: 1	Número de puntos a crear
Distancia mínima entre puntos	`MIN_DISTANCE`	[numérico: doble] Predeterminado: 0.0	La distancia mínima entre puntos
SRC destino	`TARGET_CRS`	[src] Predeterminado: *SRC de Proyecto*	SRc de la capa de puntos aleatorios
Puntos aleatorios	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Los puntos aleatorios de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Número máximo de intentos dada la distancia mínima	`NÚMERO_MÁXIMO_`	[numérico: entero] Por defecto: 200	Número máximo de intentos para ubicar puntos

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos aleatorios	`OUTPUT`	[vector: punto]	La capa de puntos aleatorios saliente.

Código Python

ID Algoritmo: `native:randompointsinextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Puntos aleatorios en límites de capa

Crea una nueva capa de puntos con un número de puntos aleatorios dado, todos ellos dentro de la extensión de la capa dada.

Una distancia mínima puede ser especificada, para evitar que los puntos estén muy cerca unos de otros.

menú predeterminado: *Vectorial ► Herramientas de Investigación*

> **Advertencia**
>
> Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa poligonal de entrada definiendo el área
Número de puntos	`POINTS_NUMBER`	[numérico: entero] Predeterminado: 1	Número de puntos a crear
Distancia mínima entre puntos	`MIN_DISTANCE`	[numérico: doble] Predeterminado: 0.0	La distancia mínima entre puntos

continúe en la próxima página

Tabla 24.520 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Puntos aleatorios	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Los puntos aleatorios de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos aleatorios	`OUTPUT`	[vector: punto]	La capa de puntos aleatorios saliente.

Código Python

Algoritmo ID: `qgis:randompointsinlayerbounds`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Puntos aleatorios en polígonos

Crea una capa de puntos con puntos ubicados dentro de los polígonos de otra capa.

Para cada entidad (polígono/multi-polígono) en la capa de entrada, el número de puntos especificado se añade a la capa resultante.

Se puede especificar el número de puntos a generar por geometría y la distancia mínima entre ellos para evitar que queden demasiado cerca unos de otros en la capa de salida. Si se especifica una distancia mínima, puede ser que no se pueda generar el número de puntos pretendido para cada entidad. En cualquier caso, en la información de salida del algoritmo se incluye el número de puntos generados y los que faltarían.

La siguiente figura muestra el efecto del número de puntos por entidad y la distancia mínima global, con distancias mínimas de cero o no-cero (utilizando el mismo valor «semilla», para que al menos el primer punto generado sea el mismo).

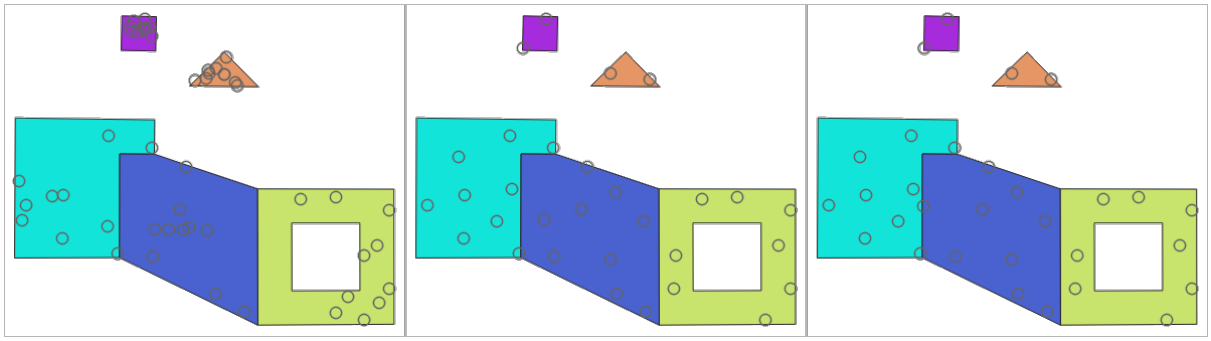


Figura 24.83: Diez puntos por polígono, *izquierda*: distancia mínima = 0, *centro*: distancia mínima = 1, *derecha*: distancia mínima = 1, distancia mínima global = 0

Se puede especificar el número de intentos por punto. Esto solo es relevante en caso de establecer una distancia mínima mayor de cero.

Se puede establecer un valor «semilla» para el generador de números aleatorios, de tal manera que se obtengan un número idéntico de secuencias aleatorias cada vez que se ejecute el algoritmo.

Se pueden incluir los atributos del polígono en el que se haya generado un punto (*Incluir atributos del polígono*).

Si quiere obtener una densidad de puntos aproximadamente igual en todas las entidades, puede definir el número de puntos en función de la geometría de la entidad poligonal.

Ver también

Puntos aleatorios dentro de polígonos

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada poligonal	`INPUT`	[vectorial: lineal]	Capa vectorial de polígonos entrante
Número de puntos en cada entidad	`POINTS_NUMBER`	[numérico: entero] Predeterminado: 1	Número de puntos a crear
Distancia mínima entre puntos Opcional	`MIN_DISTANCE`	[numérico: doble] Predeterminado: 0.0	La distancia mínima entre puntos dentro de un mismo polígono.
Puntos aleatorios en polígonos	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Los puntos aleatorios de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Distancia global mínima entre puntos Opcional	DISTANCIA_MÍNI	[numérico: doble] Predeterminado: 0.0	La distancia global mínima entre puntos. Debe ser menor que la *Distancia mínima entre puntos (dentro de cada entidad)* para que ese parámetro tenga efecto.
Número máximo de intentos de búsqueda (para una distancia mínima > 0) Opcional	NÚMERO_MÁXIMO_	[numérico: entero] Predeterminado: 10	Número máximo de intentos por punto. Solo es relevante si se establece la distancia mínima entre puntos (y mayor de 0).
Valor «semilla» para el generador de números aleatorios Opcional	VALOR_SEMILLA	[numérico: entero] Predeterminado: No establecido	Valor «semilla» para utilizar en el generador de números aleatorios.
Incluye los atributos del polígono	INCLUIR_ATRIBU	[booleano] Predeterminado: Verdadero	Si se establece, el punto toma los atributos de la línea sobre la que se sitúa.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos aleatorios en polígonos	OUTPUT	[vector: punto]	La capa de puntos aleatorios saliente.
Número de entidades con geometría vacía o ausente	ENTIDADES_CON_	[numérico: entero]	
Número total de puntos generados	PUNTOS_DE_SALI	[numérico: entero]	
Número de puntos omitidos	PUNTOS_OMITIDO	[numérico: entero]	El número de puntos que no ha podido generarse a causa de la limitación establecida por la distancia mínima entre ellos.
Número de entidades con puntos omitidos	POLÍGONOS_CON_	[numérico: entero]	No se incluyen entidades con geometría vacía o ausente

Código Python

Algoritmo ID: `native:randompointsinpolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Puntos aleatorios dentro de polígonos

Crea una nueva capa de puntos con un número determinado de puntos aleatorios dentro de cada polígono de la capa de polígono de entrada.

Dos estrategias de muestreo están disponibles:

- Recuento de puntos: número de puntos para cada entidad
- Densidad de puntos: densidad de puntos para cada entidad

Una distancia mínima puede ser especificada, para evitar que los puntos estén muy cerca unos de otros.

menú predeterminado: *Vectorial ► Herramientas de Investigación*

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Puntos aleatorios en polígonos

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa vectorial de polígonos entrante
Estrategia de muestreo	`STRATEGY`	[enumeración] Predeterminado: 0	Estrategia de muestreo a usar. Una de: • 0 — Número de puntos: número de puntos para cada entidad • 1 — Densidad de puntos: densidad de puntos para cada entidad
Recuento de puntos o densidad	`VALUE`	[numérico: doble] Predeterminado: 1.0	El número o densidad de puntosm dependiendo de la elegida *Estrategia de Muestreo*.
Distancia mínima entre puntos	`MIN_DISTANCE`	[numérico: doble] Predeterminado: 0.0	La distancia mínima entre puntos

continúe en la próxima página

Tabla 24.525 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Puntos aleatorios	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Los puntos aleatorios de salida. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos aleatorios	`OUTPUT`	[vector: punto]	La capa de puntos aleatorios saliente.

Código Python

Algoritmo ID: `qgis:randompointsinsidepolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Puntos aleatorios sobre líneas

Crea una capa de puntos con puntos ubicados sobre las líneas de otra capa.

Para cada entidad (línea / multi-línea) de la capa de entrada, se añade el número de puntos especificado en la capa resultante.

Se puede especificar el número de puntos a generar por geometría y la distancia mínima entre ellos para evitar que queden demasiado cerca unos de otros en la capa de salida. Si se especifica una distancia mínima, puede ser que no se pueda generar el número de puntos pretendido para cada entidad. En cualquier caso, en la información de salida del algoritmo se incluye el número de puntos generados y los que faltarían.

La siguiente figura muestra el efecto del número de puntos por entidad y la distancia mínima global, con distancias mínimas de cero o no-cero (utilizando el mismo valor «semilla», para que al menos el primer punto generado sea el mismo).



Figura 24.84: Cinco puntos por entidad lineal, *izquierda*: distancias mínimas = 0, *centro*: distancias mínimas != 0, *derecha*: distancia mínima != 0, distancia mínima global = 0

Se puede especificar el número de intentos por punto. Esto solo es relevante en caso de establecer una distancia mínima mayor de cero.

Se puede establecer un valor «semilla» para el generador de números aleatorios, de tal manera que se obtengan un número idéntico de secuencias aleatorias cada vez que se ejecute el algoritmo.

Los traibutos de la entidad lineal sobre la que se ha generado un punto pueden añadirse (*Incluir atributos de la línea*).

Si lo que quiere es obtener aproximadamente la misma densidad de puntos por línea, puede definir el número de puntos en función de la longitud de la entidad lineal.

Ver también

Puntos aleatorios a lo largo de la línea

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de líneas de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Número de puntos en cada entidad	`POINTS_NUMBER`	[numérico: entero] Predeterminado: 1	Número de puntos a crear

continúe en la próxima página

Tabla 24.527 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Distancia mínima entre puntos (por entidad) Opcional	MIN_DISTANCE	[numérico: doble] Predeterminado: 0.0	La distancia mínima entre puntos dentro de la misma entidad lineal.
Puntos aleatorios sobre líneas	OUTPUT	[vector: punto] Predeterminado: [Crear capa temporal]	Los puntos aleatorios de salida. *Uno de*: • Crear capa temporal (TEMPORARY_OUTPUT) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Distancia global mínima entre puntos Opcional	DISTANCIA_MÍNI	[numérico: doble] Predeterminado: 0.0	La distancia global mínima entre puntos. Debe ser menor que la *Distancia mínima entre puntos (dentro de cada entidad)* para que ese parámetro tenga efecto.
Número máximo de intentos de búsqueda (para una distancia mínima > 0) Opcional	NÚMERO_MÁXIMO_	[numérico: entero] Predeterminado: 10	Número máximo de intentos por punto. Solo es relevante si se establece la distancia mínima entre puntos (y mayor de 0).
Valor «semilla» para el generador de números aleatorios Opcional	VALOR_SEMILLA	[numérico: entero] Predeterminado: No establecido	Valor «semilla» para utilizar en el generador de números aleatorios.
Incluir atributos de línea	INCLUIR_ATRIBU	[booleano] Predeterminado: Verdadero	Si se establece, el punto toma los atributos de la línea sobre la que se sitúa.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos aleatorios sobre líneas	OUTPUT	[vector: punto]	La capa de puntos aleatorios saliente.
Número de entidades con geometría vacía o ausente	ENTIDADES_CON_	[numérico: entero]	
Número de entidades con puntos omitidos	LÍNEAS_CON_PUN	[numérico: entero]	No se incluyen entidades con geometría vacía o ausente

continúe en la próxima página

Tabla 24.529 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Número total de puntos generados	`POINTS_GENERAT`	[numérico: entero]	
Número de puntos omitidos	`PUNTOS_OMITIDO`	[numérico: entero]	El número de puntos que no ha podido generarse a causa de la limitación establecida por la distancia mínima entre ellos.

Código Python

Algoritmo ID: `native:randompointsonlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Pixels ráster a puntos

Crea una capa vectorial de puntos correspondientes a cada píxel en una capa ráster.

Convierte una capa ráster en una capa vectorial, creando objetos espaciales punto para cada centro de píxel individual en la capa ráster. Los píxeles NoData se omiten en la salida.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa ráster	`INPUT_RASTER`	[ráster]	Capa ráster de entrada
Número de banda	`RASTER_BAND`	[banda ráster]	Banda ráster de la que extraer puntos
Nombre de campo	`FIELD_NAME`	[cadena de texto] Predeterminado: “VALUE”	Nombre del campo para almacenar el valor de la banda ráster
Vectorial de puntos	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa de puntos resultante de los centroides de píxeles. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Vectorial de puntos	`OUTPUT`	[vector: punto]	La capa de puntos resultante con los centroides de los pixels

Código Python

Algoritmo ID: `native:pixelstopoints`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Pixels ráster a polígonos

Crea una capa vectorial de polígonos correspondiente a cada píxel en una capa ráster.

Convierte una capa ráster en una capa vectorial, creando objetos espaciales polígono para la extensión de cada píxel individual en la capa ráster. Los píxeles NoData se omiten en la salida.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa ráster	`INPUT_RASTER`	[ráster]	Capa ráster de entrada
Número de banda	`RASTER_BAND`	[banda ráster]	Banda ráster de la que extraer puntos
Nombre de campo	`FIELD_NAME`	[cadena de texto] Predeterminado: “VALUE”	Nombre del campo para almacenar el valor de la banda ráster
Vectorial poligonal	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de polígonos resultante de extensiones de píxeles. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Vectorial poligonal	`OUTPUT`	[vectorial: poligonal]	Capa poligonal resultante de extensiones de pixeles

Código Python

Algoritmo ID: `native:pixelstopolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Puntos regulares

Crea una nueva capa de puntos con sus puntos colocados en una cuadrícula regular dentro de una extensión determinada.

La cuadrícula se especifica por el espacio entre los puntos (el mismo espacio para todas las dimensiones) o por el número de puntos a generar. En el último caso, el espaciado se determinará a partir de la extensión. Para generar una cuadrícula rectangular completa, se genera al menos el número de puntos especificado por el usuario para este último caso.

Se pueden aplicar desplazamientos aleatorios al espaciado de puntos, lo que da como resultado un patrón de puntos no regular.

menú predeterminado: *Vectorial ► Herramientas de Investigación*

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Extensión de entrada (xmin, xmax, ymin, ymax)	`EXTENT`	[extensión]	Extensión de mapa para los puntos aleatorios Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Espaciado de puntos/recuento	`SPACING`	[numérico: entero] Predeterminado: 100	Espaciado entre los puntos, o el número de puntos, dependiendo de si `Usar espaciado de puntos` está marcado o no.
Inserción inicial desde la esquina (lado izquierdo)	`INSET`	[numérico: doble] Predeterminado: 0.0	Desplaza los puntos relativos a la esquina superior izquierda. El valor se utiliza para los ejes X e Y.
Aplicar desplazamiento aleatorio al espaciado de puntos	`RANDOMIZE`	[booleano] Predeterminado: False	Si se marca, los puntos tendrán un espaciado aleatorio
Usar espaciado de puntos	`IS_SPACING`	[booleano] Predeterminado: Verdadero	Si está desmarcado el espaciado de puntos no se tiene en cuenta
SRC de la capa de salida	`CRS`	[src] Predeterminado: *SRC de Proyecto*	SRc de la capa de puntos aleatorios
Puntos regular	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa de puntos regulares de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos regular	`OUTPUT`	[vector: punto]	La capa de puntos regulares saliente.

Código Python

Algoritmo ID: `qgis:regularpoints`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.25 Vectorial general

Asignar proyección

Asigna una nueva proyección a una capa vectorial.

Crea una nueva capa con exactamente las mismas entidades y geometrías que la entrada, pero asignada a un nuevo CRS. Las geometrías son **no** reproyectadas, solo se asignan a un CRS diferente.

Este algoritmo puede ser usado para reparar capas a las que se les ha asginado una proyección incorrecta.

Los atributos no son modificados por este algoritmo.

Ver también

Define projection, *Encontrar proyección*, *Capa reproyectada*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa vectorial con CRS erróneo o desaparecido
CRS Asignado	`CRS`	[src] Predeterminado: `EPSG:4326 – WGS84`	Seleccione el nuevo SRC a asignar a la capa vectorial

continúe en la próxima página

Tabla 24.536 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
CRS Asignado Opcional	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
CRS Asignado	`OUTPUT`	[la misma que la entrada]	Capa vectorial con proyección asignada

Código Python

Algoritmo ID: `native:assignprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Geocodificador por lotes de Nominatim

Realiza la geocodificación por lotes utilizando el servicio Nominatim contra un campo de cadena de capa de entrada. La capa de salida tendrá una geometría de puntos que reflejará la ubicación geocodificada, así como una serie de atributos asociados a la ubicación geocodificada.

Permite la modificación in situ de *características* de objetos espaciales punto

Nota

Este algoritmo cumple con la usage policy del servicio de geocodificación Nominatim proporcionado por OpenStreetMap Foundation.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial para geocodificar las entidades
Campo de dirección	`FIELD`	[campo de tabla: cadena]	Campo que contiene las direcciones a geocodificar
Geocodificado	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida que contiene sólo las direcciones geocodificadas. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Geocodificado	`OUTPUT`	[vector: punto]	Capa vectorial con entidades de puntos correspondientes a las direcciones geocodificadas

Código Python

Identificación del algoritmo: `native:batchnominatimgeocoder`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Convertir capa a marcadores espaciales

Crea marcadores espaciales correspondientes a la extensión de las entidades contenidas en una capa.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vectorial: lineal, polígonal]	La capa vectorial entrante
Marcador de Destino	`DESTINATION`	[enumeración] Predeterminado: 0	Seleccione el destino para los marcadores. Uno de: • 0 — Marcadores de proyecto • 1 — Marcadores de Usuario
Campo Nombre	`NAME_EXPRESSIO`	[expresión]	Campo o expresión que dará nombres a los marcadores generados
Campo Grupo	`GROUP_EXPRESSI`	[expresión]	Campo de expresión que proveerán grupos para los marcadores generados

Salidas

Etiqueta	Nombre	Tipo	Descripción
Recuento de marcadores añadidos	`COUNT`	[numérico: entero]	

Código Python

Algoritmo ID: `native:layertobookmarks`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Convertir marcadores espaciales a capa

Crea una nueva capa conteniendo entidades poligonales para los marcadores espaciales almacenados. La exportación se puede filtrar solo a marcadores que pertenezcan al proyecto actual, a todos los marcadores de usuario o a una combinación de ambos.

Parametros

Etiqueta	Nombre	Tipo	Descripción
fuente Marcador	`SOURCE`	[enumeración] [lista] Predeterminado: [0,1]	Seleccione la fuente(s) de los marcadores. Uno o mas de: • 0 — Marcadores de proyecto • 1 — Marcadores de Usuario
SRC saliente	`CRS`	[src] Predeterminado: `EPSG:4326 - WGS 84`	El SRC de la capa saliente
Salida	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifica la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Salida	`OUTPUT`	[vectorial: poligonal]	La capa vectorial saliente (marcadores)

Código Python

Algoritmo ID: `native:bookmarkstolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Crear índice de atributos

Crea un índice en un campo de la tabla de atributos para acelerar las consultas. El soporte para la creación de índices depende tanto del proveedor de datos de la capa como del tipo de campo.

No se crean salidas: el índice es alamcenado en la misma capa.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vectorial: cualquiera]	Seleccione la capa vectorial para la que quiera crear un índice de atributos
Atributo a índice	`FIELD`	[campo de tabla: cualquiera]	Campo de capa vectorial

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa Indexada	`OUTPUT`	[la misma que la entrada]	Una copia de la capa vectorial entrante con un índice para el campo especificado

Código Python

Algoritmo ID: `native:createattributeindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Crear índice espacial

Crea un índice para acelerar el acceso a las entidades en una capa en función de su ubicación espacial. El soporte para la creación de índice espacial depende del proveedor de datos de la capa.

No se crean nuevas capas salientes.

Menú predeterminado: *Vectorial ► Herramientas de Administración de Datos*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vector: geometría]	Capa de vector de entrada

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa indexada	`OUTPUT`	[la misma que la entrada]	Una copia de la capa vectorial saliente con un índice espacial

Código Python

Algoritmo ID: `native:createspatialindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Define projection

Sets an existing layer's projection to the provided CRS without reprojecting features.

Al contrario que el algoritmo *Asignar proyección*, modifica la capa actual y no generará una nueva capa.

Nota

Para los conjuntos de datos de Shapefile, los archivos `.prj` y `.qpj` se sobrescribirán, o se crearán si faltan, para que coincidan con el SRC proporcionado.

Menú predeterminado: *Vectorial* ► *Herramientas de Administración de Datos*

Ver también

Asignar proyección, *Encontrar proyección*, *Capa reproyectada*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa vectorial con información de proyección perdida
SRC	`CRS`	[src]	Seleccione el SRC a asignar a la capa vectorial

Salidas

Etiqueta	Nombre	Tipo	Descripción
	`INPUT`	[la misma que la entrada]	La capa vectorial entrante con la proyección definida

Código Python

Algorithm ID: `native:definecurrentprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Borrar geometrías duplicadas

Encuentra y borra geometrías duplicadas.

Los atributos no están marcados, por lo que en caso de que dos entidades tengan geometrías idénticas pero atributos diferentes, solo se agregará una de ellas a la capa de resultados.

Nota

Este algoritmo no requiere geometrías válidas como entrada.

Ver también

Descartar geometrías, *Borra geometrías nulas*, *Borrar duplicados por atributo*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	La capa con geometrías duplicadas que desee limpiar
Limpia	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida que contiene los objetos espaciales únicos. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Número de registros duplicados descartados	`DUPLICATE_COUN`	[numérico: entero]	Recuento de registros duplicados descartados
Limpia	`OUTPUT`	[la misma que la entrada]	La capa saliente sin ningunas geometrías duplicadas
Recuento de registros conservados	`RETAINED_COUNT`	[numérico: entero]	Recuento de registros únicos

Código Python

Algoritmo ID: `native:deleteduplicategeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Borrar duplicados por atributo

Deletes duplicate rows by only considering the specified field(s). The first matching row will be retained, and duplicates will be discarded.

Opcionalmente, estos registros duplicados se pueden guardar en una salida separada para su análisis.

Ver también

Borrar geometrías duplicadas

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	La capa entrante
Campos a coincidir para duplicadas	`FIELDS`	[campo de tabla: cualquiera] [lista]	Campos que definen duplicados. Las entidades con valores idénticos para todos estos campos se consideran duplicados.
Filtrados (no duplicados)	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida que contiene los objetos espaciales únicos. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Filtradas (duplicadas) Opcional	`DUPLICATES`	[la misma que la entrada] Predeterminado: `[Omitir salida]`	Especifique la capa de salida que contiene sólo los duplicados. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Filtradas (duplicadas)	`DUPLICATES`	[la misma que la entrada]	Vector layer containing the duplicate features.
Número de registros duplicados descartados	`DUPLICATE_COUN`	[numérico: entero]	Recuento de registros duplicados descartados
Filtrados (no duplicados)	`OUTPUT`	[la misma que la entrada]	Capa vectorial que contiene las entidades únicas.
Recuento de registros conservados	`RETAINED_COUNT`	[numérico: entero]	Recuento de registros únicos

Código Python

Algoritmo ID: `native:removeduplicatesbyattribute`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Detectar cambios del conjunto de datos

Compara dos capas vectoriales y determina qué entidades no tienen cambios, se agregan o se eliminan entre las dos. Está diseñado para comparar dos versiones diferentes del mismo conjunto de datos.

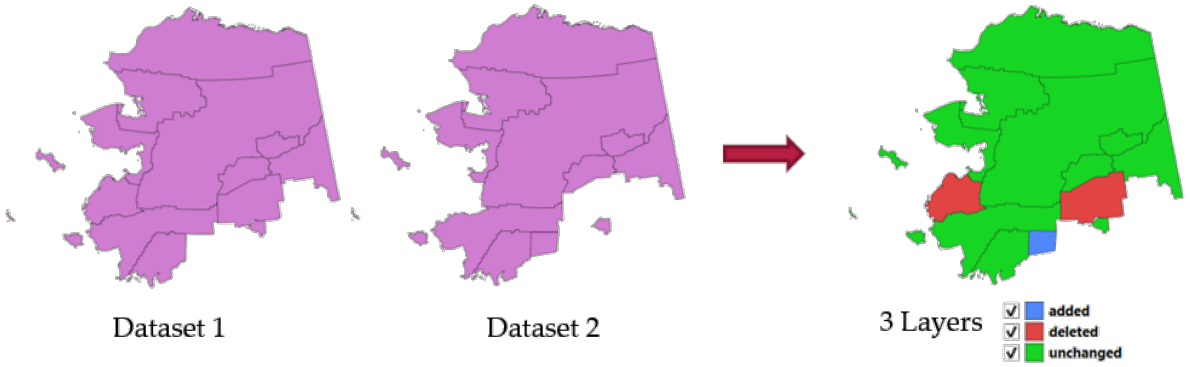


Figura 24.85: Ejemplo de detección de cambios en el conjunto de datos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa original	`ORIGINAL`	[vector: geometría]	La capa vectorial considerada como versión original
Capa revisada	`REVISED`	[vector: geometría]	La capa vectorial revisada o modificada
Atributos a considerar para igualdad Opcional	`COMPARE_ATTRIB`	[campo de tabla: cualquiera] [lista]	Atributos a considerar para coincidencia. Por defecto, todos los atributos son comparados.
Comportamiento de comparación de Geometría Opcional	`MATCH_TYPE`	[enumeración] Predeterminado: 1	Define el criterio de la comparación. Opciones: • 0 — Coincidencia Exacta: incluye el orden y recuento de vértices de geometrías • 1 — Tolerancia de coincidencia (Iguadad Topológica): Las geometrías son consideradas iguales

continúe en la próxima página

Tabla 24.554 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Entidades Inalteradas Opcional	`UNCHANGED`	[vectorial: misma que la capa Original]	Especifica la capa del vector de salida que contiene los objetos espaciales no modificados. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Entidades agregadas Opcional	`ADDED`	[vectorial: misma que la capa Original]	Especifica la capa vectorial de salida que contiene los objetos espaciales añadidos. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Entidades borradas Opcional	`DELETED`	[vectorial: misma que la capa Original]	Especifica la capa del vector de salida que contiene los objetos espaciales eliminados. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Entidades Inalteradas	`UNCHANGED`	[vectorial: misma que la capa Original]	Capa vectorial contenedora de las entidades inalteradas.
Entidades agregadas	`ADDED`	[vectorial: misma que la capa Original]	Capa vectorial contenedora de la entidades agregadas.
Entidades borradas	`DELETED`	[vectorial: misma que la capa Original]	Capa vectorial contenedora de las entidades borradas.
Recuento de entidades inalteradas	`UNCHANGED_COUN`	[numérico: entero]	Recuento de entidades inalteradas.

continúe en la próxima página

Tabla 24.555 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Recuento de entidades añadidas en capa revisada	`ADDED_COUNT`	[numérico: entero]	Recuento de entidades agregadas en capa revisada.
Recuento de entidades borradas de la capa original	`DELETED_COUNT`	[numérico: entero]	Recuento de entidades borradas de la capa original.

Código Python

Algoritmo ID: `native:detectvectorchanges`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Descartar geometrías

Crea una copia simple *sin geometría* de la tabla de atributos de la capa de entrada. Mantiene la tabla de atributos de la capa de origen.

Si el archivo está guardado en una carpeta local, puede elegir entre muchos formatos de archivo.

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Ver también

Borrar geometrías duplicadas, *Borra geometrías nulas*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	La capa vectorial entrante
Geometrías descartadas	`OUTPUT`	[vector: tabla]	Especifique la capa sin geometría de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Geometrías descartadas	`OUTPUT`	[vector: tabla]	La capa sin geometría de salida. Una copia de la tabla de atributos original.

Código Python

Algoritmo ID: `native:dropgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Ejecutar SQL

Ejecuta una consulta simple o compleja basada únicamente en SELECT con sintaxis `SQL` en la capa de origen.

Fuentes de datos de entrada son identificadas como `input1`, `input2`... `inputN` y una consulta simple se verá como `SELECT * FROM input1`.

Aparte de una consulta simple, puede agregar expresiones o variables dentro del parámetro `consulta SQL` en sí mismo. Ésto es particulamente útil si este algoritmo es ejecutado dentro de un modelo Procesos y usted quiere usar una entrada de modelo como un parámetro de la consulta. Un ejemplo de una consulta sería entonces `SELECT * FROM [% @table %]` donde `@table` es la variable que identifica la entrada del modelo.

El resultado de la consulta se añadirá como una nueva capa.

Ver también

SpatiaLite ejecuta SQL, *PostgreSQL ejecuta SQL*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Fuentes de datos de entrada adicionales (llamadas input1, .., inputN en la consulta)	`INPUT_DATASOUR`	[vectorial: cualquiera] [lista]	Lista de capas a consultar. En el editor SQL puede referir estas capas con su nombre **real** o también con **input1**, **input2**, **inputN** dependiendo de cuántas capas se hayan elegido.
****Consulta SQL*	`INPUT_QUERY`	[cadena de texto]	Escribe la cadena de tu consulta SQL. p.ejemplo `SELECT * FROM input1`.
Campo de identificador único Opcional	`INPUT_UID_FIEL`	[cadena de texto]	Especifica la columna con ID único

continúe en la próxima página

Tabla 24.558 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Campo geometría Opcional	`INPUT_GEOMETRY`	[cadena de texto]	Especifica el campo geometría
Tipo de geometría Opcional	`INPUT_GEOMETRY`	[enumeración] Predeterminado: 0	Elija la geometría del resultado. Por defecto, el algoritmo lo detectará automáticamente. Uno de: • 0 — Autodetectar • 1 — Sin geometría • 2 — Puntos • 3 — Cadena lineal • 4 — Poligonal • 5 — MultiPunto • 6 — Cadena Multilínea • 7 — MultiPoligonal
SRC Opcional	`INPUT_GEOMETRY`	[src]	El SRC a asignar a la capa saliente
Salida SQL	`OUTPUT`	[vector: geometría] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida creada por la consulta. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Salida SQL	`OUTPUT`	[vector: geometría]	Capa vectorial creada por la consulta

Código Python

Algoritmo ID: `qgis:executesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Exportar capas a DXF

Exporta capas a archivo DXF. Para cada capa, puede elegir un campo cuyos valores se utilizan para dividir entidades en capas de destino generadas en la salida DXF.

Ver también

Creando nuevos archivos DXF

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`LAYERS`	[vector: geometría] [lista]	Lista de capas vectoriales de entrada con opciones asociadas (rellenada como un elemento `QgsProcessingParameterDxfLayers` — hecho en GUI pulsando el botón *Configure Layer…* para cada capa seleccionada): **Capa [cadena] (`capa`)** Ruta completa de la capa de entrada a exportar **Atributo de la capa de salida [tablefield: any] (`attributeIndex`)** Índice de atributos para dividir la capa de entrada utilizando valores únicos **Nombre de la capa de salida [cadena] (`overriddenLayerName`)** Nombre de capa que se utilizará en el archivo DXF exportado (la capa original del proyecto permanece intacta) **Permitir bloques de símbolos definidos por datos** [boolean] (`buildDataDefinedBlocks`) **Número máximo de bloques de símbolos [numérico: entero] (`dataDefinedBlocksMaximumNumberOfClasses`)** `-1` significa sin limitación.
Modo de simbología	`SYMBOLOGY_MODE`	[enumeración] Predeterminado: 0	Tipo de simbología para aplicar a las capas de salida. Puedes elegir entre: • 0 — Sin simbología • 1 — Simbología de funciones • 2 — Simbología de capa de símbolo
Escala de simbología	`SYMBOLOGY_SCAL`	[escala] Predeterminado: 1:1 000 000	Escala predeterminada de exportación de datos.
Tema del mapa Opcional	`MAP_THEME`	[tema del mapa]	Haga coincidir el estilo de las capas con el tema del mapa proporcionado.
Codificación	`ENCODING`	[enumeración]	Codificación para aplicar a las capas.
SRC	`CRS`	[src]	Elija el CRS para la capa de salida.

continúe en la próxima página

Tabla 24.560 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extensión Opcional	`EXTENT`	[extensión]	Limita los elementos exportados a aquellos con geometrías que intersecan la extensión proporcionada.
Usar título de capa como nombre	`USE_LAYER_TITL`	[booleano] Predeterminado: Falso	Nombra la capa de salida con el título de la capa (como se establece en los metadatos de la capa o en las propiedades del servidor QGIS) en lugar del nombre de la capa.
Fuerza 2D	`FORCE_2D`	[booleano] Predeterminado: Falso	
Exportar etiquetas como elementos MTEXT	`MTEXT`	[booleano] Predeterminado: Verdadero	Exporta etiquetas como elementos MTEXT o TEXT
Utilizar sólo funciones seleccionadas	`SELECTED_FEATU`	[booleano] Predeterminado: Falso	Exporta sólo los objetos espaciales seleccionados.
DXF	`OUTPUT`	[archivo] Predeterminado: `[Save to temporary file]`	Especificación del archivo DXF de salida. *Una de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
DXF	`OUTPUT`	[archivo]	`.DXF` archivo que contiene las capas de entrada

Código Python

Algoritmo ID: `native:dxfexport`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Extraer entidades seleccionadas

Guarda las entidades seleccionadas como una nueva capa.

> **Nota**
>
> Si la capa seleccionada no tiene objetos seleccionados, la capa recién creada estará vacía.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vectorial: cualquiera]	Capa de la cuál guardar la selección
Entidades seleccionadas	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial para los objetos espaciales seleccionados. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Entidades seleccionadas	`OUTPUT`	[la misma que la entrada]	Capa vectorial con solo las entidades seleccionadas, o ninguna entidad si no se seleccionó ninguna.

Código Python

Algoritmo ID: `native:saveselectedfeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Extraer codificación archivo shape

Extrae la información de codificación de atributos incrustada en un archivo shape. Se consideran tanto la codificación especificada por un archivo `.cpg` opcional como cualquier detalle de codificación presente en el bloque de encabezado LDID `.dbf`.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vector: geometría]	Archivo shape de ESRI (`.SHP`) Capa para extraer la información de codificación.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Codificación de archivo shape	`ENCODING`	[cadena de texto]	Información de codificación especificada en el archivo de entrada
** Codificación CPG**	`CPG_ENCODING`	[cadena de texto]	Codificar información especificada en cualquier archivo `.CPG` opcional
Codificación LDID	`LDID_ENCODING`	[cadena de texto]	Codificar información especificada en bloque encabezado `.dbf` LDID

Código Python

ID Algoritmo: `native:shpencodinginfo`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Encontrar proyección

Crea una lista corta de sistemas de referencia de coordenadas candidatos, por ejemplo, para una capa con una proyección desconocida.

El área que se espera que cubra la capa debe especificarse mediante el parámetro de área de destino. El sistema de referencia de coordenadas para esta área objetivo debe ser conocido por QGIS.

El algoritmo opera probando la extensión de la capa en todos los sistemas de referencia conocidos y luego enumerando aquellos para los cuales los límites estarían cerca del área objetivo si la capa estuviera en esta proyección.

Ver también

Asignar proyección, *Define projection*, *Capa reproyectada*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vector: geometría]	Capa con proyección desconocida
Área objetivo para la capa (xmin, xmax, ymin, ymax)	`TARGET_AREA`	[extensión]	El área que cubre la capa. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC Candidatos	`OUTPUT`	[vector: tabla] Predeterminado: `[Crear capa temporal]`	Especifique la tabla (capa sin geometría) para las sugerencias SRC (códigos EPSG). *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
SRC Candidatos	`OUTPUT`	[vector: tabla]	Una tabla con todos los SRC (códigos EPSG) de los criterios de coincidencia.

Código Python

Algorithm ID: `native:findprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Emparejar Relación

Empareja una *relationship* para una capa vectorial, exportando una sola capa que contiene una característica principal por característica secundaria relacionada. Esta función principal contiene todos los atributos de las funciones relacionadas. Esto permite tener la relación como una tabla simple que puede ser p. Ej. exportado a CSV.

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

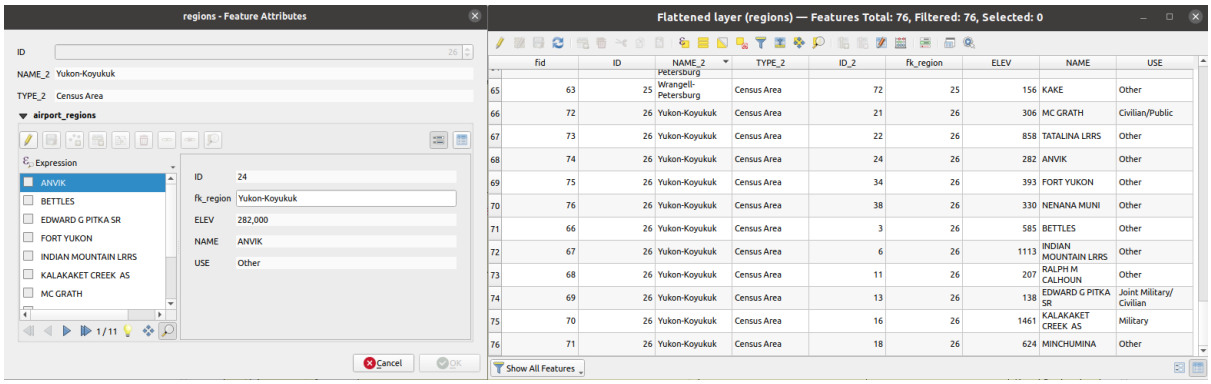


Figura 24.86: Forma de una región con elementos secundarios relacionados (izquierda)- una característica de región duplicada para cada elemento secundario relacionado, con atributos unidos (derecha)

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vectorial: cualquiera]	Capa con la relación que debe ser normalizada
Capa emparejada Opcional	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (aplanada). :ref:`Una de • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa emparejada	`OUTPUT`	[la misma que la entrada]	Una capa contenedora de las entidades maestras con todos los atributos para las entidades relatadas

Código Python

ID Algoritmo: `native:flattenrelationships`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Unir atributos por valor de campo

Toma una capa de vector de entrada y crea una nueva capa de vector que es una versión extendida de la de entrada, con atributos adicionales en su tabla de atributos.

Los atributos adicionales y sus valores se toman de una segunda capa vectorial. En cada uno de ellos se selecciona un atributo para definir los criterios de unión.

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Unir atributos por proximidad, *Unir atributos por localización*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada. La capa de salida constará de las entidades de esta capa con atributos de entidades coincidentes en la segunda capa.
Campo de tabla	`FIELD`	[campo de tabla: cualquiera]	Campo de la capa fuente a usar para la unión
Capa de entrada 2	`INPUT_2`	[vectorial: cualquiera]	Capa con la tabla de atributos a unir
Campo de tabla 2	`FIELD_2`	[campo de tabla: cualquiera]	Campo de la segunda capa (unión) que se utilizará para la combinación El tipo de campo debe ser igual (o compatible) con el tipo de campo de la tabla de entrada.
Campos de capa 2 a copiar Opcional	`FIELDS_TO_COPY`	[campo de tabla: cualquiera] [lista]	Seleccione los campos específicos que desea agregar. De forma predeterminada, se agregan todos los campos.

continúe en la próxima página

Tabla 24.570 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de unión	`METHOD`	[enumeración] Predeterminado: 1	El tipo de capa unida final. Uno de: • 0 — Cree una entidad separada para cada objeto coincidente (uno a muchos) • 1 — Tomar atributos de la primera entidad coincidente únicamente (uno a uno)
Descartar registros que no se pueden unir	`DISCARD_NONMAT`	[booleano] Predeterminado: Verdadero	Compruebe si no desea conservar los objetos que no se pudieron unir
Prefijo de campo unido Opcional	`PREFIX`	[cadena de texto]	Agregue un prefijo a los campos unidos para identificarlos fácilmente y evitar la colisión de nombres de campo
Capa unida Opcional	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida para la unión. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Entidades que no se pueden unir de la primera capa Opcional	`NON_MATCHING`	[la misma que la entrada] Predeterminado: `[Omitir salida]`	Especifique la capa del vector de salida para los objetos espaciales no unibles de la primera capa. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Número de entidades unidas de la tabla de entrada	`JOINED_COUNT`	[numérico: entero]	
Entidades que no se pueden unir de la primera capa Opcional	`NON_MATCHING`	[la misma que la entrada]	Capa vectorial con las entidades no coincidentes
Capa unida Opcional	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente con atributos añadidos de la unión

continúe en la próxima página

Tabla 24.571 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Número de entidades que no se pueden unir de la capa de entrada Opcional	`UNJOINABLE_COU`	[numérico: entero]	

Código Python

*Algoritmo ID**: `native:joinattributestable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Unir atributos por localización

Toma una capa de vector de entrada y crea una nueva capa de vector que es una versión extendida de la de entrada, con atributos adicionales en su tabla de atributos.

Los atributos adicionales y sus valores se toman de una segunda capa vectorial. Se aplica un criterio espacial para seleccionar los valores de la segunda capa que se agregan a cada entidad de la primera capa.

Menú predeterminado: *Vectorial ► Herramientas de Administración de Datos*

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Unir atributos por proximidad, *Unir atributos por valor de campo*, *Unir atributos por ubicación (resumen)*

Explorando las relaciones espaciales

Los predicados geométricos son funciones booleanas que se utilizan para determinar la relación espacial que tiene una característica con otra comparando si sus geometrías comparten una porción de espacio y cómo.

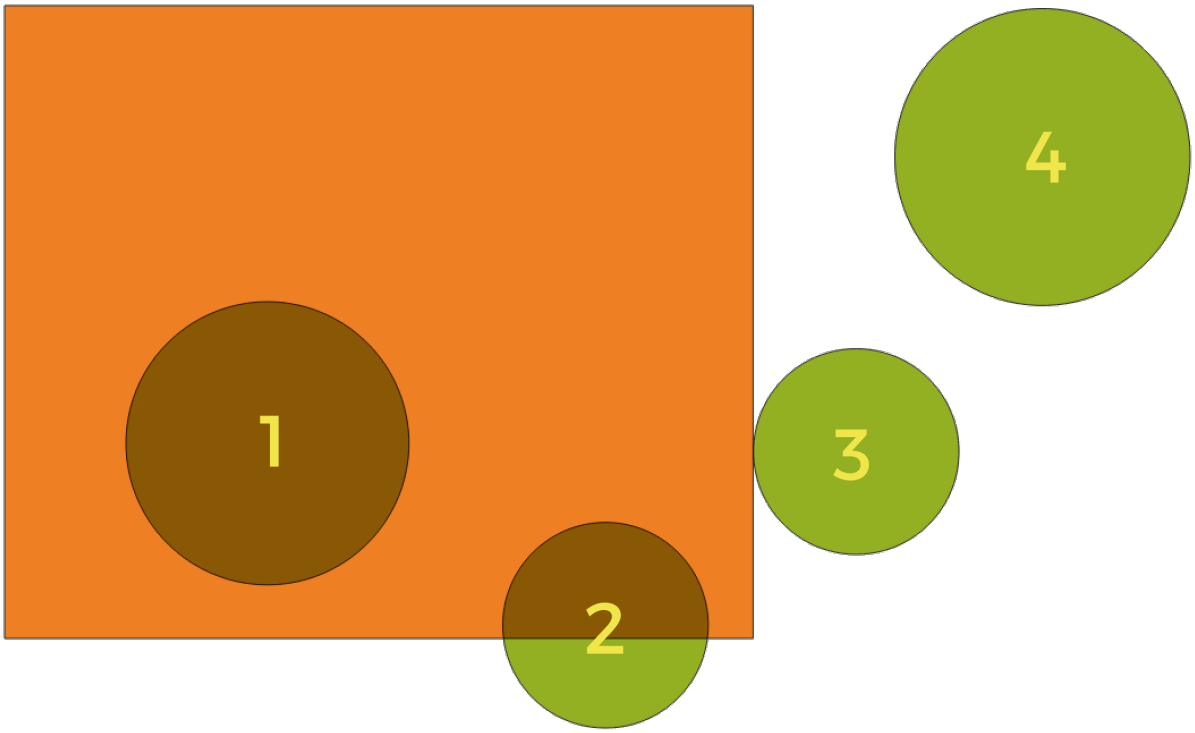


Figura 24.87: Buscando relaciones espaciales entre capas

Usando la figura de arriba, buscamos los círculos verdes comparándolos espacialmente con la característica del rectángulo naranja. Los predicados geométricos disponibles son:

Intersección
Comprueba si una geometría se cruza con otra. Devuelve 1 (verdadero) si las geometrías se intersecan espacialmente (comparten cualquier parte del espacio, se superponen o se tocan) y 0 en caso contrario. En la imagen de arriba, esto devolverá los círculos 1, 2 y 3.

Contiene
Devuelve 1 (verdadero) si y solo si ningún punto de b está en el exterior de a, y al menos un punto del interior de b está en el interior de a. En la imagen, no se devuelve ningún círculo, pero el rectángulo sí lo sería si lo buscaras al revés, ya que contiene el círculo 1 por completo. Esto es lo opuesto a *están dentro*.

Dividir
Devuelve 1 (verdadero) si las geometrías no comparten ninguna parte del espacio (no se superponen, no se tocan). Solo se devuelve el círculo 4.

Igual
Devuelve 1 (verdadero) si y solo si las geometrías son exactamente iguales. No se devolverán círculos.

Tocar
Comprueba si una geometría toca otra. Devuelve 1 (verdadero) si las geometrías tienen al menos un punto en común, pero sus interiores no se cruzan. Solo se devuelve el círculo 3.

Superponer
Comprueba si una geometría se superpone a otra. Devuelve 1 (verdadero) si las geometrías comparten espacio, tienen la misma dimensión, pero no están completamente contenidas entre sí. Solo se devuelve el círculo 2.

Están dentro
Comprueba si una geometría está dentro de otra. Devuelve 1 (verdadero) si la geometría a está completamente dentro de la geometría b. Solo se devuelve el círculo 1.

Cruza
Devuelve 1 (verdadero) si las geometrías proporcionadas tienen algunos puntos interiores en común, pero no todos, y el cruce real tiene una dimensión más baja que la geometría proporcionada más alta. Por ejemplo,

una línea que cruza un polígono se cruzará como una línea (verdadero). Dos líneas que se cruzan se cruzarán como un punto (verdadero). Dos polígonos se cruzan como un polígono (falso). En la imagen, no se devolverán círculos.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Unirse a funciones en	`INPUT`	[vector: geometría]	Capa de vector de entrada. La capa de salida constará de las entidades de esta capa con atributos de entidades coincidentes en la segunda capa.
Donde las funciones	`PREDICATE`	[enumeración] [lista] Predeterminado: [0]	Tipo de relación espacial que el objeto espacial de origen debe tener con la característica de destino para que puedan unirse. Uno o más de: • 0 — intersecta • 1 — contiene • 2 — igual • 3 — tocar • 4 — superponer • 5 — está dentro • 6 — cruzar Si se elige más de una condición, al menos una de ellas (operación OR) debe cumplirse para que se extraiga una entidad.
Por comparación con	`JOIN`	[vector: geometría]	La capa de unión. Las características de esta capa vectorial **añadirán** sus atributos a la tabla de atributos de la capa de origen si satisfacen la relación espacial.
Campos a agregar (déjelo en blanco para usar todos los campos) Opcional	`JOIN_FIELDS`	[campo de tabla: cualquiera] [lista]	Seleccione los campos específicos que desea añadir desde la capa de unión. Por defecto se añaden todos los campos.
Tipo de unión	`METHOD`	[enumeración]	El tipo de capa unida final. Uno de: • 0 — Cree una entidad separada para cada objeto coincidente (uno a muchos) • 1 — Tomar atributos de la primera entidad coincidente únicamente (uno a uno) • 2 — Tomar solo los atributos de la entidad con mayor superposición (uno a uno)
Descartar registros que no se pueden unir	`DISCARD_NONMAT`	[booleano] Predeterminado: Falso	Eliminar de la salida los objetos espaciales de la capa de entrada que no se han podido unir
Prefijo de campo unido Opcional	`PREFIX`	[cadena de texto]	Agregue un prefijo a los campos unidos para identificarlos fácilmente y evitar la colisión de nombres de campo

continúe en la próxima página

Tabla 24.572 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Capa unida Opcional	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida para la unión. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Entidades que no se pueden unir de la primera capa Opcional	`NON_MATCHING`	[la misma que la entrada] Predeterminado: `[Omitir salida]`	Especifique la capa del vector de salida para los objetos espaciales no unibles de la primera capa. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Número de entidades unidas de la tabla de entrada	`JOINED_COUNT`	[numérico: entero]	
Entidades que no se pueden unir de la primera capa Opcional	`NON_MATCHING`	[la misma que la entrada]	Capa vectorial de las entidades no coincidentes
Capa unida	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente con atributos añadidos de la unión

Código Python

Algoritmo ID: `native:joinattributesbylocation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Unir atributos por ubicación (resumen)

Toma una capa de vector de entrada y crea una nueva capa de vector que es una versión extendida de la de entrada, con atributos adicionales en su tabla de atributos.

Los atributos adicionales y sus valores se toman de una segunda capa vectorial. Se aplica un criterio espacial para seleccionar los valores de la segunda capa que se agregan a cada entidad de la primera capa.

El algoritmo calcula un resumen estadístico de los valores de las entidades coincidentes en la segunda capa (por ejemplo, valor máximo, valor medio, etc.).

Ver también

Unir atributos por localización

Explorando las relaciones espaciales

Los predicados geométricos son funciones booleanas que se utilizan para determinar la relación espacial que tiene una característica con otra comparando si sus geometrías comparten una porción de espacio y cómo.

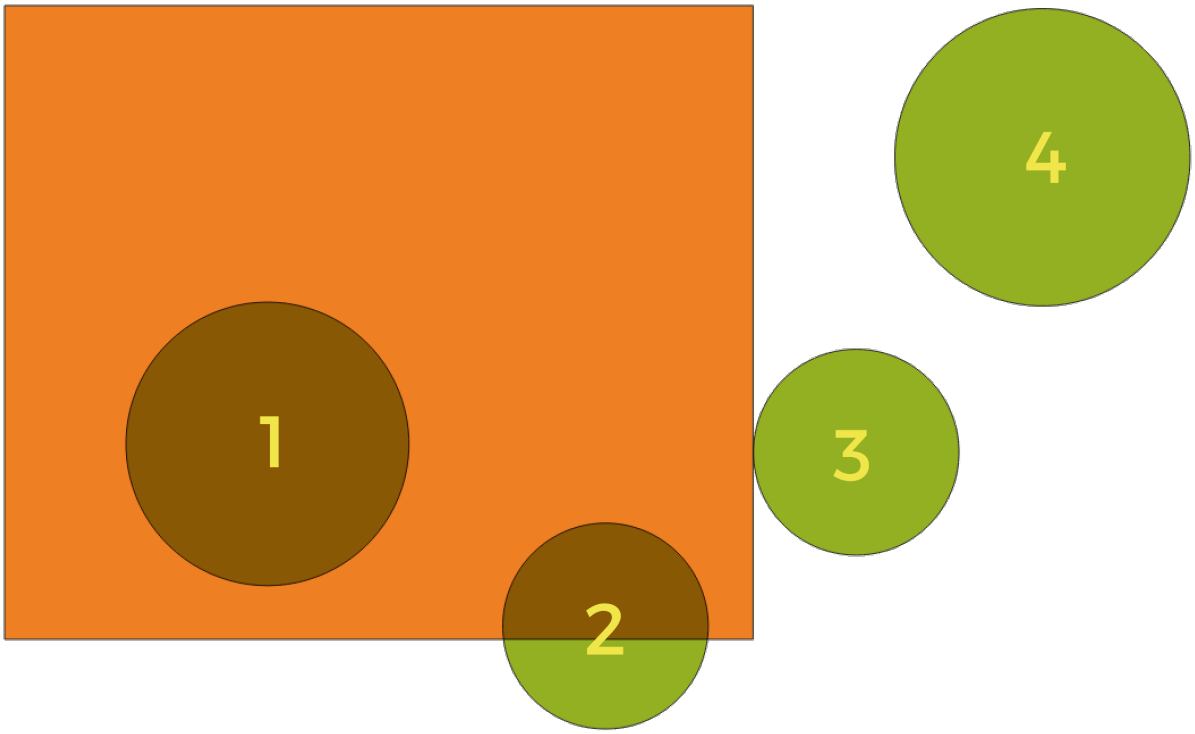


Figura 24.88: Buscando relaciones espaciales entre capas

Usando la figura de arriba, buscamos los círculos verdes comparándolos espacialmente con la característica del rectángulo naranja. Los predicados geométricos disponibles son:

Intersección
: Comprueba si una geometría se cruza con otra. Devuelve 1 (verdadero) si las geometrías se intersecan espacialmente (comparten cualquier parte del espacio, se superponen o se tocan) y 0 en caso contrario. En la imagen de arriba, esto devolverá los círculos 1, 2 y 3.

Contiene
: Devuelve 1 (verdadero) si y solo si ningún punto de b está en el exterior de a, y al menos un punto del interior

de b está en el interior de a. En la imagen, no se devuelve ningún círculo, pero el rectángulo sí lo sería si lo buscaras al revés, ya que contiene el círculo 1 por completo. Esto es lo opuesto a *están dentro*.

Dividir
Devuelve 1 (verdadero) si las geometrías no comparten ninguna parte del espacio (no se superponen, no se tocan). Solo se devuelve el círculo 4.

Igual
Devuelve 1 (verdadero) si y solo si las geometrías son exactamente iguales. No se devolverán círculos.

Tocar
Comprueba si una geometría toca otra. Devuelve 1 (verdadero) si las geometrías tienen al menos un punto en común, pero sus interiores no se cruzan. Solo se devuelve el círculo 3.

Superponer
Comprueba si una geometría se superpone a otra. Devuelve 1 (verdadero) si las geometrías comparten espacio, tienen la misma dimensión, pero no están completamente contenidas entre sí. Solo se devuelve el círculo 2.

Están dentro
Comprueba si una geometría está dentro de otra. Devuelve 1 (verdadero) si la geometría a está completamente dentro de la geometría b. Solo se devuelve el círculo 1.

Cruza
Devuelve 1 (verdadero) si las geometrías proporcionadas tienen algunos puntos interiores en común, pero no todos, y el cruce real tiene una dimensión más baja que la geometría proporcionada más alta. Por ejemplo, una línea que cruza un polígono se cruzará como una línea (verdadero). Dos líneas que se cruzan se cruzarán como un punto (verdadero). Dos polígonos se cruzan como un polígono (falso). En la imagen, no se devolverán círculos.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Unirse a funciones en	`INPUT`	[vector: geometría]	Capa de vector de entrada. La capa de salida constará de las entidades de esta capa con atributos de entidades coincidentes en la segunda capa.
Donde las funciones	`PREDICATE`	[enumeración] [lista] Predeterminado: [0]	Tipo de relación espacial que el objeto espacial de origen debe tener con la característica de destino para que puedan unirse. Uno o más de: • 0 — intersecta • 1 — contiene • 2 — igual • 3 — tocar • 4 — superponer • 5 — está dentro • 6 — cruzar Si se elige más de una condición, al menos una de ellas (operación OR) debe cumplirse para que se extraiga una entidad.
Por comparación con	`JOIN`	[vector: geometría]	La capa de unión. Los objetos espaciales de esta capa vectorial **añadirán** resúmenes de sus atributos a la tabla de atributos de la capa de origen si satisfacen la relación espacial.

continúe en la próxima página

Tabla 24.574 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Campos a resumir (dejar vacío para usar todos los campos) Opcional	`JOIN_FIELDS`	[campo de tabla: cualquiera] [lista]	Seleccione los campos específicos que desea añadir desde la capa de unión. Por defecto se añaden todos los campos.
Resúmenes a calcular (dejar en blanco para usar todos los campos) Opcional	`SUMMARIES`	[enumeración] [lista] Predeterminado: []	Para cada objeto espacial de entrada, se calculan estadísticas sobre campos unidos de sus características coincidentes. Uno o más de: • 0 — recuento • 1 — único • 2 — min • 3 — max • 4 — rango • 5 — suma • 6 — media • 7 — mediana • 8 — desviación típica • 9 — minoría • 10 — mayoría • 11 — q1 • 12 — q3 • 13 — iqr • 14 — vacío • 15 — lleno • 16 — longitud mínima • 17 — longitud máxima • 18 — longitud media
Descartar registros que no se pueden unir	`DISCARD_NONMAT`	[booleano] Predeterminado: Falso	Eliminar de la salida los objetos espaciales de la capa de entrada que no se han podido unir
Capa unida	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida para la unión. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa unida	`OUTPUT`	[la misma que la entrada]	Capa de vector de salida con atributos resumidos de la unión

Código Python

Algorithm ID: `native:joinbylocationsummary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Unir atributos por proximidad

Toma una capa de vector de entrada y crea una nueva capa de vector con campos adicionales en su tabla de atributos. Los atributos adicionales y sus valores se toman de una segunda capa vectorial. Las entidades se unen buscando las entidades más cercanas de cada capa.

De forma predeterminada, solo se une la entidad más cercana, pero la unión también puede unirse a las k entidades vecinas más cercanas.

Si se especifica una distancia máxima, solo se igualarán las entidades que estén más cerca que esta distancia.

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Análisis de vecino mas próximo, *Unir atributos por valor de campo*, *Unir atributos por localización*, *Matriz distancia*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	La capa entrante.
Capa de entrada 2	`INPUT_2`	[vector: geometría]	La capa unión
Campos de capa 2 a copiar (dejar en blanco para copiar todos los campos)	`FIELDS_TO_COPY`	[campos]	Unir campos de capa para copiar (si está vacío, se copiarán todos los campos).
Descartar registros que no se pueden unir	`DISCARD_NONMAT`	[booleano] Predeterminado: Falso	Eliminar de la salida los registros de la capa de entrada que no se pudieron unir
Prefijo de campo unido	`PREFIX`	[cadena de texto]	Prefijo de campo unido
Máximo de vecinos más cercanos	`NEIGHBORS`	[numérico: entero] Predeterminado: 1	Máximo número de vecinos mas cercanos
Distancia máxima	`MAX_DISTANCE`	[numérico: doble]	Distancia máxima de búsqueda

continúe en la próxima página

Tabla 24.576 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Capa unida Opcional	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial que contiene los objetos espaciales unidos. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Entidades que no se pueden unir de la primera capa	`NON_MATCHING`	[la misma que la entrada] Predeterminado: `[Omitir salida]`	Especifique la capa vectorial que contiene los objetos espaciales que no se han podido unir. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa unida	`OUTPUT`	[la misma que la entrada]	La capa de unión saliente.
Entidades que no se pueden unir de la primera capa	`NON_MATCHING`	[la misma que la entrada]	Capa que contiene las entidades de la primera capa que no se pudieron unir a ninguna entidad de la capa de unión.
Número de entidades unidas de la tabla de entrada	`JOINED_COUNT`	[numérico: entero]	Número de entidades de la tabla de entrada que se han unido.
Número de entidades que no se pueden unir de la capa de entrada	`UNJOINABLE_COU`	[numérico: entero]	Número de entidades de la tabla de entrada que no se pudieron unir.

Código Python

Algoritmo ID: `native:joinbynearest`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

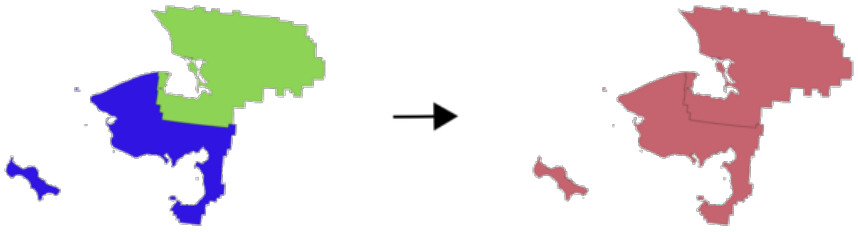
Mezclar capas vectoriales

Combina múltiples capas vectoriales del **mismo tipo geométrico** en una sola.

The attribute table of the resulting layer will contain the fields from all input layers. If fields with the same name but different types are found then the exported field will be automatically converted into a string type field. Optionally, new fields storing the original layer name and source can be added.

Si alguna capa de entrada contiene valores Z o M, la capa de salida también contendrá estos valores. Del mismo modo, si alguna de las capas de entrada es de varias partes, la capa de salida también será una capa de varias partes.

Opcionalmente, se puede configurar el sistema de referencia de coordenadas de destino (CRS) para la capa fusionada. Si no está configurado, el SRC se tomará de la primera capa de entrada. Todas las capas se volverán a proyectar para que coincidan con este SRC.



Menú predeterminado: *Vectorial ► Herramientas de Administración de Datos*

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Dividir capa vectorial

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`LAYERS`	[vectorial: cualquiera] [lista]	Las capas que se van a fusionar en una sola capa. Las capas deben ser del mismo tipo de geometría.
SRC de destino Opcional	`CRS`	[src]	Elija el SRC para la capa de salida. Si no se especifica, se utiliza el SRC de la primera capa de entrada.
Add source layer information (layer name and path)	`ADD_SOURCE_FIE`	[booleano] Predeterminado: Verdadero	Add fields storing the original layer name and path

continúe en la próxima página

Tabla 24.578 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Mezclado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Mezclado	`OUTPUT`	[la misma que la entrada]	Capa de vector de salida que contiene todas las características y atributos de las capas de entrada.

Código Python

Algoritmo ID: `native:mergevectorlayers`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Ordenar por expresión

Ordena una capa vectorial según una expresión: cambia el índice de entidad según una expresión.

Tenga cuidado, es posible que no funcione como se esperaba con algunos proveedores, es posible que el orden no se mantenga siempre.

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vectorial: cualquiera]	Capa vectorial entrante a ordenar
Expresión	`EXPRESSION`	[expresión]	Expresión a usar para la ordenación
Orden ascendente	`ASCENDING`	[booleano] Predeterminado: Verdadero	Si está marcada, la capa vectorial se ordenará de valores pequeños a grandes.
Ordenar nulos primero	`NULLS_FIRST`	[booleano] Predeterminado: Falso	Si está marcado, los valores nulos se colocan primero
Ordenado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ordenado	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente (ordenada)

Código Python

Algoritmo ID: `native:orderbyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Repara Shapefile

REpara un conjunto de datos dañado de Shapefile de ESRI mediante (re)creación del archivo SHX.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Shapefile entrante	`INPUT`	[archivo]	Ruta completa al conjunto de datos ESRI Shapefile con un archivo SHX faltante o dañado

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa reparada	`OUTPUT`	[vector: geometría]	La capa vectorial entrante con el archivo SHX reparado

Código Python

Algoritmo ID: `native:repairshapefile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Capa reproyectada

Reproyecta una capa vectorial en un SRC diferente. La capa reproyectada tendrá las mismas entidades y atributos que la capa de entrada.

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Ver también

Asignar proyección, *Define projection*, *Encontrar proyección*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vector: geometría]	Capa vectorial entrante a reproyectar
SRC destino	`TARGET_CRS`	[src] Predeterminado: `EPSG:4326 – WGS 84`	Sistema de coordenadas de referencia de destino
Convertir geometrías curvas en segmentos rectos	`CONVERT_CURVED`	[booleano] Predeterminado: Falso	Si se marca, las geometrías curvas se convertirán en segmentos rectos en el proceso, evitando posibles problemas de distorsión.
Reproyectado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Operar Coordenada Opcional	`OPERATION`	[cadena de texto]	Operación específica a usar para una tarea de reproyección particular, en lugar de forzar siempre el uso de la configuración de transformación del proyecto actual. Útil cuando se vuelve a proyectar una capa en particular y se requiere control sobre la línea de transformación exacta. Requiere versión proj >= 6. Leer mas en *Transformaciones de Datum*.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Reproyectado	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente (reproyectada)

Código Python

Algoritmo ID: `native:reprojectlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Guardar objetos vectoriales en archivo

Guarda características vectoriales en un conjunto de datos de un archivo especificado.

Para los formatos de conjuntos de datos que admiten capas, se puede utilizar un parámetro opcional de nombre de capa para especificar una cadena personalizada. Se pueden especificar opciones opcionales de conjuntos de datos y capas definidas por GDAL. Para más información al respecto, lea la documentación en línea de GDAL sobre el formato.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Objetos espaciales vectoriales	`INPUT`	[vectorial: cualquiera]	Capa vectorial de entrada.
Objetos espaciales guardados	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Save to temporary file]`	Especifique el archivo en el que se guardarán los objetos espaciales. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Nombre de la capa Opcional	`LAYER_NAME`	[cadena de texto]	Nombre a usar para la capa de salida
Opciones de conjunto de datos GDAL Opcional	`DATASOURCE_OPT`	[cadena de texto]	Opciones de creación de conjuntos de datos GDAL del formato de salida. Separe las opciones individuales con punto y coma.
Opciones de capa GDAL Opcional	`LAYER_OPTIONS`	[cadena de texto]	Opciones de creación de capas GDAL del formato de salida. Separe las opciones individuales con punto y coma.

continúe en la próxima página

Tabla 24.588 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Acción a tomar sobre archivo preexistente	`ACTION_ON_EXIS`	[enumeración] Predeterminado: 0	Cómo gestionar los objetos espaeciales existentes. Los métodos válidos son: • 0 — Crear o sobrescribir archivo • 1 — Crear o sobrescribir capa • 2 — Añadir objetos espaciales a la capa existente, pero no crear nuevos campos • 3 — Añadir objetos espaciales a la capa existente y crear nuevos campos si es necesario

Salidas

Etiqueta	Nombre	Tipo	Descripción
Objetos espaciales guardados	`OUTPUT`	[la misma que la entrada]	Capa vectorial con los objetos espaciales guardados.
Nombre y ruta de archivo	`FILE_PATH`	[cadena de texto]	Nombre y ruta de archivo de salida
Nombre de la capa	`LAYER_NAME`	[cadena de texto]	Nombre de la capa, si existe.

Código Python

Algoritmo ID: `native:savefeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Definir codificación de capa

Establece la codificación utilizada para leer los atributos de una capa. No se realizan cambios permanentes en la capa, sino que solo afecta cómo se lee la capa durante la sesión actual.

> **Nota**
>
> El cambio de codificación solo se admite para algunas fuentes de datos de la capa vectorial.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Objetos espaciales guardados	`INPUT`	[vector: geometría]	Capa vectorial para definir la codificación
Codificación	`ENCODING`	[cadena de texto]	Codificación de texto para asignar a la capa en la sesión QGIS actual.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de salida	`OUTPUT`	[la misma que la entrada]	Capa vectorial de entrada con la codificación del conjunto.

Código Python

ID Algoritmo: `native:setlayerencoding`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Dividir entidades por carácter

Las entidades se dividen en varias entidades de salida dividiendo el valor de un campo en un carácter específico. Por ejemplo, si una capa contiene entidades con varios valores separados por comas contenidos en un solo campo, este algoritmo se puede utilizar para dividir estos valores en varias entidades de salida. Las geometrías y otros atributos permanecen sin cambios en la salida. Opcionalmente, la cadena de separación puede ser una expresión regular para mayor flexibilidad.

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada
Dividir usando valores en el campo	`FIELD`	[campo de tabla: cualquiera]	Campo a usar para la división

continúe en la próxima página

Tabla 24.592 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Dividir valor usando carácter	`CHAR`	[cadena de texto]	Carácter a usar para dividir
Usar separador de expresión regular	`REGEX`	[booleano] Predeterminado: Falso	
Cortar	`OUTPUT`	[la misma que la entrada] Predeterminado: `Crear capa temporal`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cortar	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente.

Código Python

ID Algoritmo: `native:splitfeaturesbycharacter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Dividir capa vectorial

Crea un conjunto de vectores en una carpeta de salida basándose en una capa de entrada y un atributo. La carpeta de salida contendrá tantas capas como los valores únicos encontrados en el campo deseado.

El número de archivos generados es igual al número de valores diferentes encontrados para el atributo especificado.

Es la operación opuesta a *mezclar*.

Menú predeterminado: *Vectorial ► Herramientas de Administración de Datos*

Ver también

Mezclar capas vectoriales

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada
Campo de ID único	`FIELD`	[campo de tabla: cualquiera]	Campo a usar para la división
Directorio de salida	`OUTPUT`	[carpeta] Predeterminado: `[Guardar en directorio temporal]`	Especifique el directorio para las capas de salida. *Uno de*: • Guardar en Directorio Temporal • Guardar en Directorio

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tipo de campo de salida Opcional	`FILE_TYPE`	[enumeración] Predeterminado: `gpkg` en la ventana de diálogo	Seleccione la extensión de los archivos de salida. Si no se especifica o no es válida, el formato de los archivos de salida será el establecido en el ajuste de procesamiento «Extensión de la capa vectorial de salida por defecto».

Salidas

Etiqueta	Nombre	Tipo	Descripción
Directorio de salida	`OUTPUT`	[carpeta]	El directorio para las capas salientes
Capas salientes	`OUTPUT_LAYERS`	[misma que la entrada] [lista]	Las capas vectoriales salientes resultantes de la división.

Código Python

ID Algoritmo: `native:splitvectorlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Truncar tabla

Trunca una capa, eliminando todas las entidades dentro de la capa.

> **Advertencia**
>
> Este algoritmo modifica la capa en su lugar y las entidades eliminadas no se pueden restaurar.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa Entrante	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa truncada	`OUTPUT`	[carpeta]	La capa de entrada, todos los objetos espaciales eliminados

Código Python

ID Algoritmo: `native:truncatetable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.26 Geometría vectorial

Agregar atributos de geometría

Calcula las propiedades geométricas de los objetos de una capa vectorial y las incluye en la capa resultado.

Genera una nueva capa vectorial con el mismo contenido de la capa de entrada, pero con atributos adicionales que contienen las medidas geométricas basadas en el sistema de coordenadas de referencia.

Los atributos agregados a la tabla dependen del tipo de geometría y de la dimensión de la capa de entrada:

- para **punto** capas: coordenadas X (`xcoord`), Y (`ycoord`), Z (`zcoord`) y/o valor M (`mvalue`)
- para **línea** capas: `longitud` y para tipos de geometría LineString y CompoundCurve, la característica `sinuosidad` y la distancia recta (`straightdis`)
- para **polígono** capas: `perímetro` y `área`

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Menú predeterminado: *Vectorial ► Herramientas de geometría*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Usando el cálculo	`CALC_METHOD`	[enumeración] Predeterminado: 0	Cálculo de los parámetros a usar para las propiedades geométricas. Uno de: • 0 — SRC de la Capa • 1 — SRC del Proyecto • 2 — Elipsoidal
Añadida información geométrica	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (copia de entrada con geometría). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Añadida información geométrica	`OUTPUT`	[la misma que la entrada]	Copia de la capa de entrada de vector con la adición de los campos de geometría

Código Python

Algorithm ID: `native:exportaddgeometrycolumns`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Transformación afinar

Aplica una transformación afín a las geometrías de la capa. Las transformaciones afines pueden incluir traslación, escalado y rotación. Las operaciones se realizan en el siguiente orden: escala, rotación y traslación.

Los valores Z y M (si están presentes) se pueden traducir y escalar.

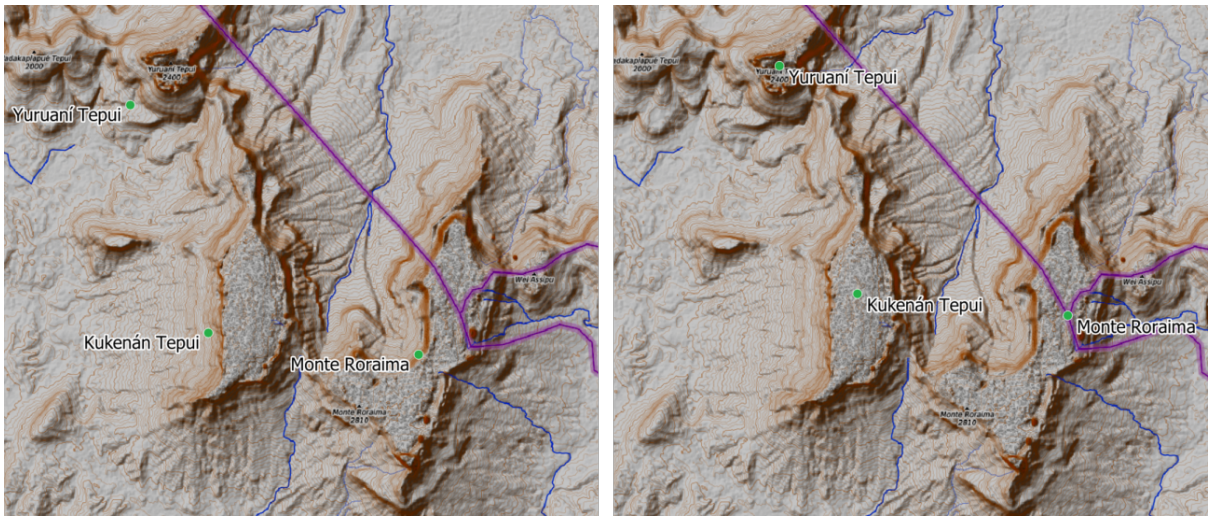


Figura 24.89: Capa de puntos vectoriales (puntos verdes) antes (izquierda) y después (derecha) de una transformación afín (traslación).

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Ver también

Traslado

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Translació (eje-x)	`DELTA_X`	[numérico: doble] Predeterminado: 0.0	Desplazamiento a aplicar en el eje X.
Translación (eje-y)	`DELTA_Y`	[numérico: doble] Predeterminado: 0.0	Desplazamiento a aplicar en el eje Y.
Translación (eje-z)	`DELTA_Z`	[numérico: doble] Predeterminado: 0.0	Desplazamiento a aplicar en el eje Z.

continúe en la próxima página

Tabla 24.601 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Translación (valores-m)	DELTA_M	[numérico: doble] Predeterminado: 0.0	Ajuste a aplicar en valores m.
Factor de escala (eje-x)	SCALE_X	[numérico: doble] Predeterminado: 1.0	Valor de escalado (expansión o contracción) a aplicar en el eje X.
Factor de escala (eje-y)	SCALE_Y	[numérico: doble] Predeterminado: 1.0	Valor de escalado (expansión o contracción) a aplicar en el eje Y.
Factor de escala (eje-z)	SCALE_Z	[numérico: doble] Predeterminado: 1.0	Valor de escalado (expansión o contracción) a aplicar en el eje Z.
Factor de escala (valores-m)	SCALE_M	[numérico: doble] Predeterminado: 1.0	Valor de escalado (expansión o contracción) a aplicar en valores m.
Rotación alrededor del eje-z (grados en sentido antihorario)	ROTATION_Z	[numérico: doble] Predeterminado: 0.0	Angulo de rotación en grados.
Transformado	OUTPUT	[la misma que la entrada] Predeterminado: [Crear capa temporal]	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (TEMPORARY_OUTPUT) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Transformado	OUTPUT	[la misma que la entrada]	Capa vectorial saliente (transformada) :

Código Python

ID Algoritmo: `native:affinetransform`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Agregar

Toma una capa de vector o table y crea una nueva capa agregando característica en base a una expresión `agrupar por`.

Características por las que la expresión `agrupar por` al devolver el mismo valor se agrupan juntas.

Es posible agrupar todas las características fuente juntas usandoun valor constante en el parámetro `agrupar por`, ejemplo: NULL.

También es posible agrupar características por múltiples campos usando la función Array, ejemplo: Array(«Field1», «Field2»).

Las geometrías (si están presentes) se combinan en una geometría multiparte para cada grupo. Los atributos de salida se calculan en función de una definición agregada dada.

Este algoritmo permite el uso de las *aggregates functions* predeterminadas de la máquina QGIS Expression.

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Ver también

Coleccionar geometrías, *Disolver*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada
Agrupar por expresión	`GROUP_BY`	[campo de tabla: cualquiera] Predeterminado: “NULL”	Elegir el campo de agrupación. Si *NULL* se agruparán todas las características.

continúe en la próxima página

Tabla 24.603 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Agregados	`AGGREGATES`	[lista]	Lista de definiciones de campo de capa de salida. Ejemplo de una definición de campo: *{"agregado": "sum", "delimitador": ",", "input": " $area", "longitud": 10, "nombre": "totarea", "precisión": 0, "tipo": 6}* Por defecto, la lista contiene todos los campos de la capa de entrada. En la GUI, usted puede editar estos campos y sus definiciones y puede también: • Pulsar el botón |nuevoAtributo| para añadir un nuevo campo. • Pulsar el botón |borrarAtributo| para borrar el campo seleccionado. • Usar |flechaArriba| y |flechaAbajo| para cambiar el orden de los campos. • Pulse |limpiar Texto| para reajustar a los predeterminados (los campos de la capa de entrada). Por cada campo del que desee recuperar información, necesita definir lo siguiente: ***Expresión de entrada*** **(`input`) [expresión]** Campo o expresión de la capa de entrada. ***Función agregada*** **(`aggregate`) [enumeración]** Función que se utiliza en la expresión de entrada para devolver el valor agregado. Se trata principalmente de la función *Agregar Funciones*, además de las funciones específicas «first_value» y «last_value» para elementos de listas ordenadas. Predeterminado: *concatenado* (para tipo de datos cadena), *suma* (para tipo de datos numérico) ***Delimitador*** **(`delimiter`) [cadena]** Cadena de texto para separar valores agregados, por ejemplo en caso de concatenación. Por defecto: , ***Nombre del campo de salida*** **(`name`) [cadena]** Nombre del campo agregado en la capa de salida. Por defecto se mantiene el nombre del campo de entrada. ***Tipo*** **(`type`) [enumeration]** Tipo de datos del campo de salida. Los tipos disponibles pueden no ser compatibles con el proveedor de la capa de salida. Uno de: **Atención** Para ciertos tipos de campo, por ejemplo las listas, un parámetro adicional `sub_type` ayuda a refinar el tipo específico de los datos. Se establece automáticamente en la interfaz gráfica de usuario, pero puede ser necesario si ejecuta el algoritmo

Tabla 24.603 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Cargar campos desde la capa	Solo GUI	[vectorial: cualquiera]	Usted puede cargar campos desde otra capa y usarlos para la agregación
Agregado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (agregada) *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Agregado	`OUTPUT`	[la misma que la entrada]	Capa de vector multigeometría con valores agregados

Código Python

Algoritmo ID: `native:aggregate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Contorno

Lo que devuelve el cierre del límite combinatorio de las geometrías de entrada (i.e. el límite topológico de la geometría)

Solo para capas de polígono y línea.

Para **geometrías de polígono** , el límite consta de todas las lineas que forman el anillo del poligono.

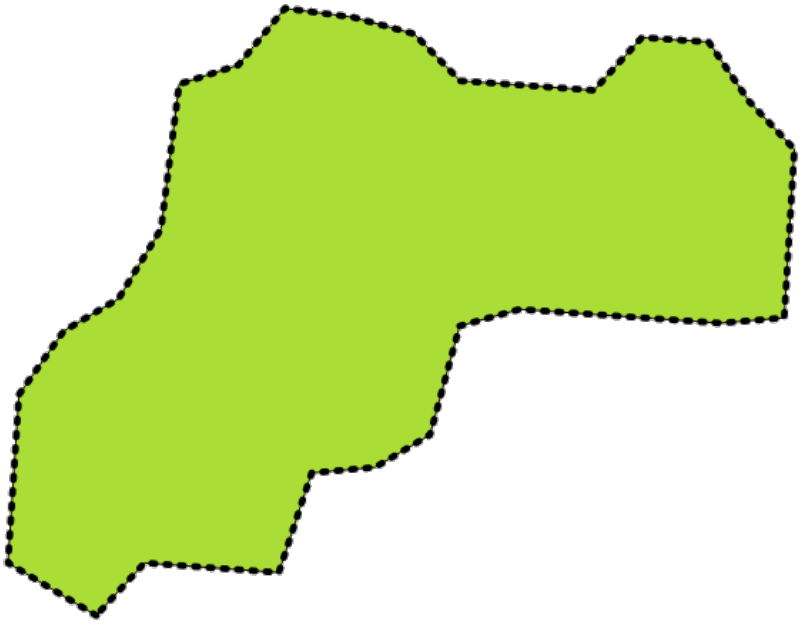


Figura 24.90: Límites (línea discontinua negra) de la capa fuente del polígono

Para **geometrías de lineas**, los límites son sus puntos finales.

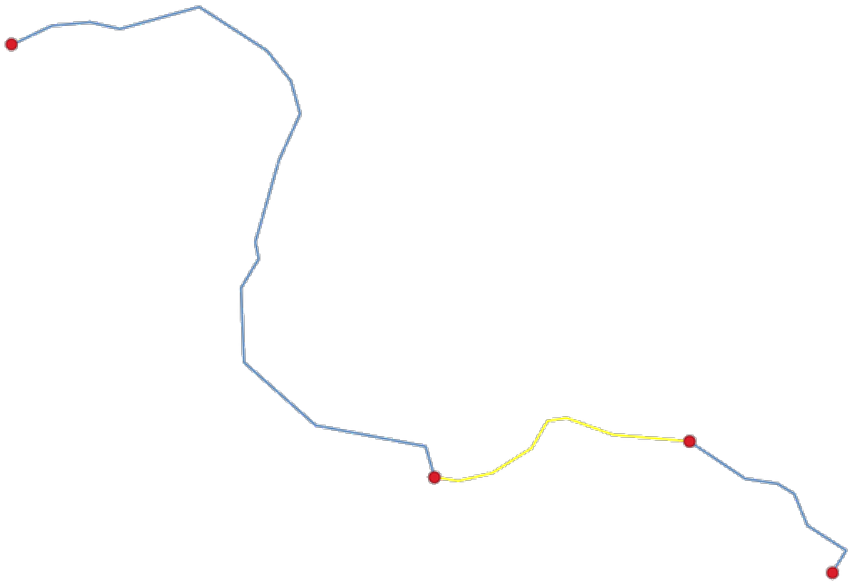


Figura 24.91: Capa límite (puntos rojos) para líneas. En amarillo una característica seleccionada.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Perímetro	`OUTPUT`	[vector: punto, línea] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (límite). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Perímetro	`OUTPUT`	[vector: punto, línea]	Limites de la capa de entrada (punto para línea y línea para polígono)

Código Python

Algoritmo ID: `native:boundary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Recuadros delimitadores

Calcula el cuadro delimitador (sobre) de cada entidad en una capa de entrada. Se soportan geometrías de línea y polígono.

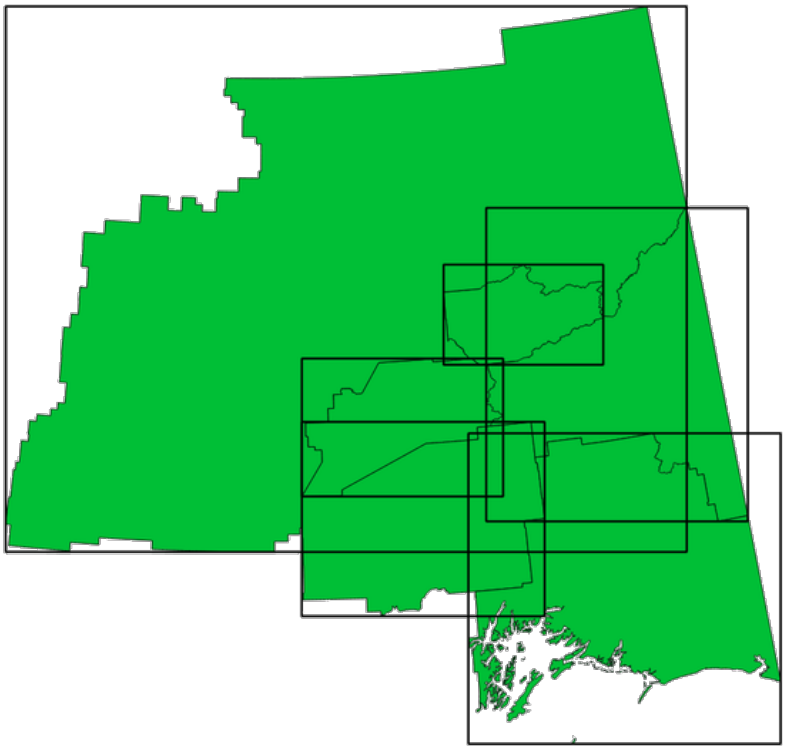


Figura 24.92: Las líneas negras representan los cuadros delimitadores de cada unidad poligonal

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Ver también

Geometría de mínima frontera

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Límites	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (cuadro delimitador). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Límites	`OUTPUT`	[vectorial: poligonal]	Cuadros delimitadores de la capa de entrada. Además de los atributos de entrada, la capa de salida también contiene los siguientes campos: `width`, `height`, `area` y `perimeter` del polígono generado.

Código Python

Algoritmo ID: `native:boundingboxes`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Buffer

Calcula un área intermedia para todos los objetos espaciales de una capa de entrada, utilizando una distancia fija o definida por los datos.

Es posible usar una distancia negativa para capas de entrada de polígono. En este caso, el búfer dará como resultado un polígono mas pequeño (retroceso).

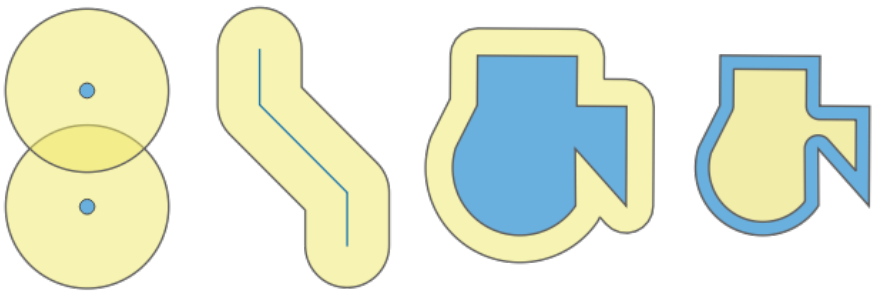


Figura 24.93: Buffer (en amarillo) de puntos, línea, polígono con buffer positivo y polígono con buffer negativo.

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Menú predeterminado: *Vectorial ► Herramientas de geoprocesamiento*

Advertencia

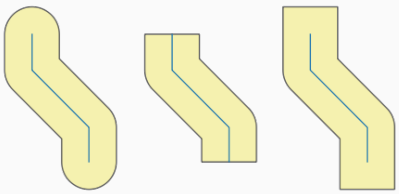
Este algoritmo puede eliminar las claves primarias o los valores FID existentes y regenerarlos en las capas de salida, dependiendo de los parámetros de entrada.

Ver también

Búfer de distancia variable, *Buffer multi-anillos (distancia constante)*, *Ancura de buffer variable (por valor de M)*

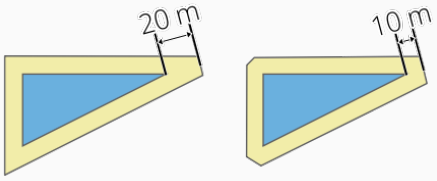
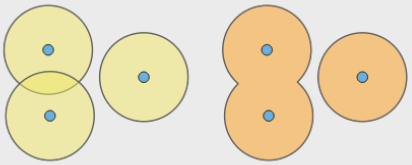
Parametros

Parámetros básicos

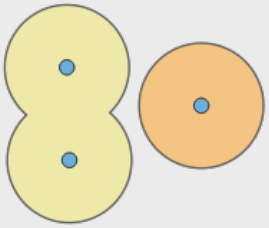
Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Distancia	`DISTANCE`	[numérico: doble] Predeterminado: 10.0	Distancia de amortiguación (desde el límite de cada elemento). Puede utilizar el botón Definido por datos situado a la derecha para seleccionar un campo a partir del cual se calculará el radio. De esta forma, puede establecer un radio diferente para cada elemento.
Segmentos	`SEGMENTS`	[numérico: entero] Predeterminado: 5	Controla el número de segmentos de linea a usar para aproximadamente un cuarto de circulo cuando se crean compensaciones redondeadas.
Estilo tapa final	`END_CAP_STYLE`	[enumeración] Predeterminado: 0	Controla como se manejan los finales de linea en el búfer. Uno de: • 0 — Redondo • 1 — Plano • 2 — Cuadrado  Figura 24.94: Estilos de tapa redondo, plano y cuadrado
Unir estilo	`JOIN_STYLE`	[enumeración] Predeterminado: 0	Especifica si se deben utilizar uniones redondas, de inglete o biseladas al compensar esquinas en una línea. Las opciones son: • 0 — Redondo • 1 — Inglete • 2 — Bisel  Figura 24.95: Estilos de unión redonda, a inglete y en bisel

continúe en la próxima página

Tabla 24.609 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Límite de inglete	`MITER_LIMIT`	[numérico: doble] Preestablecido: 2.0	Establezca la distancia máxima desde la geometría de desplazamiento que se utilizará al crear una unión a inglete como factor de la distancia de desplazamiento (sólo aplicable a los estilos de unión a inglete). Mínimo: 1.0  Figura 24.96: Un área de influencia (búfer) de 10 m con un límite de 2 y un área de influencia de 10 m con un límite de 1
Resultado de disolución	`DISSOLVE`	[booleano] Predeterminado: Falso	Disolver el área de influencia (búfer) final. Si es `Verdadero` (marcado), los búferes superpuestos se disolverán (combinarán) en una única característica multiparte.  Figura 24.97: Estándar (tres características de una sola parte - izquierda), disuelta (1 objeto espacial multiparte con 2 partes - derecha)
Buffered	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (búfer). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Mantener separados los objetos espaciales disjuntos	`SEPARATE_DISJO`	[booleano] Predeterminado: Falso	Si `True` (marcado) y disuelto está marcado, los objetos espaciales que no se solapen ni se toquen se exportarán como objetos espaciales separados (en lugar de como partes de un objeto espacial multiparte).  Figura 24.98: Resultados en 2 objetos espaciales de una sola pieza

Salidas

Etiqueta	Nombre	Tipo	Descripción
Buffered	`OUTPUT`	[vectorial: poligonal]	Salida capa poligonal (buffer)

Código Python

Algoritmo ID: `native:buffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Centroides

Crea una nueva capa de puntos, con puntos representado los centroides de las geometrías de la capa de entrada.

El centroide es un punto simple representando el baricentro (de todas las partes) de la entidad, además puede estar fuera de los límites de la entidad. Pero puede también ser un punto en cada parte de la entidad.

Los atributos de los puntos en la capa de salida son los mismos que para las entidades originales.

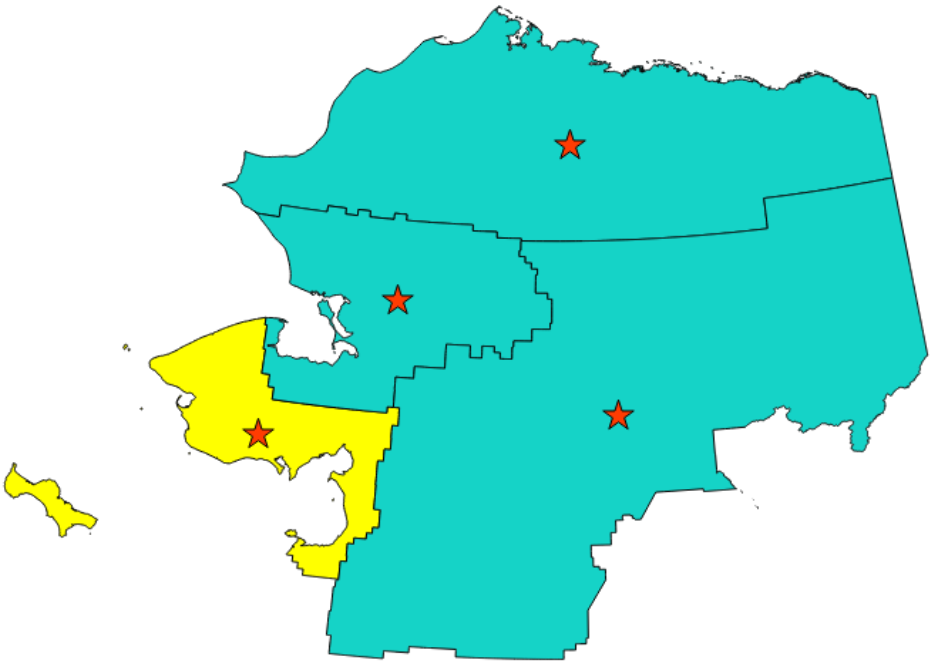


Figura 24.99: Las estrellas rojas representan los centroides de las entidades de la capa entrante.

Permite la modificación in situ de *características* de objetos espaciales punto

Menú predeterminado: *Vectorial* ► *Herramientas de geometría*

Advertencia

Este algoritmo puede eliminar las claves primarias o los valores FID existentes y regenerarlos en las capas de salida, dependiendo de los parámetros de entrada.

Ver también

Punto en la superficie

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Crear centroide para cada capa	`ALL_PARTS`	[booleano] Predeterminado: Falso	Si es verdadero (marcado), un centroide puede ser creado para cada parte de la geometría

continúe en la próxima página

Tabla 24.612 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Centroides	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (centroide). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Centroides	`OUTPUT`	[vector: punto]	Capa vectorial saliente de puntos (centroides)

Código Python

Algoritmo ID: `native:centroids`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Comprobar validez

Realiza una verificación de validez en las geometrías de una capa vectorial.

Las geometrías son clasificadas en tres grupos (válida, inválida y errónea) y para cada grupo, se genera una capa vectorial con sus geometrías:

- La **salida Válida** contiene solo las geometrías válidas (sin errores topológicos).
- La capa **salida Inválida** contiene todas las geometrías inválidas encontradas por el algoritmo.
- La capa **salida errónea** es una capa de puntos que señala donde se han encontrado geometrías inválidas.

Las tablas de atributos de las capas generadas contendrán alguna información adicional («message» para la capa **error** , «FID» y «_errors» para la capa **invalid** y solo «FID» para la capa **valid**):

La tabla de atributos de cada capa vectorial generada contendrá información adicional (número de errores encontrados y tipos de error):

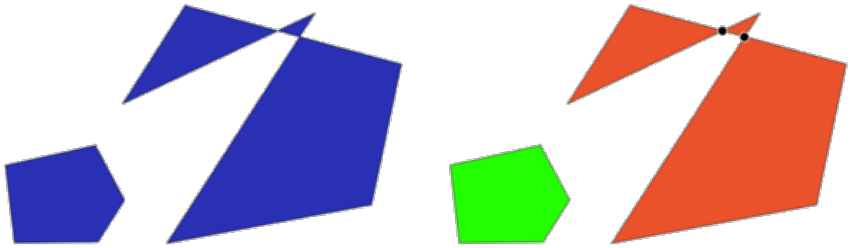


Figura 24.100: Izquierda: la capa de entrada. Derecha: la capa válida (verde), la capa no válida (naranja)

Menú predeterminado: *Vectorial* ► *Herramientas de geometría*

Ver también

Check Geometry, *Corregir geometrías*, *Validar la cobertura*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT_LAYER`	[vector: geometría]	Capa de vector de entrada
Methodo	`METHOD`	[enumeración] Preestablecido: 2	Método a usar para probar validez. Opciones: • 0: El seleccionado en la configuración de digitalización • 1: QGIS • 2: GEOS
Ignorar la autointersección del anillo	`IGNORE_RING_SE`	[booleano] Predeterminado: Falso	Ignorar los anillos auto intersectantes cuando verifique la validez.
Salida Válida	`VALID_OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial que contendrá una copia de los objetos espaciales válidos de la capa de origen. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

continúe en la próxima página

Tabla 24.614 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Salida Inválida	`INVALID_OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Capa vectorial que contiene una copia de los objetos espaciales no válidos de la capa de origen con el campo `_errors` que enumera el resumen de los errores encontrados. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Salida Error	`ERROR_OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Capa de puntos con la posición exacta de los problemas de validez detectados con el campo `mensaje` que describe los errores encontrados. :ref:`Uno de • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Recuento de errores	`ERROR_COUNT`	[numérico: entero]	El número de geometrías que causan errores.
Salida Error	`ERROR_OUTPUT`	[vector: punto]	Capa de puntos de la posición exacta de los problemas de validez detectados con el campo `message` que describe los error(es) encontrados.
Recuento de geometrías inválidas	`INVALID_COUNT`	[numérico: entero]	El número de geometrías inválidas.
Salida Inválida	`INVALID_OUTPUT`	[la misma que la entrada]	Capa vectorial que contiene una copia de las geometrías no válidas de la capa fuente con el campo `_errors` que enumera el resumen de los error(es) encontrados.
Recuento de geometrías válidas	`VALID_COUNT`	[numérico: entero]	El número de geometrías válidas.
Salida Válida	`VALID_OUTPUT`	[la misma que la entrada]	Capa vectorial conteniendo una copia de las geometrías válidas de la capa fuente.

Código Python

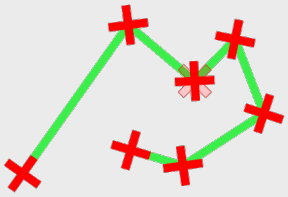
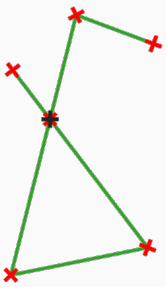
Algorithm ID: `native:checkvalidity`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

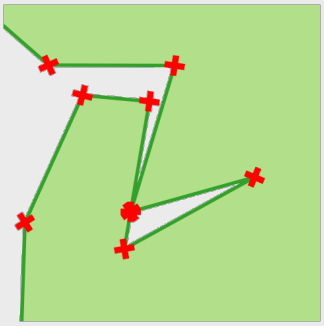
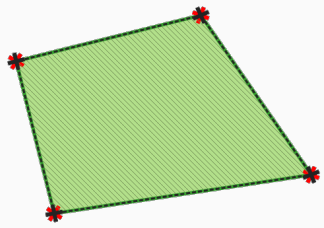
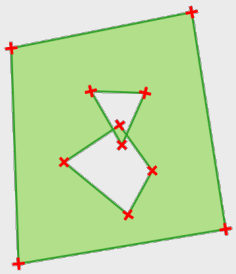
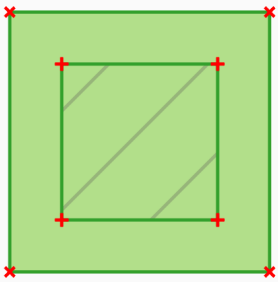
Tipos de mensajes de error y sus significados

Tabla 24.616: Algunos ejemplos de errores en la comprobación geométrica en QGIS

Contexto geométrico	Validación GEOS y mensaje de error	Validación de QGIS y mensaje de error
Los puntos consecutivos en una línea tienen las mismas coordenadas 	*Punto repetido*	*La línea a contiene x nodo(s) duplicado(s) en b*
Los segmentos de una línea se cruzan entre sí 		*Los segmentos a y b de la línea c se cruzan en d*

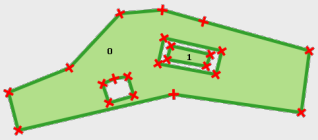
continúe en la próxima página

Tabla 24.616 – proviene de la página anterior

Contexto geométrico	Validación GEOS y mensaje de error	Validación de QGIS y mensaje de error
La geometría polígono se toca a sí misma y genera un anillo 	*Intersección del anillo consigo mismo*	*Intersección del anillo consigo mismo*
Dos anillos (exteriores o interiores) de una geometría poligonal son idénticos 	*Anillos duplicados*	
La geometría se toca a sí misma 	*Autointersección*	
Una geometría poligonal se encuentra sobre otra geometría poligonal 	*Shell anidado*	

continúe en la próxima página

Tabla 24.616 – proviene de la página anterior

Contexto geométrico	Validación GEOS y mensaje de error	Validación de QGIS y mensaje de error
Parte de una geometría multipoligonal se encuentra dentro de un agujero de una geometría multipoligonal 		✗ *El polígono a se encuentra dentro del polígono b*
La geometría del punto no tiene un par de coordenadas adecuado. El par de coordenadas no contiene un valor de latitud y un valor de longitud en ese orden.	✗ *Coordenadas no válidas*	

Coleccionar geometrías

Toma una capa vectorial y colecciona sus geometrías en nuevas geometrías multiparte.

Uno o mas atributos se pueden especificar para recoger solo geometrías pertenecientes a la misma clase (teniendo el mismo valor para los atributos especificados), alternativamente todas las geometrías pueden ser coleccionadas.

Todas las geometrías salientes serán convertidas a multi geometrías, incluso aquellas con una única parte. Este algoritmo no disuelve geometrías superpuestas - ellas serán coleccionadas juntas sin modificar la forma de cada geometría parte.

Vea los algoritmos “Promote to multipart” o “Aggregate” para opciones alternativas.

Menú predeterminado: *Vectorial* ► *Herramientas de geometría*

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Agregar, *Elevar a multiparte*, *Disolver*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Campos de ID únicos	`FIELD`	[campo de tabla: cualquiera] [lista]	Elija uno o mas atributos para coleccionar las geometrías
Coleccionadas	`OUTPUT`	[la misma que la entrada]	Capa vectorial con las geometrías recopiladas

Salidas

Etiqueta	Nombre	Tipo	Descripción
Coleccionadas	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida para las geometrías recopiladas. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Código Python

Algoritmo ID: `native:collect`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Casco cóncavo

Calcula el casco cóncavo de los objetos espaciales de una capa de puntos de entrada.

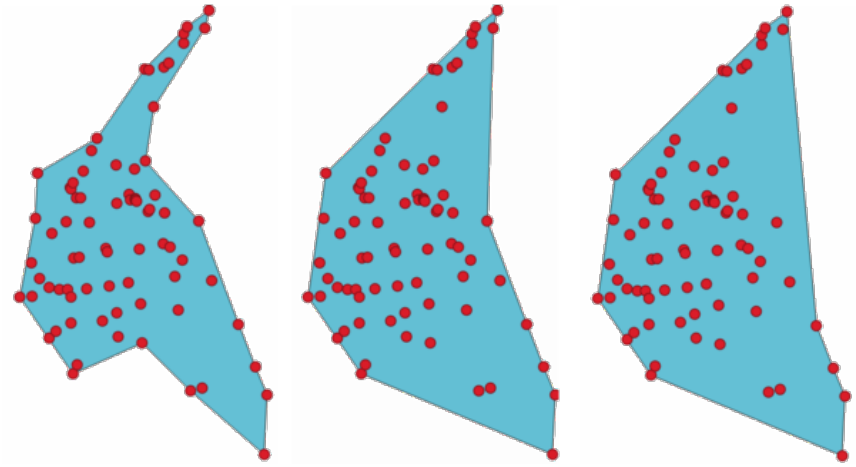


Figura 24.101: Cascos cóncavos con diferentes umbrales (0.3, 0.6, 0.9)

Ver también

Casco convexo

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada de puntos	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Umbral	`ALPHA`	[numérico: doble] Preestablecido: 0.3	Número desde 0 (casco cóncavo máximo) a 1 (casco convexo).
Permitir huecos	`HOLES`	[booleano] Predeterminado: Verdadero	Elija si desea permitir agujeros en el casco cóncavo final
Dividir geometría multiparte en geometrías de única parte	`NO_MULTIGEOMET`	[booleano] Predeterminado: Falso	Compruebe si desea tener geometrías de una sola parte en lugar de las de varias partes.
Casco Cóncavo	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Casco Cóncavo	`OUTPUT`	[vectorial: poligonal]	La capa vectorial saliente

Código Python

ID algoritmo: `native:concavehull`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Convierte el tipo de geometría

Genera una nueva capa basada en una existente, con un tipo diferente de geometría.

La tabla de atributos de la capa de salida es la misma que la de la capa de entrada.

No todas las conversiones son posibles. Por ejemplo, una capa de líneas puede convertirse en una capa de puntos, pero una capa de puntos no puede convertirse en una capa de líneas.

Ver también

:referencia:`qgispolygonize`, **:referencia:`qgislinestopolygons`**, **:referencia:`qgispolygonstolines`**, **:referencia:`qgispointstopath`**

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Nuevo tipo gemetría	`TYPE`	[enumeración] Predeterminado: 0	Tipo de geometría a aplicar para las entidades salientes. Una de: • 0 — Centroides • 1 — Nodos • 2 — Cadenas lineales • 3 — Multilíneas • 4 — Poligonal

continúe en la próxima página

Tabla 24.621 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Convertido	`OUTPUT`	[vector: geometría] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Convertido	`OUTPUT`	[vector: geometría]	Capa vectorial saliente - el tipo depende de los parámetros

Código Python

Algorithm ID: `native:convertgeometrytype`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Convertir a geometrías curvadas

Convierte una geometría en su equivalente a geometría de curva.

Las geometrías ya curvadas se mantendrán sin cambios.

 Permite la modificación in situ de las características de . de objetos espaciales líneas y polígonos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: line or polygon]	Capa de vector de entrada
Tolerancia de distancia máxima	`DISTANCE`	[numérico: doble] Preestablecido: 0.000001	La distancia máxima permitida entre la ubicación original de los vértices y donde caerían en las geometrías curvas convertidas

continúe en la próxima página

Tabla 24.623 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tolerancia de ángulo máxima	`ANGLE`	[numérico: doble] Preestablecido: 0.000001	Los segmentos se consideran adecuados para ser reemplazados por un arco si los puntos están todos regularmente espaciados en el arco candidato. Este parámetro especifica la desviación angular máxima (en grados) permitida al comprobar el espaciado regular de los puntos. Entre 0 y 45°.
Curvas	`OUTPUT`	[vector: compoundcurve or curvepolygon] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Curvas	`OUTPUT`	[vector: compoundcurve or curvepolygon]	Capa vectorial de salida con geometrías curvas

Código Python

Algoritmo ID: `native:converttocurves`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Casco convexo

Calcula el casco convexo para cada entidad en la capa de entrada.

Consulte el algoritmo “Geometría de límite mínimo” para obtener un cálculo de casco convexo que cubra toda la capa o subconjuntos agrupados de entidades.

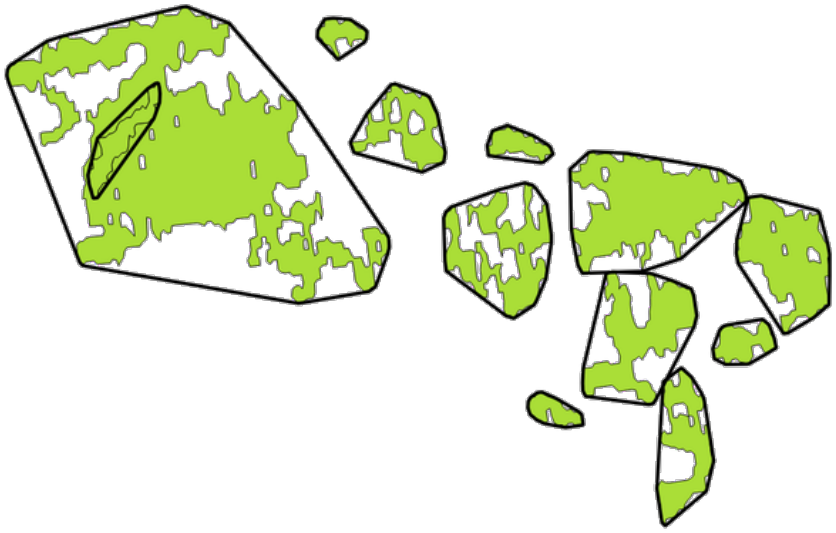


Figura 24.102: Las líneas negras identifican el casco convexo para cada entidad de capa

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Menú predeterminado: *Vectorial* ► *Herramientas de geoprocesamiento*

Ver también

Geometría de mínima frontera, *Casco cóncavo*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Casco cóncavo	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Casco cóncavo	`OUTPUT`	[vectorial: poligonal]	La capa vectorial de salida (envoltura convexa). Además de los atributos de entrada, la capa de salida también contiene los siguientes campos: `área` y `perímetro` del polígono generado.

Código Python

Algoritmo ID: `native:convexhull`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Crear buffer de cuñas

Crea buffers de formas acuñadas desde los puntos entrantes

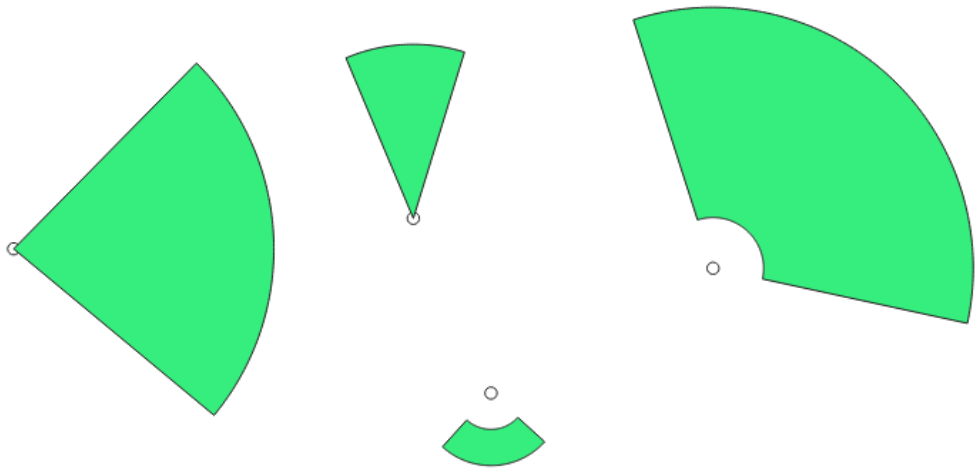


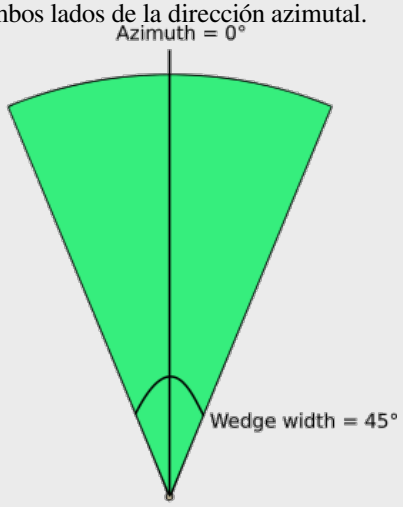
Figura 24.103: Buffers de cuñas

La salida nativa de este algoritmo son geometrías CurvePolygon, pero pueden segmentarse automáticamente a Polígonos según el formato de salida.

Ver también

Buffer, *Ancura de buffer variable (por valor de M)*, *Zonas de influencia estrechas*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Azimuth (grados al Norte)	`AZIMUTH`	[numérico: doble] Predeterminado: 0.0	Ángulo (en grados) como el valor medio de la cuña
Ancho de cuña (en grados)	`WIDTH`	[numérico: doble] Preestablecido: 45.0	Ancho (en grados) del búfer. La cuña se extenderá hasta la mitad del ancho angular a ambos lados de la dirección azimutal.  Figura 24.104: Azimut y valores de ancho del buffer de cuñas
Radio exterior	`OUTER_RADIUS`	[numérico: doble] Predeterminado: 1.0	El *tamaño* exterior (longitud) de la cuña: el tamaño se entiende desde el punto de origen hasta el borde de la forma de la cuña.
Radio Interior Opcional	`INNER_RADIUS`	[numérico: doble] Predeterminado: 0.0	Valor de radio interior. Si es 0 la cuña empezará desde el punto origen.
Buffers	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Buffers	`OUTPUT`	[vectorial: poligonal]	La capa vectorial de salida (buffer de cuña)

Código Python

Algoritmo ID: `native:wedgebuffers`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Triangulación Delaunay

Crea una capa poligonal mediante triangulación Delaunay correspondiendo a la capa entrante de puntos.

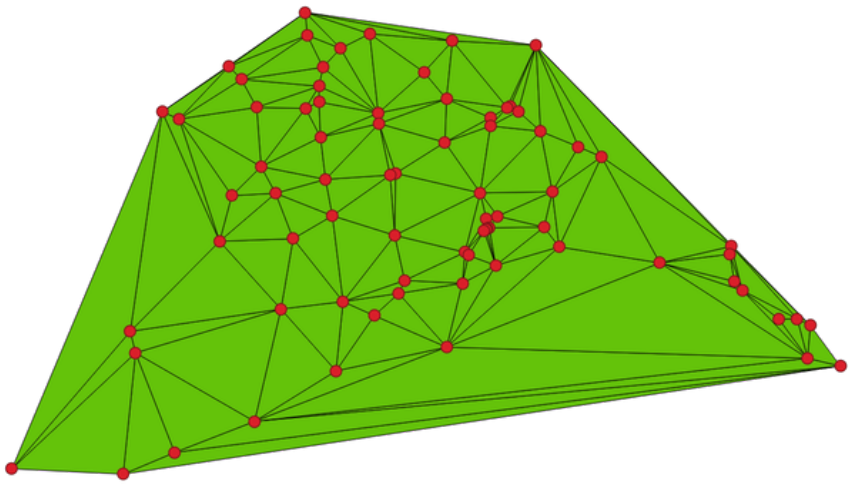


Figura 24.105: Triangulación Delaunay en puntos

Menú predeterminado: *Vectorial ► Herramientas de geometría*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Tolerancia Opcional	`TOLERANCE`	[numérico: doble] Predeterminado: 0.0	Especifica una tolerancia de ajuste opcional que puede utilizarse para mejorar la solidez de la triangulación.
Añadir ID de puntos a la salida	`ADD_ATTRIBUTES`	[booleano] Predeterminado: Verdadero	Especifica si los campos que almacenan el ID de los objetos espaciales de los puntos implicados deben añadirse a la salida. Si es falso, se utiliza un campo `id` para identificar los polígonos.
Triangulación Delaunay	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Triangulación Delaunay	`OUTPUT`	[vectorial: poligonal]	La capa vectorial saliente (triangulación Delaunay)

Código Python

ID algoritmo: `native:delaunaytriangulation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Borrar agujeros

Toma una capa de polígonos y elimina agujeros en polígonos. Crea una nueva capa vectorial en la que los polígonos con agujeros han sido reemplazados por polígonos con solo su anillo externo. Los atributos no se modifican.

Un parámetro de área mínima opcional permite eliminar solo los agujeros que son más pequeños que un umbral de área especificado. Si deja este parámetro en `` 0.0 “”, se eliminarán todos los agujeros.

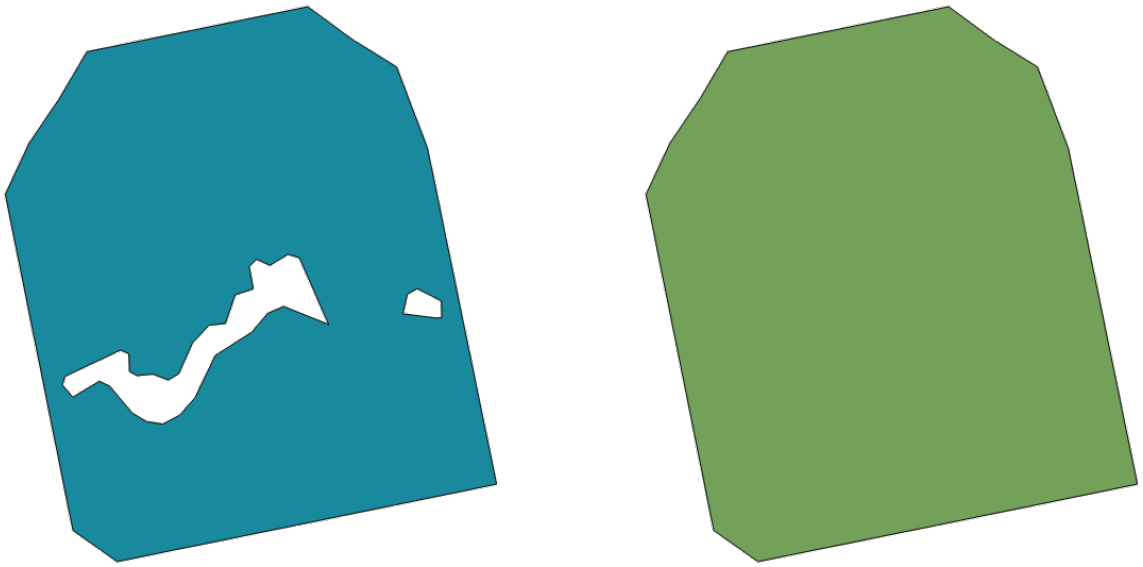


Figura 24.106: Antes y después de la limpieza

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Ver también

qgisfixgeometryhole

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa vectorial de polígonos entrante
Borra huecos con área menor que Opcional	`MIN_AREA`	[numérico: doble] Predeterminado: 0.0	Solo se eliminarán los agujeros con un área menor que este umbral. Con un valor de `0.0`, **todos** los agujeros serán eliminados.
Limpia	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Limpia	`OUTPUT`	[la misma que la entrada]	La capa (limpia) vectorial saliente

Código Python

Algoritmo ID: `native:deleteholes`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Densificar por conteo

Toma una capa lieal o poligonal y genera una nueva en la cual las geometrías tienen un número de vértices mayor que la original.

Si las geometrías tienen presentes valores Z o M estos serán interpolados linealmente a los vértices añadidos.

El número de nuevos vértices a añadir a cada segmento es especificado como un parámetro de entrada.

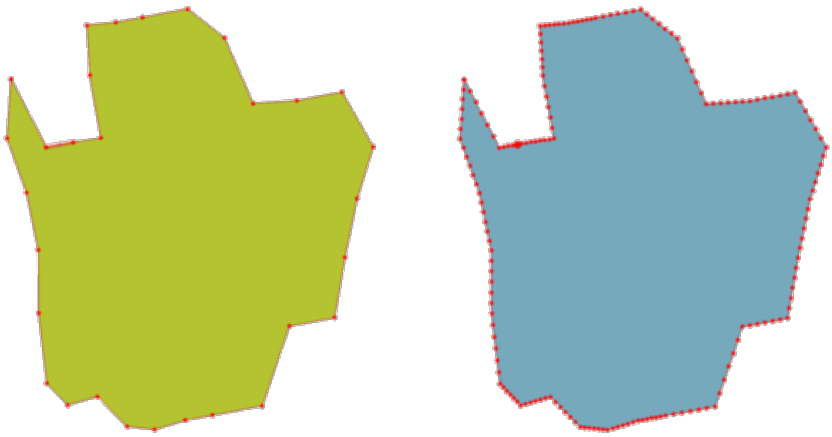


Figura 24.107: Los puntos rojos muestran los vértices antes y despues de densificar

Permite la modificación in situ de las características de . de objetos espaciales líneas y polígonos

Menú predeterminado: *Vectorial ► Herramientas de geometría*

Ver también

Densificar por intervalo

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Vertices a añadir	`VERTICES`	[numérico: entero] Predeterminado: 1	Número de vértices a añadir a cada segmento
Densificado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Densificado	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (densificada)

Código Python

Algoritmo ID: `native:densifygeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Densificar por intervalo

Toma una capa lieal o poligonal y genera una nueva en la cual las geometrías tienen un número de vértices mayor que la original.

Las geometrías se densifican añadiendo vértices adicionales regularmente colocados dentro de cada segmento, de modo que la distancia máxima entre dos vértices cualesquiera no supere la distancia especificada.

Si las geometrías tienen presentes valores Z o M estos serán interpolados linealmente a los vértices añadidos.

Ejemplo

Especificando una distancia de 3 causaría que el segmento `[0 0] -> [10 0]` sea convertido a `[0 0] -> [2.5 0] -> [5 0] -> [7.5 0] -> [10 0]`, desde 3 vértices extra son requeridos en el segmento y espaciandolos a incrementos 2.5 permite que sean convenientemente espaciados a lo largo del segmento.

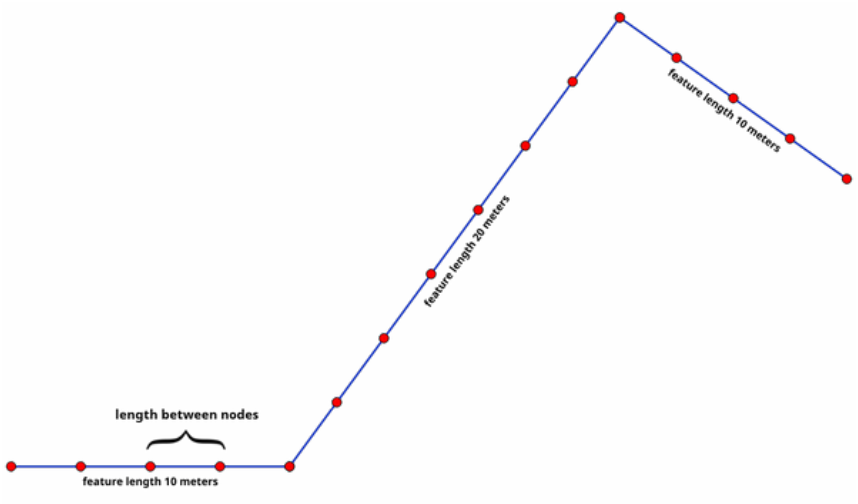


Figura 24.108: Densificar geometría a un intervalo dado

Permite la modificación in situ de las características de . de objetos espaciales líneas y polígonos

Ver también

Densificar por conteo

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Intervalo entre vértices a añadir	`INTERVAL`	[numérico: doble] Predeterminado: 1.0	Distancia máxima entre dos vértices consecutivos
Densificado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Densificado	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (densificada)

Código Python

Algoritmo ID: `native:densifygeometriesgivenaninterval`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Disolver

Toma una capa vectorial y combina sus características en nuevas entidades. Se pueden especificar uno o más atributos para disolver entidades que pertenecen a la misma clase (que tienen el mismo valor para los atributos especificados), alternativamente, todas las entidades se pueden disolver en una sola entidad.

Todas las geometrías de salida se convertirán en geometrías múltiples. En caso de que la entrada sea una capa poligonal, se borrarán los límites comunes de los polígonos adyacentes que se disuelvan. Si se activa, el ajuste opcional «Mantener características disjuntas separadas» hará que las características y partes que no se solapen ni se toquen se exporten como objetos espaciales separados (en lugar de como partes de un único objeto espacial multiparte).

La tabla de atributos resultante tendrá los mismos campos que la capa de entrada. Los valores en los campos de la capa de salida son los de la primera entidad de entrada que se procesa.

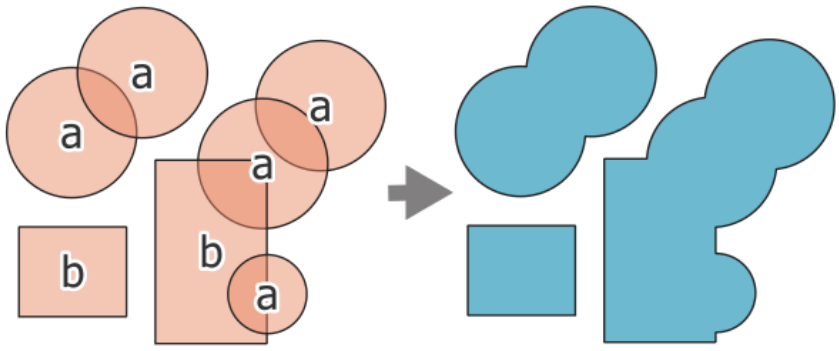


Figura 24.109: Disolución de una capa en un único elemento multiparte

Menú predeterminado: *Vectorial ► Herramientas de geoprocesamiento*

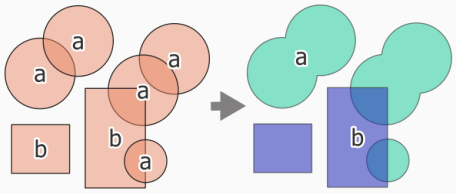
> **Advertencia**
>
> Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

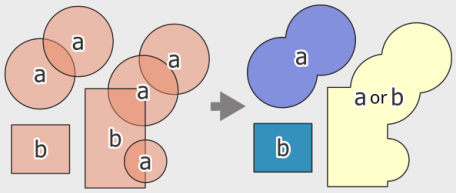
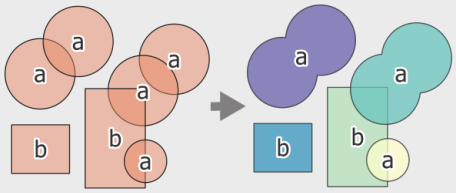
Disolver la cobertura, *Agregar*, *Coleccionar geometrías*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Disolver campo(s) Opcional	`FIELD`	[campo de tabla: cualquiera] [lista] Predeterminado: []	Las entidades que tengan el mismo valor para el campo(s) seleccionado se reemplazarán por una sola y sus geometrías se fusionarán. Si no se proporciona ningún campo, todas las características se disuelven, lo que da como resultado una característica única (multiparte).  Figura 24.110: Disolver la capa de polígonos en un atributo común (2 objetos espaciales multiparte)
Disuelto	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Mantener separados los objetos espaciales disjuntos	`SEPARATE_DISJO`	[booleano] Predeterminado: Falso	Las partes de elementos disueltos se exportan como elementos separados (en lugar de como partes de un elemento multiparte).  Figura 24.111: fuente (izquierda), disolver todo (3 objetos espaciales distintos - derecha)  Figura 24.112: fuente (izquierda), disolución en el campo (5 objetos espaciales distintos - derecha)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Disuelto	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente con geometrías disueltas

Código Python

Algoritmo ID: `native:dissolve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Drapeado (establecer el valor Z del ráster)

Utiliza valores muestreados de una banda dentro de una capa ráster para establecer el valor Z para cada vértice superpuesto en la geometría de la entidad. Los valores ráster pueden escalarse opcionalmente en una cantidad preestablecida.

Si ya existen valores Z en la capa, se sobrescribirán con el nuevo valor. Si no existen valores Z, la geometría se actualizará para incluir la dimensión Z.

Permite *modificación in situ* de objetos espaciales punto, línea, y polígono con Z habilitada

Ver también

Establecer el valor de M desde ráster, *Establecer valor Z*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Capa ráster	`RASTER`	[ráster]	Capa ráster con valores Z
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	La banda de la capa ráster de donde tomar valores Z
Valor para NoData o vértices no intersecantes	`NODATA`	[numérico: doble] Predeterminado: 0.0	Valor a usar en caso de que el vértice no intersecte (un pixel valido de) el ráster
Factor de escala	`SCALE`	[numérico: doble] Predeterminado: 1.0	Valor de escalado: los valores de banda se multiplican por este valor.
Compensado	`OFFSET`	[numérico: doble] Predeterminado: 0.0	Valor de desplazamiento: se añade algebraicamente a los valores de la banda tras aplicar el «Factor de escala».
Actualizado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida (con valores Z de la capa ráster). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Actualizado	`OUTPUT`	[la misma que la entrada]	La capa vetorial saliente con valores Z desde la capa ráster

Código Python

Algoritmo ID: `native:setzfromraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Soltar valores M/Z

Borra los valores M (medida) o Z (altitud) de geometrías entrantes.

Ver también

Establecer valor M, *Establecer valor Z*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa vectorial entrante con valores M o Z
Descartar valores M	`DROP_M_VALUES`	[booleano] Predeterminado: Falso	Borra los valores M de las geometrías
Descartar valores Z	`DROP_Z_VALUES`	[booleano] Predeterminado: Falso	Borra los valores Z de las geometrías
Z/M soltados	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Z/M soltados	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (idéntica a la capa entrante, excepto que las dimensiones M y/o Z han sido borrados de las geometrías):

Código Python

Algoritmo ID: `native:dropmzvalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Eliminar polígonos seleccionados

Combina los polígonos seleccionados de la capa de entrada con ciertos polígonos adyacentes borrando su límite común. El polígono adyacente puede ser el que tiene el área más grande o la más pequeña o el que comparte el límite común más grande con el polígono a eliminar.

Eliminar normalmente se usa para eliminar los polígonos de astilla, es decir diminutos polígonos que son el resultado de procesos de intersección de polígonos donde los límites de las entradas son similares pero no idénticos.

Menú predeterminado: *Vectorial ► Herramientas de geoprocesamiento*

Ver también

Corregir geometrías

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa vectorial de polígonos entrante
Combinar selección con el polígono vecino	`MODE`	[enumeración] Predeterminado: No establecido	Elija el parámetro a usar para deshacerse de los polígonos seleccionados: • 0 — Mayor Área • 1 — Menor Área • 2 — Frontera común mas larga

continúe en la próxima página

Tabla 24.644 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Eliminado	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Eliminado	`OUTPUT`	[vectorial: poligonal]	La capa vectorial poligonal saliente

Código Python

Algoritmo ID: `qgis:eliminateselectedpolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Explotar lineas

Toma una capa de líneas y crea una nueva en la que cada capa de línea se reemplaza por un conjunto de líneas que representan los segmentos en la línea original.

Cada línea en la capa resultante contiene solo un punto inicial y final, sin vértices intermedios entre ellos.

Si la capa de entrada está formada por CircularStrings o CompoundCurves, la capa de salida será del mismo tipo y solo contendrá segmentos de curva únicos.

Nota

- Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.
- Este algoritmo no requiere geometrías válidas como entrada.

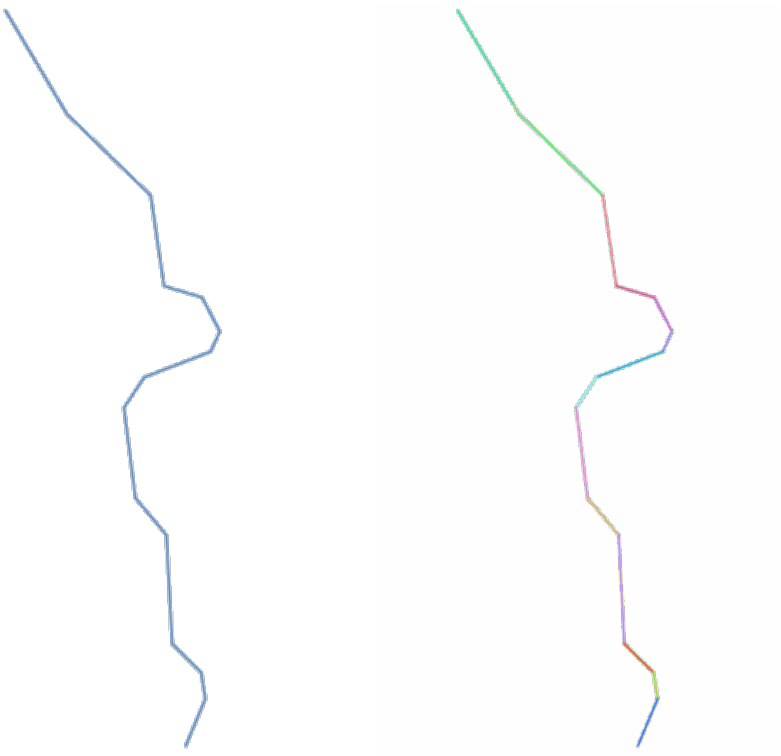


Figura 24.113: La capa de lineas original y la explotada

Permite *modificación in situ* de objetos espaciales línea

Ver también

Subdividir, *Subcadena de línea*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Explotada	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Explotada	`OUTPUT`	[vectorial: lineal]	La capa vectorial lineal saliente con entidades representado cada segmento de la capa entrante.

Código Python

Algoritmo ID: `native:explodelines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extender líneas

Extiende la geometría en un determinado valor al principio o final de la línea.

Las líneas se extienden utilizando el rumbo del primer y último segmento de la línea.

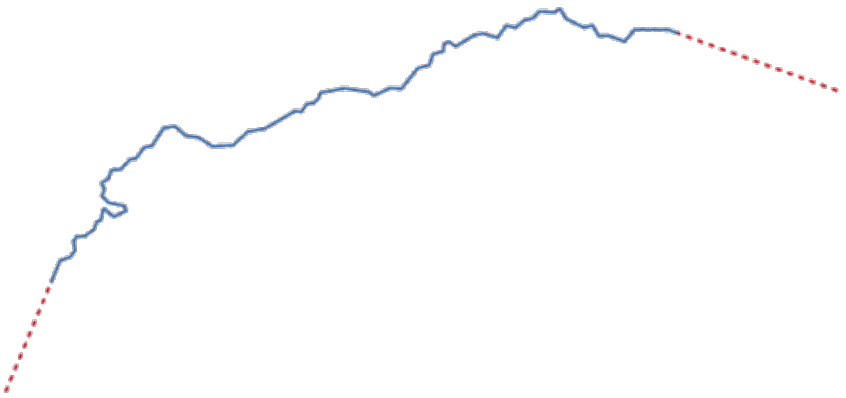


Figura 24.114: Los guiones rojos representan la extensión inicial y final de la capa original

Permite *modificación in situ* de objetos espaciales línea

Ver también

Subcadena de línea

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Distancia inicial	`START_DISTANCE`	[numérico: doble]  Predeterminado: 0.0	Distancia a la que extender el primer segmento de la línea (punto inicial)
Distancia final	`END_DISTANCE`	[numérico: doble]  Predeterminado: 0.0	Distancia a la que extender el último segmento de la línea (punto final)
Extendido	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extendido	`OUTPUT`	[vectorial: lineal]	La línea saliente (extendida) de la capa vectorial.

Código Python

Algoritmo ID: `native:extendlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extraer valores M

Extrae los valores M de las geometrías en atributos de entidades.

De manera predeterminada, solo se extrae el valor M del primer vértice de cada entidad, sin embargo, el algoritmo puede calcular opcionalmente estadísticas sobre todos los valores M de la geometría, incluyendo suma, media, mínimo y máximo.

Ver también

Extraer los valores Z, *Establecer valor M*, *Soltar valores M/Z*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Summarios a calcular	`SUMMARIES`	[enumeración] Predeterminado: [0]	Estadísticas en los valores M de una geometría. una o mas de: • 0 — Primera • 1 — Última • 2 — Recuento • 3 — Suma • 4 — Promedio • 5 — Mediana • 6 — Desviación estándar (pop) • 7 — Mínimo • 8 — Máximo • 9 — Rango • 10 — Minoría • 11 — Mayoría • 12 — Variación • 13 — Q1 • 14 — Q3 • 15 — IQR
Prefijo de columna saliente	`COLUMN_PREFIX`	[cadena de texto] Preestablecido: “m_”	El prefijo para la columna saliente (M)
Extraido	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifica la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extraido	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (con valores M)

Código Python

Algoritmo ID: `native:extractmvalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extráe vértices específicos

Toma una capa vectorial y genera una capa de puntos con puntos representando vértices específicos en las geometrías entrantes.

Por ejemplo, este algoritmo se puede usar para extraer el primer o el último vértice de la geometría. Los atributos asociados a cada punto son los mismos asociados a la entidad a la que pertenece el vértice.

El parámetro de índices de vértice acepta una cadena separada por comas que especifica los índices de los vértices a extraer. El primer vértice corresponde a un índice de 0, el segundo vértice tiene un índice de 1, etc. Los índices negativos se pueden usar para encontrar vértices al final de la geometría, por ejemplo, un índice de -1 corresponde al último vértice, -2 corresponde al segundo último vértice, etc.

Se agregan campos adicionales a los vértices que indican la posición específica del vértice (p. Ej., 0, -1, etc.), el índice de vértice original, la parte del vértice y su índice dentro de la parte (así como su anillo para polígonos), la distancia a lo largo de geometría original y ángulo de vértice bisector para la geometría original.

Permite la modificación in situ de *características* de objetos espaciales punto

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Extraer vértices, *Filtrar vértices por valor de M*, *Filtrar vértices por valor de Z*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Índices de vértices	`VERTICES`	[cadena de texto] Preestablecido: “0”	Cadena separad por comas de los índices de los vértices aextraer.
Vértices	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Vértices	`OUTPUT`	[vector: punto]	La capa vectorial de salida (de tipo punto) contiene los vértices especificados de las geometrías de la capa de entrada. Además de los atributos de entrada, la capa de salida también contiene los siguientes campos: • `vertex_pos`: el índice del vértice en la geometría, según lo solicitado por el usuario. • `vertex_index`: índice del vértice dentro de la geometría. El primer vértice tiene el índice 0. • `vertex_part`: número de la parte de la geometría a la que pertenece el vértice. En el caso de geometrías de una sola parte, este valor siempre será 0. En el caso de geometrías de varias partes (por ejemplo, MultiPolygon o MultiLine), este valor indica a qué parte pertenece el vértice. • `vertex_part_ring`: en el caso de las geometrías poligonales, este parámetro indica si el vértice pertenece al anillo exterior (0) o a uno de los anillos interiores (1, 2, …). • `vertex_part_index`: índice del vértice dentro de su respectivo anillo (exterior o interior). • `distancia`: longitud total de los segmentos de línea desde el primer vértice de la geometría. • `ángulo`: el ángulo (en grados) entre el vértice anterior y el actual.

Código Python

Algoritmo ID: `native:extractspecificvertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extraer vértices

Toma una capa vectorial y genera una capa de puntos con puntos representando los vértices en las geometrías de entrada.

Los atributos asociados a cada punto somo los mismos asociados a la entidad a la que pertenece el vértice.

Se agregan campos adicionales a los vértices que indican el índice de vértice (comenzando en 0), la parte de la entidad y su índice dentro de la parte (así como su anillo para los polígonos), la distancia a lo largo de la geometría original y el ángulo de vértice bisector de la geometría original.

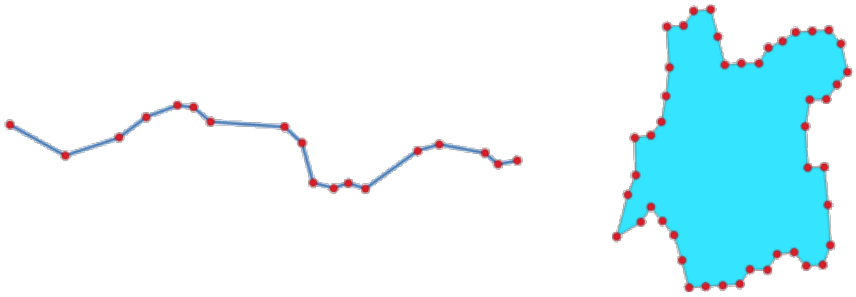


Figura 24.115: Vértices extraidos para la capa lineal y poligonal

Permite la modificación in situ de *características* de objetos espaciales punto

Menú predeterminado: *Vectorial ► Herramientas de geometría*

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Extráe vértices específicos, *Filtrar vértices por valor de M*, *Filtrar vértices por valor de Z*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Vértices	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Vértices	`OUTPUT`	[vector: punto]	La capa vectorial de salida (de tipo punto) contiene los vértices de las geometrías de la capa de entrada. Además de los atributos de entrada, la capa de salida también contiene los siguientes campos: • `vertex_index`: índice del vértice dentro de la geometría. El primer vértice tiene el índice 0. • `vertex_part`: número de la parte de la geometría a la que pertenece el vértice. En el caso de geometrías de una sola parte, este valor siempre será 0. En el caso de geometrías de varias partes (por ejemplo, MultiPolygon o MultiLine), este valor indica a qué parte pertenece el vértice. • `vertex_part_ring`: en el caso de las geometrías poligonales, este parámetro indica si el vértice pertenece al anillo exterior (0) o a uno de los anillos interiores (1, 2, …). • `vertex_part_index`: índice del vértice dentro de su respectivo anillo (exterior o interior). • `distancia`: la distancia desde el vértice anterior. • `ángulo`: devuelve el ángulo aproximado en un vértice. Normalmente se trata del ángulo promedio entre los segmentos adyacentes, y se puede visualizar como la orientación de una línea que sigue la curvatura de la geometría en el vértice especificado.

Código Python

Algoritmo ID: `native:extractvertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extraer los valores Z

Extrae los valores Z de las geometrías en los atributos de la entidad.

Por defecto, solo se extrae el valor Z del primer vértice de cada entidad, sin embargo, el algoritmo puede calcular opcionalmente estadísticas sobre todos los valores Z de la geometría, incluyendo suma, media, mínimo y máximo.

Ver también

Extraer valores M, *Establecer valor Z*, *Soltar valores M/Z*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Summarios a calcular	`SUMMARIES`	[enumeración] Predeterminado: [0]	Estadísticas en los valores Z de una geometría. Uno o mas de: • 0 — Primera • 1 — Última • 2 — Recuento • 3 — Suma • 4 — Promedio • 5 — Mediana • 6 — Desviación estándar (pop) • 7 — Mínimo • 8 — Máximo • 9 — Rango • 10 — Minoría • 11 — Mayoría • 12 — Variación • 13 — Q1 • 14 — Q3 • 15 — IQR
Prefijo de columna saliente	`COLUMN_PREFIX`	[cadena de texto] Preestablecido: “z_”	El prefijo para la columna de salida (Z)
Extraido	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifica la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extraido	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (con valores Z)

Código Python

Algoritmo ID: `native:extractzvalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Filtrar vértices por valor de M

Filtra los vértices en función de su valor M, devolviendo geometrías con solo puntos de vértice que tienen un valor M mayor o igual que el valor mínimo especificado y / o menor o igual que el valor máximo.

Si no se especifica el valor mínimo, solo se prueba el valor máximo y, de manera similar, si no se especifica el valor máximo, solo se prueba el valor mínimo.

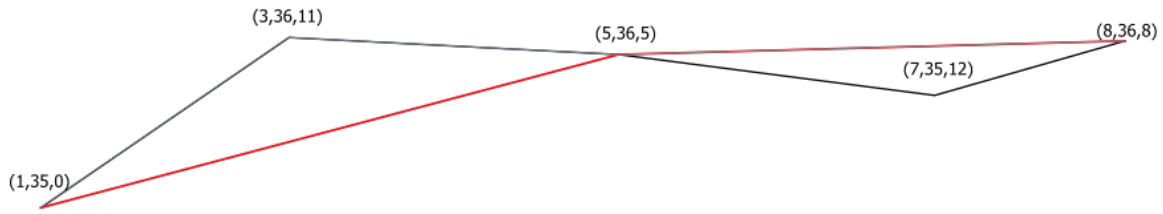


Figura 24.116: La línea roja representa la línea negra con solo vértices cuyo valor M es <= 10.

Permite *modificar los objetos espaciales in situ.* de objetos espaciales de puntos, líneas y polígonos con M habilitado

Advertencia

Dependiendo de los atributos geométricos de entrada y los filtros utilizados, las geometrías creadas por este algoritmo pueden dejar de ser válidas.

Ver también

Filtrar vértices por valor de Z, *Extraer vértices*, *Extráe vértices específicos*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capas de vectores de entrada para eliminar vértices de
Mínimo Opcional	`MIN`	[numérico: doble] Predeterminado: No establecido	Mínimo permitido de los valores M
Máximo Opcional	`MAX`	[numérico: doble] Predeterminado: No establecido	Máximo permitido de valores M
Filtrado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Filtrado	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente de entidades con sólo los vértices filtrados.

Código Python

Algoritmo ID: `native:filterverticesbym`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Filtrar vértices por valor de Z

Filtra los vértices en función de su valor Z, devolviendo geometrías con solo puntos de vértice que tienen un valor Z mayor o igual que el valor mínimo especificado y / o menor o igual que el valor máximo.

Si no se especifica el valor mínimo, solo se prueba el valor máximo y, de manera similar, si no se especifica el valor máximo, solo se prueba el valor mínimo.

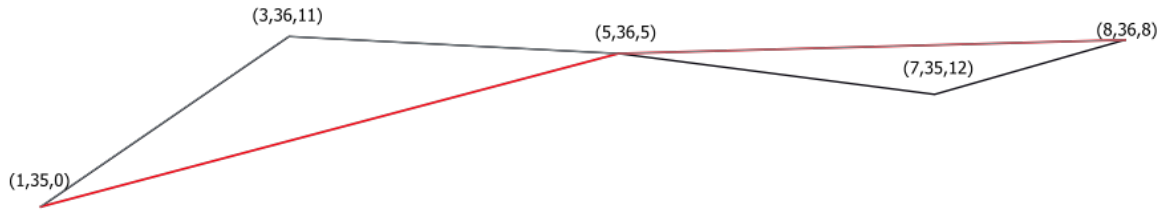


Figura 24.117: La línea roja representa la línea negra con solo vértices cuyo valor Z es <= 10.

Permite *modificar los objetos espaciales in situ.* de objetos espaciales de punto, línea y polígono con Z habilitado

Advertencia

Dependiendo de los atributos geométricos de entrada y los filtros utilizados, las geometrías creadas por este algoritmo pueden dejar de ser válidas. Es posible que tenga que ejecutar el algoritmo *Corregir geometrías* para garantizar su validez.

Ver también

Filtrar vértices por valor de M, *Extraer vértices*, *Extráe vértices específicos*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capas de vectores de entrada para eliminar vértices de
Mínimo Opcional	`MIN`	[numérico: doble] Predeterminado: No establecido	Mínimo permitido de valores Z
Máximo Opcional	`MAX`	[numérico: doble] Predeterminado: No establecido	Máximo permitido de valores Z

continúe en la próxima página

Tabla 24.660 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Filtrado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Filtrado	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente de entidades con sólo los vértices filtrados.

Código Python

Algoritmo ID: `native:filterverticesbyz`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Corregir geometrías

Intenta crear una representación válida de una geometría no válida dada sin perder ninguno de los vértices de entrada. Ya se devuelven geometrías válidas sin más intervención. Siempre genera una capa de geometría múltiple.

Permite *modificación in situ* de objetos espaciales punto, línea, y polígono con M habilitada

Advertencia

Los valores M se eliminarán de la salida.

Ver también

Comprobar validez, *Fix Geometry*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Método de reparación	`METHOD`	[enumeración] Predeterminado: 1	Método utilizado para reparar las geometrías. Uno de: • 0 — `Líneas de trabajo`: combina todos los anillos en un conjunto de líneas nodadas y luego extrae polígonos válidos de ese linework • 1 — `Structure`: first makes all rings valid and then merges shells and subtracts holes from shells to generate valid result. Assumes that holes and shells are correctly categorized. **Atención** Using this method requires QGIS installed with GEOS >= 3.10 (see *Help* ► *About* menu).
Geometrías corregidas	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Geometrías corregidas	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente con las geometrías corregidas.

Código Python

Algoritmo ID: `native:fixgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Fuerza la regla de la mano derecha

Obliga a las geometrías de polígonos a respetar la regla de la mano derecha, en la que el área delimitada por un polígono está a la derecha del límite. En particular, el anillo exterior está orientado en sentido horario y cualquier anillo interior en sentido antihorario.

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa de vector de entrada
Reorientado	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Reorientado	`OUTPUT`	[vectorial: poligonal]	La capa vectorial saliente con geometrías reorientadas.

Código Python

Algoritmo ID: `native:forcerhr`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Línea geodésica dividida en antimeridiano

Divide una línea en múltiples segmentos geodésicos, siempre que la línea cruza el antimeridiano (±180 grados de longitud).

La división en el antimeridiano ayuda a la representación visual de las líneas en algunas proyecciones. La geometría devuelta siempre será una geometría de varias partes.

Siempre que los segmentos de línea en la geometría de entrada crucen el antimeridiano, se dividirán en dos segmentos, y la latitud del punto de ruptura se determinará utilizando una línea geodésica que conecta los puntos a cada lado de este segmento. La configuración actual del elipsoide del proyecto se utilizará al calcular este punto de interrupción.

Si la geometría de entrada contiene valores M o Z, estos se interpolarán linealmente para los nuevos vértices creados en el antimeridiano.

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Permite *modificación in situ* de objetos espaciales línea

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Cortar	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de la línea de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cortar	`OUTPUT`	[vectorial: lineal]	La capa vectorial lineal saliente cortada en el antimeridiano.

Código Python

Algoritmo ID: `native:antimeridiansplit`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Geometría por expresión

Actualiza las geometrías existentes (o crea nuevas geometrías) para las características de entrada mediante el uso de una expresión QGIS.

Esto permite modificaciones de geometría complejas que pueden utilizar toda la flexibilidad del motor de expresión QGIS para manipular y crear geometrías para las características de salida.

Para obtener ayuda con las funciones de expresión QGIS, consulte la ayuda incorporada disponible en *generador de expresiones*.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada
Tipo de geometría saliente	`OUTPUT_GEOMETR`	[enumeración] Predeterminado: 0	La geometría de salida depende en gran medida de la expresión: por ejemplo, si crea un búfer, el tipo de geometría debe ser polígono. Uno de: • 0 — Poligonal • 1 — Lineal • 2 — Puntos
Geometría saliente tiene valores Z	`WITH_Z`	[booleano] Predeterminado: Falso	Elija si la geometría de salida debe incluir la dimensión Z
Geometría saliente tiene valores M	`WITH_M`	[booleano] Predeterminado: Falso	Elija si la geometría de salida debe incluir la dimensión M
Geometría de expresión	`EXPRESSION`	[expresión] Preestablecido: “$geometry”	Agregue la expresión de geometría que desea usar. Puede usar el botón para abrir el cuadro de diálogo Expresión. El cuadro de diálogo enumera todas las expresiones relevantes, junto con su ayuda y guía.

continúe en la próxima página

Tabla 24.668 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Geometría modificada	`OUTPUT`	[vector: geometría] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Geometría modificada	`OUTPUT`	[vector: geometría]	La capa vectorial saliente

Código Python

Algoritmo ID: `native:geometrybyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Interpolar puntos en línea

Crea una geometría de punto interpolada a una distancia establecida a lo largo de geometrías de línea o curva.

Los valores Z y M se interpolan linealmente de los valores existentes.

Si se encuentra una geometría multiparte, solo se considera la primera parte al calcular la subcadena.

Si la distancia especificada es mayor que la longitud de la entidad de entrada, la entidad resultante tendrá una geometría nula.

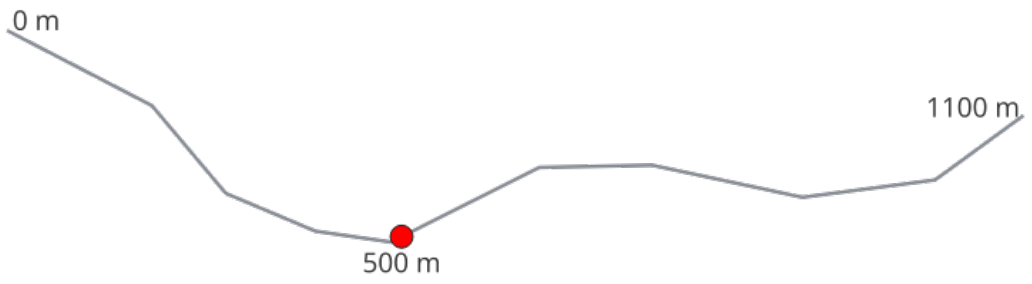


Figura 24.118: Punto interpolado a 500 m del comienzo de la línea.

Ver también

Puntos a lo largo de la geometrías

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Distancia	`DISTANCE`	[numérico: doble]  Predeterminado: 0.0	Distancia desde el comienzo de la línea
Puntos interpolados	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos interpolados	`OUTPUT`	[vector: punto]	La capa vectorial saliente de punto con entidades a una distancia establecida a lo largo de la línea o el límite del polígono

Código Python

Algoritmo ID: `native:interpolatepoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Mantener N partes mayores

Toma una capa con polígonos o multipolígonos y devuelve una nueva capa en la que solo se mantienen los polígonos más grandes * n * de cada entidad multipolígono. Si una entidad tiene * n * o menos partes, la entidad solo se copiará.

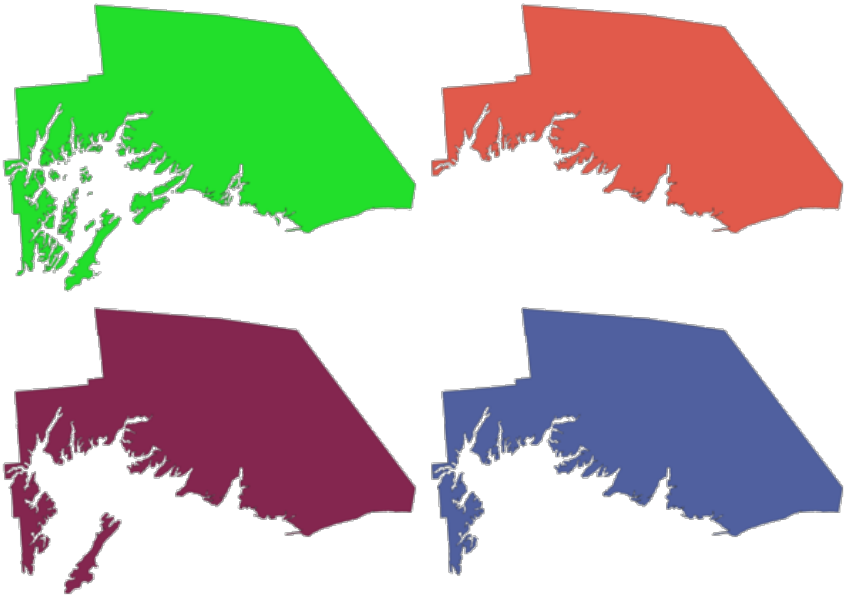


Figura 24.119: En el sentido de las agujas del reloj desde la parte superior izquierda: función multiparte original, una, dos y tres partes más grandes

Parametros

Etiqueta	Nombre	Tipo	Descripción
Polígonos	`INPUT`	[vectorial: poligonal]	Capa vectorial de polígonos entrante
Partes a mantener	`PARTS`	[numérico: entero] Predeterminado: 1	Número de piezas para guardar. Si es 1, solo se mantendrá la mayor parte de la función.
Partes	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial del polígono de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Partes	`OUTPUT`	[vectorial: poligonal]	La capa vectorial poligonal saliente con las N partes mayores de cada entidad

Código Python

Algorithm ID: `native:keepnbiggestparts`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Subcadena de línea

Devuelve la parte de una línea (o curva) que se encuentra entre las distancias de inicio y fin especificadas (medidas desde el principio de la línea)

Los valores Z y M se interpolan linealmente de los valores existentes.

Si se encuentra una geometría multiparte, solo se considera la primera parte al calcular la subcadena.



Figura 24.120: Línea de subcadena con distancia de inicio establecida a 0 metros y distancia de finalización a 250 metros.

Permite *modificación in situ* de objetos espaciales línea

Ver también

Extender líneas

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Distancia inicial	`START_DISTANCE`	[numérico: doble]  Predeterminado: 0.0	Distancia a lo largo de la línea de entrada al punto de inicio de la función de salida
Distancia final	`END_DISTANCE`	[numérico: doble]  Predeterminado: 1.0	Distancia a lo largo de la línea de entrada hasta el punto final de la función de salida
Subcadena	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de la línea de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Subcadena	`OUTPUT`	[vectorial: lineal]	La capa saliente vectorial lineal.

Código Python

Algoritmo ID: `native:linesubstring`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Lineas a polígonos

Genera una capa poligonal usando como anillos poligonales las líneas desde una capa lineal de entrada.

La tabla de atributos de la capa de salida es la misma que la de la capa de entrada.

Menú predeterminado: *Vectorial ► Herramientas de geometría*

Ver también

:referencia:`qgispolygonstolines`, **:referencia:`qgispolygonize`**, **:referencia:`qgisconvertgeometrytype`**

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Polígonos	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial del polígono de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Polígonos	`OUTPUT`	[vectorial: poligonal]	La capa vectorial poligonal saliente

Código Python

Algoritmo ID: `qgis:linestopolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Combinar líneas

Combina todas las partes conectadas de geometrías MultiLineString en simples geometrías LineString.

Si alguna parte de las geometrías MultiLineString de entrada no está conectada, la geometría resultante será una MultiLineString que contiene las líneas que podrían fusionarse y las partes de la línea no conectadas.

Permite *modificación in situ* de objetos espaciales línea

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Mezclado	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de la línea de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Mezclado	`OUTPUT`	[vectorial: lineal]	La saliente (mezclada) capa lineal vectorial.

Código Python

Algoritmo ID: `native:mergelines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Geometría de mínima frontera

Crea geometrías que encierran las entidades desde una capa de entrada. Las características se pueden agrupar por un campo. La capa de salida contendrá una entidad por valor de grupo con una geometría (MBB) que cubra las geometrías de las entidades con un valor coincidente.

Los siguientes tipos de geometrías cercantes son soportados:

- cuadro delimitador (envoltorio)
- rectángulo orientado
- círculo
- casco convexo

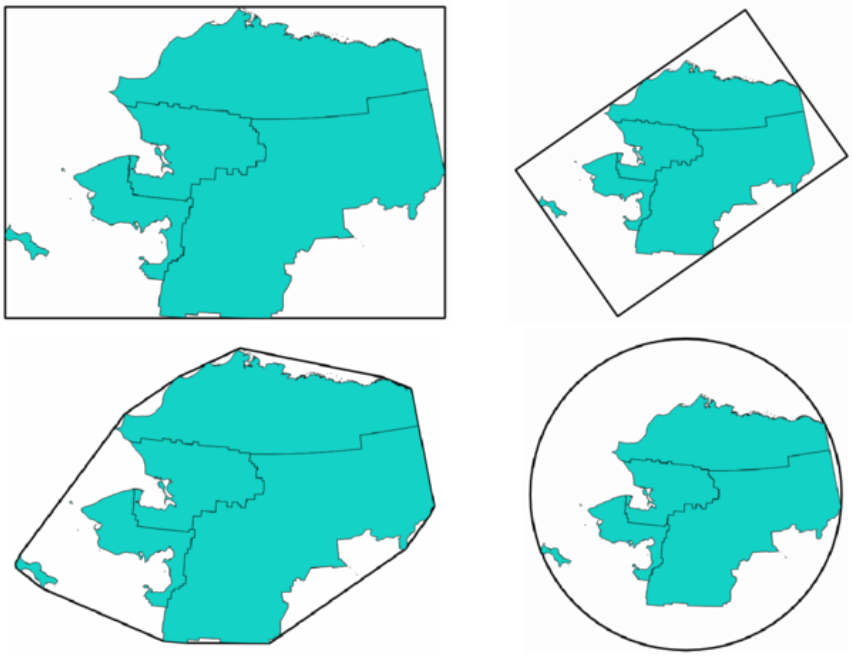


Figura 24.121: En el sentido de las agujas del reloj desde la parte superior izquierda: envoltorio, rectángulo orientado, círculo, casco convexo

Ver también

Cícrculos mínimos cercantes

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Campo Opcional	`FIELD`	[campo de tabla: cualquiera]	Las entidades se pueden agrupar por un campo. Si se establece, esto hace que la capa de salida contenga una entidad por valor agrupado con una geometría mínima que cubra solo las entidades con valores coincidentes.

continúe en la próxima página

Tabla 24.680 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de geometría	`TYPE`	[enumeración] Predeterminado: 0	Tipos de geoemtrías cercantes. Una de: • 0 — Envoltorio (Cuadro delimitador) • 1 — Rectángulo mínimo orientado • 2 — Circulo mínimo cercante • 3 — Casco convexo
Geometrías frontera	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial del polígono de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Geometrías frontera	`OUTPUT`	[vectorial: poligonal]	Capas vectoriales de polígonos de delimitación (salida). Contienen los campos de la capa de entrada y, según el parámetro «TIPO» seleccionado, también incluyen los siguientes atributos: • `anchura`, `altura`, `área` y `perímetro` de la envolvente (cuadro delimitador) • `anchura`, `altura`, `área`, `ángulo` y `perímetro` del rectángulo orientado mínimo. • `radio` y `área` del círculo mínimo circundante • `Área` y `perímetro` del casco convexo

Código Python

Algoritmo ID: `qgis:minimumboundinggeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cícrculos mínimos cercantes

Calcula los círculos mínimos que cercan las entidades en la capa entrante.

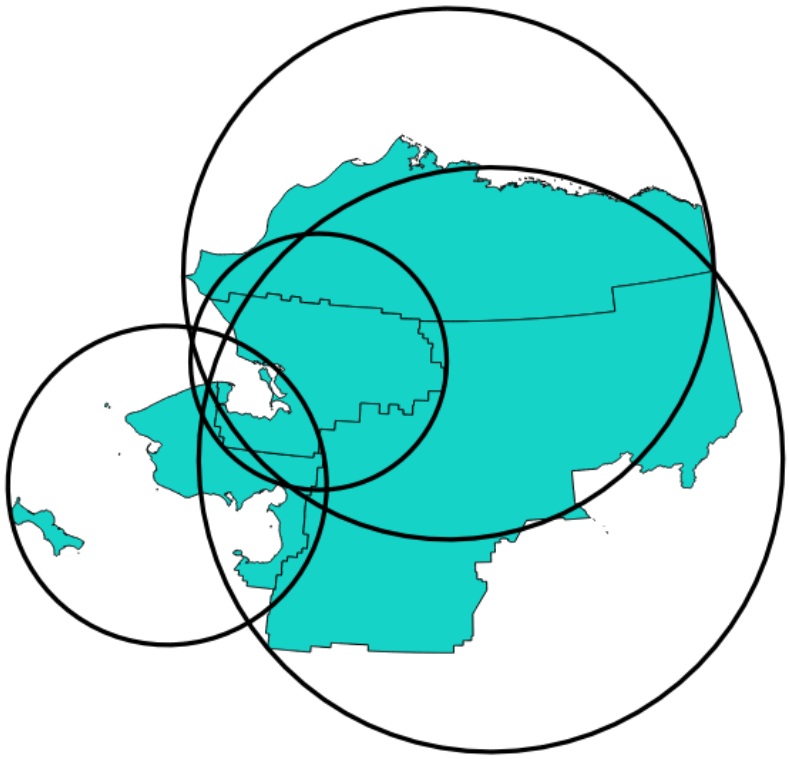


Figura 24.122: Circulos cercantes para cada entidad

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Ver también

Geometría de mínima frontera

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Número de segmentos en círculos	`SEGMENTS`	[numérico: entero] Preestablecido: 72	Número de segmentos utilizados para aproximar un círculo. Mínimo 8, máximo 100000.

continúe en la próxima página

Tabla 24.682 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Círculos mínimos cercantes	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial del polígono de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Círculos mínimos cercantes	`OUTPUT`	[vectorial: poligonal]	La capa vectorial de polígonos de salida. Además de los atributos de entrada, la capa de salida también contiene los siguientes campos: `area` y `radius` del polígono generado.

Código Python

Algoritmo ID: `native:minimumenclosingcircle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Buffer multi-anillos (distancia constante)

Calcula el búfer de múltiples anillos (*donut*) para las características de la capa de entrada, utilizando una distancia fija o dinámica y el número de anillos.

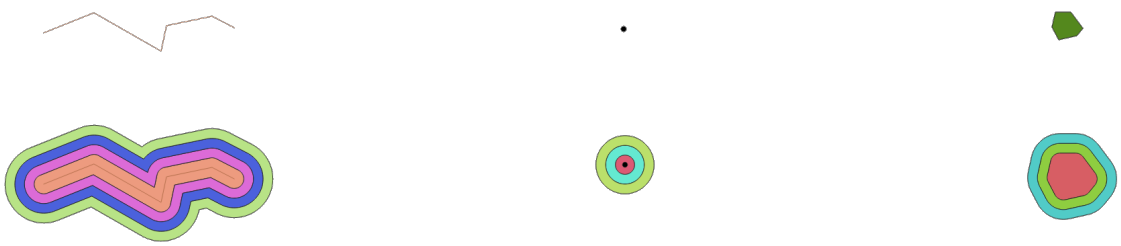


Figura 24.123: Tampón de múltiples anillos para una capa de línea, punto y polígono

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Buffer, *Búfer de distancia variable*, *Rectángulos, óvalos, diamantes*, *Buffer a un solo lado*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Número de anillos	`RINGS`	[numérico: entero] Predeterminado: 1	El número de anillos. Puede ser un valor único (la misma cantidad de anillos para todas las entidades) o puede tomarse de los datos de las entidades (la cantidad de anillos depende de los valores de las entidades).
Distancia entre anillos	`DISTANCE`	[numérico: doble] Predeterminado: 1.0	Distancia entre los anillos. Puede ser un valor único (la misma distancia para todas las entidades) o puede tomarse de los datos de las entidades(la distancia depende de los valores de las entidades).
Buffer de múltiplles anillos (distancia constante)	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial del polígono de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Buffer de múltiplles anillos (distancia constante)	`OUTPUT`	[vectorial: poligonal]	La capa vectorial de polígonos de salida. Además de los atributos de entrada, la capa de salida también contiene los siguientes campos: «ringId» y «distance» del polígono de buffer generado.

Código Python

Algoritmo ID: `native:multiringconstantbuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Multipartes a partes sencillas

Divide entidades multiparte en la capa entrante en entidades sencillas.

Los atributos para la capa saliente son los mismos que los originales pero separados en entidades simples.



Figura 24.124: Izquierda la capa de origen multiparte y a la derecha el resultado de salida de parte simple

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Menú predeterminado: *Vectorial* ► *Herramientas de geometría*

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Coleccionar geometrías, *Elevar a multiparte*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada

continúe en la próxima página

Tabla 24.686 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Partes simples	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial del polígono de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Partes simples	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente.

Código Python

Algoritmo ID: `native:multiparttosingleparts`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Compensado de líneas

Compensa las líneas por una distancia especificada. Las distancias positivas compensarán las líneas a la izquierda, y las distancias negativas las compensarán a la derecha.

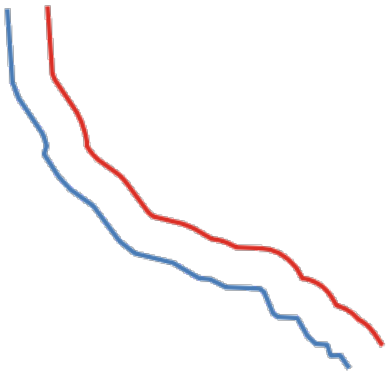


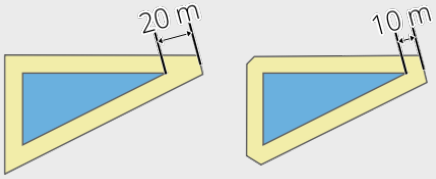
Figura 24.125: En azul la capa fuente, en rojo la compensada

Permite *modificación in situ* de objetos espaciales línea

Ver también

Colección de lineas desplazadas (paralelas), *Traslado*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Distancia	`DISTANCE`	[numérico: doble]  Predeterminado: 10.0	Distancia de compensación. Puede usar el botón Datos definidos a la derecha para elegir un campo a partir del cual se calculará el radio. De esta manera, puede tener un radio diferente para cada característica (ver *Búfer de distancia variable*).
Segmentos	`SEGMENTS`	[numérico: entero] Preestablecido: 8	Controla el número de segmentos de linea a usar para aproximadamente un cuarto de circulo cuando se crean compensaciones redondeadas.
Unir estilo	`JOIN_STYLE`	[enumeración] Predeterminado: 0	Especifica si se deben utilizar uniones redondas, de inglete o biseladas al compensar esquinas en una línea. Las opciones son: • 0 — Redondo • 1 — Inglete • 2 — Bisel  Figura 24.126: Estilos de unión redonda, a inglete y en bisel
Límite de inglete	`MITER_LIMIT`	[numérico: doble] Preestablecido: 2.0	Establezca la distancia máxima desde la geometría de desplazamiento que se utilizará al crear una unión a inglete como factor de la distancia de desplazamiento (sólo aplicable a los estilos de unión a inglete). Mínimo: 1.0  Figura 24.127: Un área de influencia (búfer) de 10 m con un límite de 2 y un área de influencia de 10 m con un límite de 1

continúe en la próxima página

Tabla 24.688 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Compensado	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (desplazamiento). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Compensado	`OUTPUT`	[vectorial: lineal]	Capa saliente lineal (compensada)

Código Python

Algoritmo ID: `native:offsetline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cuador delimitador mínimo orientado

Calcula el rectángulo rotado de área mínima para cada entidad de la capa entrante.

Nota

For singlepart point features, the output corresponds to the bounding box of the geometry.

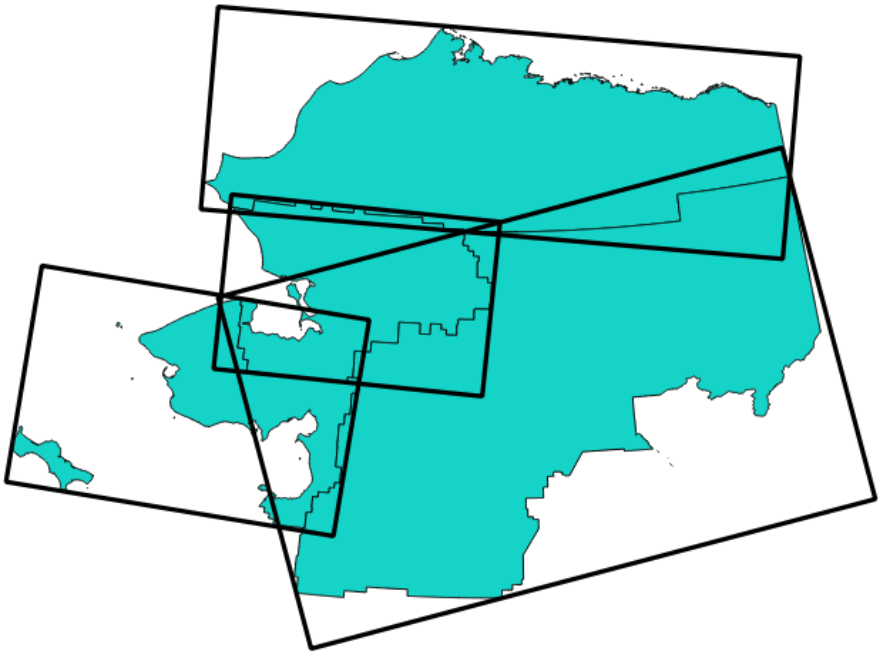


Figura 24.128: Cuador delimitador mínimo orientado

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Ver también

Geometría de mínima frontera

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Cajas frontera	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial del polígono de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cajas frontera	`OUTPUT`	[vectorial: poligonal]	La capa vectorial de polígonos de salida. Además de los atributos de entrada, la capa de salida también contiene los siguientes campos: `width`, `height`, `angle`, `area` y `perimeter` del polígono generado.

Código Python

Algoritmo ID: `native:orientedminimumboundingbox`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ortogonalizar

Intenta ortogonalizar las geometrías de la línea de entrada o capa de polígono. Este proceso desplaza los vértices en las geometrías para tratar de hacer que cada ángulo de la geometría sea un ángulo recto o una línea recta.

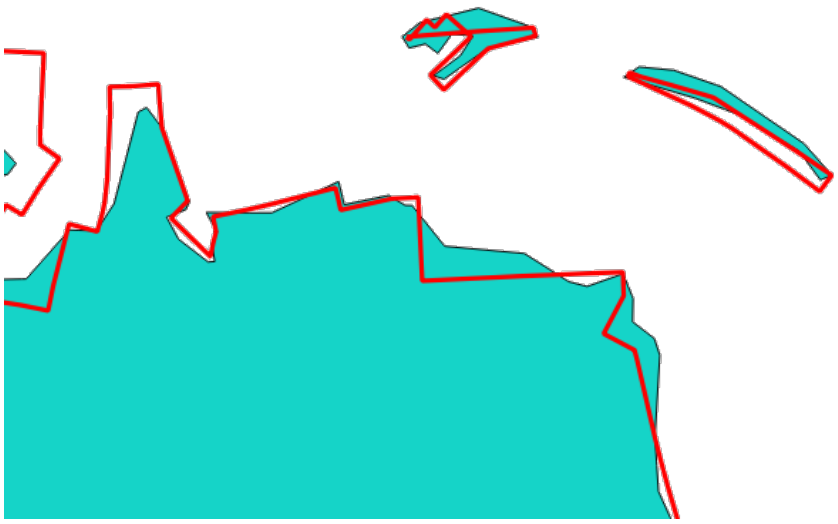


Figura 24.129: En azul la capa fuente y la roja la resultante ortogonalizada

Permite la modificación in situ de las características de . de objetos espaciales líneas y polígonos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Tolerancia de ángulo máximo (grados)	`ANGLE_TOLERANC`	[numérico: doble] Predeterminado: 15,0	Especifique la desviación máxima de un ángulo recto o una línea recta que puede tener un vértice para ser ajustada. Las tolerancias más pequeñas significan que solo se ajustarán los vértices que ya están más cerca de los ángulos rectos, y las tolerancias más grandes significan que los vértices que se desvían más de los ángulos rectos también se ajustarán.
Iteraciones máximas de algortimo	`MAX_ITERATIONS`	[numérico: entero] Preestablecido: 1000	Estableciendo un número mayor para el número máximo de iteraciones dará como resultado una geometría más ortogonal a costa del tiempo de procesamiento adicional.
Ortogonalizado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial del polígono de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ortogonalizado	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente con los ángulos ajustados.

Código Python

Algoritmo ID: `native:orthogonalize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Punto en la superficie

Para cada entidad de la capa entrante, devuelve un punto que está garantizado que descansa sobre la superficie de la geometría de la entidad.

Permite la modificación in situ de *características* de objetos espaciales punto

Advertencia

Este algoritmo puede eliminar las claves primarias o los valores FID existentes y regenerarlos en las capas de salida, dependiendo de los parámetros de entrada.

Ver también

Centroides

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Crear punto en la superficie de cada parte	`ANGLE_TOLERANC`	[booleano]	Si está marcada, un punto será creado para cada parte de la geometría.
Punto	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa de vectores de puntos de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Punto	`OUTPUT`	[vector: punto]	La capa vectorial saliente de puntos..

Código Python

Algoritmo ID: `native:pointonsurface`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Puntos a lo largo de la geometrías

Crea puntos a intervalos regulares a lo largo de geometrías de línea o polígono. Los puntos creados tendrán nuevos atributos agregados para la distancia a lo largo de la geometría y el ángulo de la línea en el punto.

Se puede especificar un desplazamiento opcional de inicio y fin, que controla cuán lejos del inicio y el final de la geometría deben crearse los puntos.

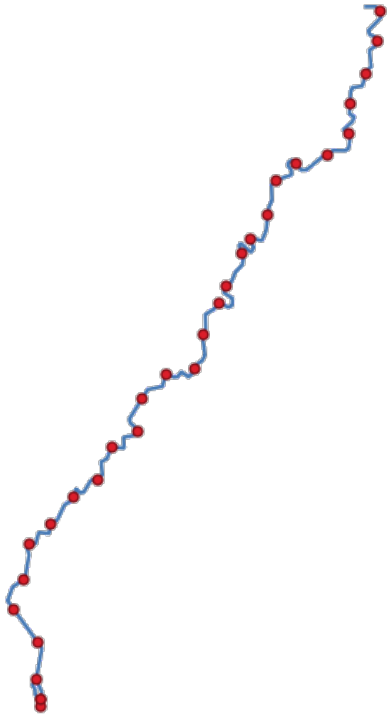


Figura 24.130: Puntos creados a lo largo de la capa de líneas fuente

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Interpolar puntos en línea

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Distancia	`DISTANCE`	[numérico: doble] Predeterminado: 1.0	Distancia entre dos puntos consecutivos a lo largo de la línea
Compensado de inicio	`START_OFFSET`	[numérico: doble] Predeterminado: 0.0	Distancia desde el comienzo de la línea de entrada, que representa la posición del primer punto.
Compensado final	`END_OFFSET`	[numérico: doble] Predeterminado: 0.0	Distancia desde el final de la línea de entrada, que representa la posición más allá de la cual no se debe crear ninguna característica de punto.
Puntos interpolados	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos interpolados	`OUTPUT`	[vector: punto]	Capa vectorial de puntos con elementos situados a lo largo de líneas o límites poligonales de la capa de entrada. Además de los atributos de entrada de cada punto relacionado, la capa de salida también contiene los siguientes campos: `distance` y `angle` del punto generado.

Código Python

Algoritmo ID: `native:pointsalonglines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Desplazamiento de puntos

Dada una distancia de proximidad, identifica los puntos cercanos y los distribuye radialmente sobre un círculo cuyo centro representa su baricentro. Una herramienta conveniente para dispersar entidades superpuestas.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Distancia mínima a otros puntos	`PROXIMITY`	[numérico: doble] Predeterminado: 1.0	Distancia por debajo de la cual las entidades de punto se consideran cercanas. Las entidades cercanas se distribuyen por completo.
Distancia de desplazamiento	`DISTANCE`	[numérico: doble] Predeterminado: 1.0	Radio del círculo en el que se colocan las entidades cercanas
Distribución horizontal para caso de dos puntos	`HORIZONTAL`	[booleano] Predeterminado: Falso	Cuando solo dos puntos se identifican como cercanos, los alinea horizontalmente en el círculo en lugar de verticalmente.
Desplazado	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Desplazado	`OUTPUT`	[vector: punto]	Capa vectorial saliente de puntos

Código Python

Algoritmo ID: `qgis:pointsdisplacement`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Polo de inaccesibilidad

Calcula el polo de inaccesibilidad para una capa de polígono, que es el punto interno más distante del límite de la superficie.

Este algoritmo utiliza el algoritmo "polylabel" (Vladimir Agafonkin, 2016), que es un enfoque iterativo garantizado para encontrar el verdadero polo de inaccesibilidad dentro de una tolerancia especificada. Una tolerancia más precisa (valor más bajo) requiere más iteraciones y costará más tiempo calcularla.



Figura 24.131: Polo de inaccesibilidad

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa de vector de entrada
Tolerancia	`TOLERANCE`	[numérico: doble] Predeterminado: 1.0	Establece la tolerancia para el cálculo
Punto	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial del polígono de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Punto	`OUTPUT`	[vector: punto]	La capa de vectores del punto de salida. Contendrá un nuevo atributo con la distancia `dist_pole` desde el polo calculado hasta el límite del polígono.

Código Python

Algoritmo ID: `native:poleofinaccessibility`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Poligonizar

Crea una capa de polígono cuyos límites de entidades se generan a partir de una capa de línea de entidades **cerradas**.

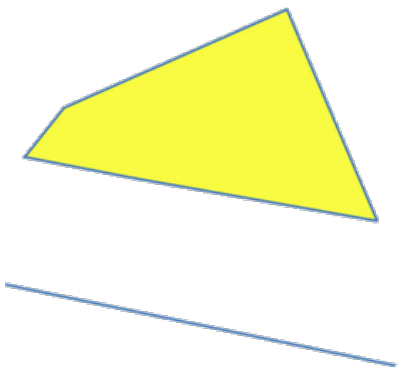


Figura 24.132: Los polígonos amarillos generados de líneas cerradas

Nota

La capa de línea debe tener formas cerradas para transformarse en un polígono.

Ver también

Polígonos a líneas, *Lineas a polígonos*, *Convierte el tipo de geometría*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Mantener campos de la capa de entrada	`KEEP_FIELDS`	[booleano] Predeterminado: Falso	Comprobar para mantener los campos (sólo la estructura de la tabla, no los valores) de la capa de entrada
Polígonos desde líneas	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial del polígono de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Polígonos desde líneas	`OUTPUT`	[vectorial: poligonal]	La capa vectorial poligonal saliente desde líneas

Código Python

Algoritmo ID: `native:polygonize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Polígonos a líneas

Toma una capa poligonal y crea una capa lineal, con líneas representando las fronteras de los polígonos en la capa entrante.

La tabla de atributos de la capa de salida es la misma que la de la capa de entrada.



Figura 24.133: Líneas negras como resultado del algoritmo

Menú predeterminado: *Vectorial ► Herramientas de geometría*

Ver también

Lineas a polígonos, *Poligonizar*, *Convierte el tipo de geometría*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa vectorial de polígonos entrante
Líneas	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de la línea de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Líneas	`OUTPUT`	[vectorial: lineal]	La capa vectorial lineal saliente desde polígonos

Código Python

Algoritmo ID: *native:polygonstolines`*

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Proyección de puntos (Cartesiano)

Proyectar geometrías puntutales por una distancia especificada y rumbo (azimut).

Permite la modificación in situ de *características* de objetos espaciales punto

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Rumbo (grados del Norte)	`BEARING`	[numérico: doble] Predeterminado: 0.0	Ángulo en sentido de las agujas del reloj desde el Norte, en unidades de grados(º)
Distancia	`DISTANCE`	[numérico: doble] Predeterminado: 1.0	Distancia para compensar geometrías, en unidades de la capa
Proyectados	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifique la capa de vectores de puntos de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Proyectados	`OUTPUT`	[vector: punto]	La capa vectorial de puntos (proyectados) saliente

Código Python

Algoritmo ID: `native:projectpointcartesian`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Elevar a multiparte

Toma una capa vectorial con geometrías de partes simples y genera una nueva en la cual las geometrías son multiparte.

Entidades entrantes que son todavía entidades multiparte se mantendrán sin cambios.

Este algoritmo puede ser usado para forzar geometrías a tipos multiparte en orden a ser compatible con proveedores de datos que requieren entidades multiparte.

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Ver también

Agregar, *Coleccionar geometrías*, *Convert to strictly multipart*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Multipartes	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial multiparte de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Multipartes	`OUTPUT`	[la misma que la entrada]	La capa vectorial multiparte saliente

Código Python

Algoritmo ID: `native:promotetomulti`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Rectángulos, óvalos, diamantes

Crea un área de influencia con forma de rectángulo, óvalo o diamante para cada entidad de la capa de puntos de entrada.

Los parámetros de forma se pueden fijar para todas las entidades o dinámicas mediante un campo o una expresión.

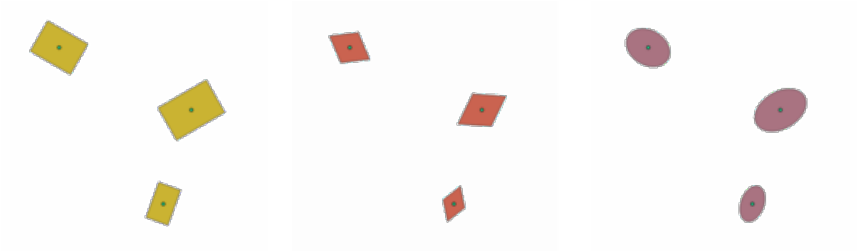


Figura 24.134: Diferentes formas de búfer con parámetros dinámicos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Buffer de forma	`SHAPE`	[enumeración]	La forma a usar. Una de: • 0 — Rectángulos • 1 — Óvalos • 2 — Diamantes
Anchura	`WIDTH`	[numérico: doble] Predeterminado: 1.0	Anchura del buffer de forma
Altura	`HEIGHT`	[numérico: doble] Predeterminado: 1.0	Altura del buffer de forma
Rotación Opcional	`ROTATION`	[numérico: doble] Predeterminado: 0.0	Rotación del buffer de forma
Número de segmentos	`SEGMENTS`	[numérico: entero] Preestablecido: 36	Número de segmentos para un círculo completo (forma *Ovalada*)

continúe en la próxima página

Tabla 24.710 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Salida	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Salida	`OUTPUT`	[vectorial: poligonal]	La capa vectorial saliente (con las formas del buffer)

Código Python

Algoritmo ID: `native:rectanglesovalsdiamonds`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Eliminar vértices duplicados

Borra vértices duplicados de entidades, siempre y cuando el borrado de los vértices no derive en una geometría degenerada.

El parámetro tolerancia especifica la tolerancia para coordenadas al determinar si unos vértices son idénticos.

Por defecto, los valores Z no se tienen en cuenta al detectar vértices duplicados. P.ej. dos vértices con la misma coordenada X e Y pero diferentes valores de Z aún se considerarán duplicados y uno se eliminará. Si el parámetro :guvalor Zilabel:`Usar ` es verdadero, entonces los valores Z también se prueban y los vértices con la misma X e Y pero diferente Z se mantendrán.

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Nota

Los vértices duplicados no se prueban entre diferentes partes de una geometría multiparte, p. ejemplo este método no cambiará una geometría multipunto con puntos superpuestos.

Ver también

Extraer vértices, *Extráe vértices específicos*, *Borrar geometrías duplicadas*, *Delete duplicated vertices*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Tolerancia	`TOLERANCE`	[numérico: doble] Preestablecido: 0.000001	Vértices mas cercanos que la distancia especificada se consideran duplicados
Usar valor Z	`USE_Z_VALUE`	[booleano] Predeterminado: Falso	Si el parámetro *Usar valor Z* es verdadero, entonces los valores Z también se prueban y los vértices con la misma X e Y pero diferente Z se mantendrán.
Limpia	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Limpia	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (sin vértices duplicados)

Código Python

Algoritmo ID: `native:removeduplicatevertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Borra geometrías nulas

Elimina cualquier entidad que no tenga geometría de una capa vectorial. Todas las demás entidades se copiarán sin cambios.

Las entidades con geometrías nulas pueden ser guardadas en una capa aparte.

Si *Eliminar también geometrías vacías* está marcado, el algoritmo elimina entidades cuyas geometrías no tienen coordenadas, es decir, geometrías que están vacías. En ese caso, también la salida nula reflejará esta opción, que contiene geometrías nulas y vacías.

Ver también

Borrar geometrías duplicadas

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa vectorial entrante (con geometrías no-NULAS)
También borrar geoemtrías vacías	`REMOVE_EMPTY`	[booleano]	
Geometrías no nulas	`OUTPUT` Opcional	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida para las geometrías no NULL (y no vacías). *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Geometrías nulas Opcional	`NULL_OUTPUT`	[la misma que la entrada] Preestablecido: `[Omitir salida]`	Especifique la capa vectorial de salida para las geometrías NULL (y vacías). *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Geometrías nulas	`NULL_OUTPUT`	[la misma que la entrada]	Capa vectorial saliente (para NULL y, si son elegidas, entidades vacías)
Geometrías no nulas	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (sin NULL y, si son elegidas, entidades vacías)

Código Python

Algoritmo ID: `native:removenullgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Línea de dirección reversible

Invierte la dirección de una capa lineal.

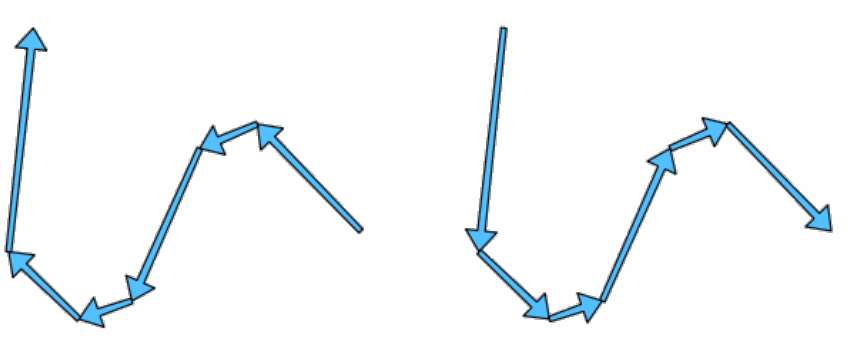


Figura 24.135: Antes y después de la inversión de dirección

Permite *modificación in situ* de objetos espaciales línea

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas

continúe en la próxima página

Tabla 24.716 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Reversión	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de la línea de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Reversión	`OUTPUT`	[vectorial: lineal]	La capa vectorial saliente (con líneas invertidas)

Código Python

Algoritmo ID: `native:reverselinedirection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Rotar

Rota entidades de geometría en el ángulo horario especificado. La rotación sucede alrededor del centroide de cada entidad, u opcioálmente alrededor de un único punto preestablecido.

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Ver también

Traslado, *Intercmbiar coordenadas X e Y*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Rotación (grados en el sentido horario)	`ANGLE`	[numérico: doble]  Predeterminado: 0.0	Ángulo de rotación en grados
Anchura del punto de rotación (x,y) Opcional	`ANCHOR`	[coordenadas] Predeterminado: No establecido	Las coordenadas X,Y del punto alrededor del cuál rotar las entidades. Si no se establece la rotación se llevará a cabo alrededor del centroide de cada entidad.
Rotado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida (con geometrías rotadas). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Rotado	`OUTPUT`	[la misma que la entrada]	La capa vecotiral saliente con las geometrías rotadas

Código Python

Algoritmo ID: `native:rotatefeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Redondez

Calcula la redondez de cada característica y la almacena como un nuevo campo denominado `roundness`. La capa vectorial de entrada debe contener polígonos.

La redondez de un polígono se define como $4\pi \times$ área del polígono / perímetro^2. El valor de la redondez varía entre 0 y 1. Un círculo perfecto tiene una redondez de 1, mientras que un polígono completamente plano tiene una redondez de 0.

Nota

El algoritmo devuelve NULL para entidades poligonales multiparte.

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa de vector de entrada
Redondez	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida (con el campo de redondez). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Redondez	`OUTPUT`	[la misma que la entrada]	La capa vectorial de salida con el valor de `redondez` en un campo

Código Python

ID de algoritmo: `native:redondez`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Segmentar por ángulo máximo

Segmentar una geometría por conversión de secciones a secciones lineales.

La segmentación se realiza especificando el ángulo de radio máximo permitido entre vértices en la geometría enderezada (por ejemplo, el ángulo del arco creado desde el centro del arco original a los vértices de salida consecutivos en la geometría linealizada). Las geometrías no curvas se retendrán sin cambios.

Ver también

Segmentar por distancia máxima, *Simplificar*, *Suavizar*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Ángulo máximo entre vértices (ángulos)	`ANGLE`	[numérico: doble]  Preestablecido: 5.0	Máximo radio angular permitido entre vértices en la geometría enderezada
Segmentado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida (con geometrías segmentadas). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Segmentado	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente con geometrías segmentadas

Código Python

Algoritmo ID: `native:segmentizebymaxangle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Segmentar por distancia máxima

Segmentar una geometría por conversión de secciones a secciones lineales.

La segmentación se realiza especificando la distancia de desplazamiento máxima permitida entre la curva original y la representación segmentada. Las geometrías no curvas se retendrán sin cambios.

Ver también

Segmentar por ángulo máximo, *Simplificar*, *Suavizar*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Máxima distancia de desplazamiento	`DISTANCE`	[numérico: doble]  Predeterminado: 1.0	Distancia de desplazamiento máxima permitida entre la curva original y la representación segmentada, en las unidades de capa.
Segmentado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida (con geometrías segmentadas). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Segmentado	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente con geometrías segmentadas

Código Python

Algoritmo ID: `native:segmentizebymaxdistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

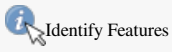
Establecer valor M

Define el valor de M para las geometrías de una capa.

Si ya existen valores M en la capa, se sobrescribirán con el nuevo valor. Si no existen valores M, la geometría se actualizará para incluir los valores M y el valor especificado utilizado como el valor M inicial para todas las geometrías.

Permite *modificación in situ* de objetos espaciales punto, línea, y polígono con M habilitada

Truco

Use el botón 

para verificar el valor M agregado: los resultados están disponibles en el diálogo *Identify Results*.

Ver también

Establecer el valor de M desde ráster, *Establecer valor Z*, *Soltar valores M/Z*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Valor M	`M_VALUE`	[numérico: doble] Predeterminado: 0.0	Valor M a asignar a las geometrías de la entidad
M Añadida	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
M Añadida	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (con los valores M asignados a las geometrías)

Código Python

Algoritmo ID: `native:setmvalue`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Establecer el valor de M desde ráster

Utiliza valores muestreados de una banda dentro de una capa ráster para establecer el valor M para cada vértice superpuesto en la geometría de la entidad. Los valores rasterizados se pueden escalar opcionalmente por una cantidad preestablecida.

Si ya existen valores M en la capa, se sobrescribirán con el nuevo valor. Si no existen valores de M, la geometría se actualizará para incluir valores de M.

Permite *modificación in situ* de objetos espaciales punto, línea, y polígono con M habilitada

Ver también

Drapeado (establecer el valor Z del ráster), *Establecer valor M*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Capa ráster	`RASTER`	[ráster]	Capa ráster con valores M
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	La bana ráster desde la cual los valores M son tomados
Valor para NoData o vértices no intersecantes	`NODATA`	[numérico: doble] Predeterminado: 0.0	Valor a usar en caso de que el vértice no intersecte (un pixel valido de) el ráster
Factor de escala	`SCALE`	[numérico: doble] Predeterminado: 1.0	Valor de escalado: los valores de banda se multiplican por este valor.
Compensado	`OFFSET`	[numérico: doble] Predeterminado: 0.0	Valor de desplazamiento: se añade algebraicamente a los valores de la banda tras aplicar el «Factor de escala».

continúe en la próxima página

Tabla 24.728 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Actualizado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida (con los valores M actualizados). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Actualizado	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (con valores M actualizados)

Código Python

Algoritmo ID: `native:setmfromraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Establecer valor Z

Establece el valor Z para las geometrías de una capa.

Si ya existen valores Z en la capa, se sobrescribirán con el nuevo valor. Si no existen valores Z, la geometría se actualizará para incluir valores Z y el valor especificado utilizado como el valor Z inicial para todas las geometrías.

Permite *modificación in situ* de objetos espaciales punto, línea, y polígono con Z habilitada

Truco

Use el botón Identify Features para verificar el valor Z agregado: los resultados están disponibles en el diálogo *Identify Results*.

Ver también

Drapeado (establecer el valor Z del ráster), *Establecer valor M*, *Soltar valores M/Z*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Valor Z	`Z_VALUE`	[numérico: doble]  Predeterminado: 0.0	Valor Z a asignar a las geoemtrías de entidad
Z añadido	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Z añadido	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (con valores Z asignados)

Código Python

Algoritmo ID: `native:setzvalue`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Simplificar

Simplifica las geometrías en una línea o capa de polígono. Crea una nueva capa con las mismas características que las de la capa de entrada, pero con geometrías que contienen un número menor de vértices.

El algoritmo ofrece una variedad de métodos de simplificación, que incluyen la distancia (el algoritmo «Douglas-Peucker»), el área (algoritmo «Visvalingam») y el ajuste de geometrías a la cuadrícula.



Figura 24.136: De izquierda a derecha, capa de origen y tolerancias de simplificación crecientes

Permite la modificación in situ de las características de . de objetos espaciales líneas y polígonos

Menú predeterminado: *Vectorial ► Herramientas de geometría*

Ver también

Simplificar la cobertura, *Suavizar*, *Densificar por conteo*, *Densificar por intervalo*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Método de simplificación	`METHOD`	[enumeración] Predeterminado: 0	Método de simplificación. Uno de: • 0 — Distancia (Douglas-Peucker) • 1 — Ajustar a cuadrícula • 2 — Área (Visvalingam)
Tolerancia	`TOLERANCE`	[numérico: doble] Predeterminado: 1.0	Tolerancia de umbral (en unidades de la capa): si la distancia entre dos nodos es menor que el valor de tolerancia, el segmento se simplificará y se eliminarán los vértices.
Simplificado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida (simplificada). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Simplificado	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente (simplificada)

Código Python

Algoritmo ID: `native:simplifygeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Buffer a un solo lado

Calcula un buffer en líneas por una distancia especificada en un solo lado de la línea.

El buffer siempre resulta en una capa poligonal.

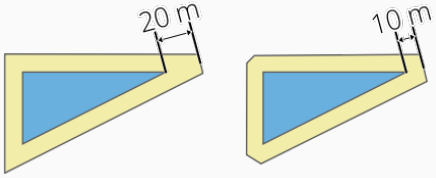


Figura 24.137: Izquierda versus lado derecho en la misma capa vectorial lineal

Ver también

Buffer

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Distancia	`DISTANCE`	[numérico: doble]  Predeterminado: 10.0	Distancia de buffer.
Lado	`SIDE`	[enumeración] Predeterminado: 0	Lado del cuál crear el buffer. Uno de: • 0 – Izquierdo • 1 – Derecho
Segmentos	`SEGMENTS`	[numérico: entero] Preestablecido: 8	Controla el número de segmentos de linea a usar para aproximadamente un cuarto de circulo cuando se crean compensaciones redondeadas.
Unir estilo	`JOIN_STYLE`	[enumeración] Predeterminado: 0	Especifica si se deben utilizar uniones redondas, de inglete o biseladas al compensar esquinas en una línea. Las opciones son: • 0 — Redondo • 1 — Inglete • 2 — Bisel  Figura 24.138: Estilos de unión redonda, a inglete y en bisel
Límite de inglete	`MITER_LIMIT`	[numérico: doble] Preestablecido: 2.0	Establezca la distancia máxima desde la geometría de desplazamiento que se utilizará al crear una unión a inglete como factor de la distancia de desplazamiento (sólo aplicable a los estilos de unión a inglete). Mínimo: 1.0  Figura 24.139: Un área de influencia (búfer) de 10 m con un límite de 2 y un área de influencia de 10 m con un límite de 1

continúe en la próxima página

Tabla 24.734 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Buffer	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (búfer). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Buffer	`OUTPUT`	[vectorial: poligonal]	Salida capa poligonal (buffer)

Código Python

Algoritmo ID: `native:singlesidedbuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Suavizar

Suaviza las geometrías en una línea o capa de polígono agregando más ** vértices y esquinas ** a las geometrías de entidades.

El parámetro de iteraciones dicta cuántas iteraciones de suavizado se aplicarán a cada geometría. Un mayor número de iteraciones da como resultado geometrías más suaves con el costo de un mayor número de nodos en las geometrías.

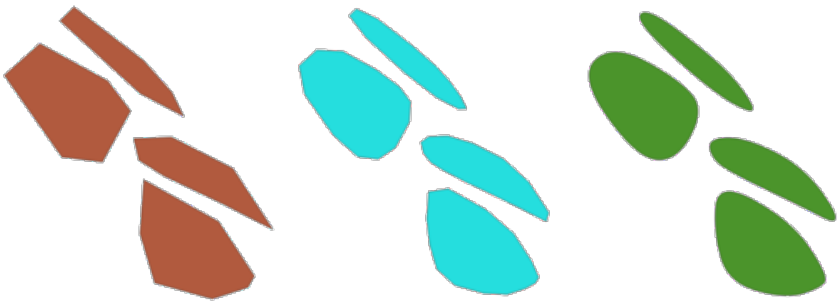


Figura 24.140: Aumentando el número de iteraciones provoca geometrías mas suaves

El parámetro offset controla cuán «fielmente» las geometrías suavizadas siguen las geometrías originales. Los valores más pequeños dan como resultado un ajuste mayor, y los valores más grandes crearán un ajuste más flojo.

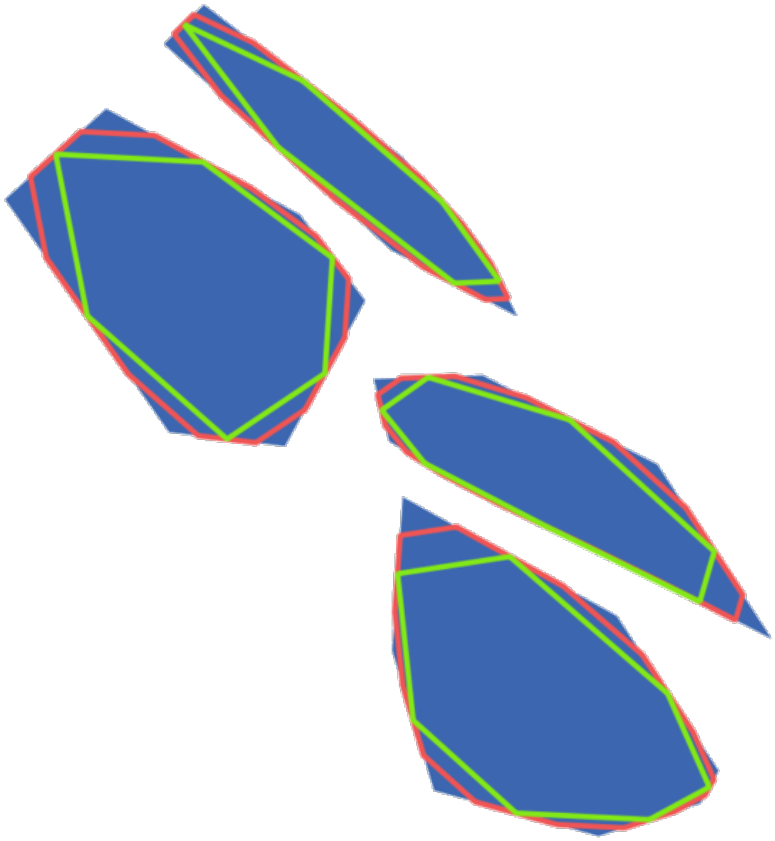


Figura 24.141: Azul: la capa entrante. Compensado de 0.25 da la línea roja, mientras que compensado de 0.50 da la línea verde.

El parámetro de ángulo máximo se puede utilizar para evitar el suavizado de nodos con ángulos grandes. Cualquier nodo donde el ángulo de los segmentos a ambos lados sea mayor que este no se suavizará. Por ejemplo, establecer el ángulo máximo a 90 grados o menos conservaría los ángulos rectos en la geometría.

Permite la modificación in situ de las características de . de objetos espaciales líneas y polígonos

Ver también

Simplificar, *Simplificar la cobertura*, *Densificar por conteo*, *Densificar por intervalo*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Línea de entrada o capa de vector poligonal
Iteraciones	`ITERATIONS`	[numérico: entero] Predeterminado: 1	Aumentar el número de iteraciones dará geometrías más suaves (y más vértices).

continúe en la próxima página

Tabla 24.736 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Compensado	`OFFSET`	[numérico: doble] Preestablecido: 0.25	Los valores crecientes *moverán* las líneas / límites suavizados más lejos de las líneas / límites de entrada.
Ángulo máximo de nodo para suavizar	`MAX_ANGLE`	[numérico: doble] Preestablecido: 180.0	Cada nodo por debajo de este valor se suavizará
Suavizado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (suavizada). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Suavizado	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente (suavizada)

Código Python

Algoritmo ID: `native:smoothgeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ajustar geoemtrías a capa

Ajusta las geometrías de una capa a las geometrías de otra capa o a las geometrías dentro de la misma capa.

El emparejamiento se realiza en base a una distancia de tolerancia, y los vértices se insertan o se eliminan según sea necesario para que las geometrías coincidan con las geometrías de referencia.

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Ver también

Ajustar puntos a rejilla

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	INPUT	[vector: geometría]	Capa de vector de entrada
Capa de referencia	REFERENCE_LAYE	[vector: geometría]	Capa vectorial a ajustar
Tolerancia	TOLERANCE	[numérico: doble] Predeterminado: 10.0	Controla como de cerca necesitan estar los vértices entrantes para ser a las geometrías de la capa referente antes de que se ajuste.

continúe en la próxima página

Tabla 24.738 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Comportamiento	`BEHAVIOR`	[enumeración] Predeterminado: 0	El ajuste se puede hacer a un nodo existente o un segmento (su punto más cercano al vértice para moverse). Opciones de ajuste disponibles: • 0 — Preferir alinear nodos, inserte vértices adicionales donde sea necesario Prefiere ajustarse a los nodos, incluso cuando un segmento puede estar más cerca que un nodo. Se insertarán nuevos nodos para que las geometrías se sigan exactamente cuando estén dentro de la tolerancia permitida. • 1 — Preferir el punto más cercano, inserte vértices adicionales donde sea necesario Ajustar al punto más cercano, independientemente de que sea un nodo o un segmento. Se insertarán nuevos nodos para que las geometrías se sigan exactamente cuando estén dentro de la tolerancia permitida. • 2 — Preferir alinear nodos, no insertar nuevos vértices Prefiere ajustarse a los nodos, incluso cuando un segmento puede estar más cerca que un nodo. No se insertarán nuevos nodos. • 3 — Prefiere el punto más cercano, no inserte vértices nuevos Ajustar al punto más cercano, independientemente de que sea un nodo o un segmento. No se insertarán nuevos nodos. • 4 — Mover solo puntos finales, preferir nodos alineados Sólo ajustar los puntos de inicio/fin de las líneas (las características de los puntos también se ajustarán, las características de los polígonos no se modificarán), prefiera ajustar los nodos. • 5 — Mover solo puntos finales, preferir puntos más próximos Ajustar sólo los puntos de inicio/fin de las líneas (las características de los puntos también se ajustarán, las características de los polígonos no se modificarán), ajustar al punto más cercano • 6 — Ajustar puntos finales a solo puntos finales Sólo encaja los puntos de inicio/fin de las líneas con otros puntos de inicio/fin de las líneas • 7 — Ajustar a nodos de anclaje (solo una capa)

continúe en la próxima página

Tabla 24.738 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Geometría ajustada	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (ajustada). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Geometría ajustada	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente (ajustada)

Código Python

Algoritmo ID: `native:snapgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ajustar puntos a rejilla

Modifica las coordenadas de geometrías en una capa vectorial, de tal manera que todos los puntos o vértices al punto mas cercano de la rejilla.

Si la geometría ajustada no se puede calcular (o está totalmente contraída), la geometría de la entidad se borrará.

El ajuste se puede realizar en los ejes X, Y, Z o M. Un espacio de cuadrícula de 0 para cualquier eje deshabilitará el ajuste para ese eje.

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Advertencia

Ajustando a la rejilla puede generar una geometría inválida en algunos casos de esquinas.

Ver también

Ajustar geoemtrías a capa

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Espaciado X de rejilla	`HSPACING`	[numérico: doble] Predeterminado: 1.0	Espaciado de rejilla en el eje X
Espaciado Y de rejilla	`VSPACING`	[numérico: doble] Predeterminado: 1.0	Espaciado de rejilla en el eje Y
Espaciado Z de rejilla	`ZSPACING`	[numérico: doble] Predeterminado: 0.0	Espaciado de rejilla en el eje Z
Espaciado M de rejilla	`MSPACING`	[numérico: doble] Predeterminado: 0.0	Espaciado de rejilla en el eje M
Ajustado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (ajustada). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Ajustado	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente (ajustada)

Código Python

Algoritmo ID: `native:snappointstogrid`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cortar líneas por longitud máxima

Toma una capa de línea (o curva) y divide cada entidad en varias partes, donde cada parte tiene una longitud máxima especificada. Los valores Z y M al inicio y al final de las nuevas subcadenas de línea se interpolan linealmente a partir de los valores existentes.

Permite *modificación in situ* de objetos espaciales línea

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	La capa vectorial lienal entrante
Máxima longitud lineal	`LENGTH`	[numérico: doble] Predeterminado: 10.0	La máxima longitud de una línea en la salida.
Cortar	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de la línea de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cortar	`OUTPUT`	[vectorial: lineal]	La nueva capa de vectores de línea. La longitud de las geometrías de las características es menor o igual que la longitud especificada en el parámetro LENGTH. Se añadirá un campo adicional que contiene el `order`.

Código Python

Algoritmo ID: `native:splitlinesbylength`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Subdividir

Subdivide la geometría. La geometría devuelta será una colección que contiene partes subdivididas de la geometría original, donde ninguna parte tiene más que el número máximo especificado de nodos.

Esto es útil para dividir una geometría compleja en partes menos complejas, más fáciles de indexar espacialmente y más rápido para realizar operaciones espaciales. Las geometrías curvas se segmentarán antes de la subdivisión.

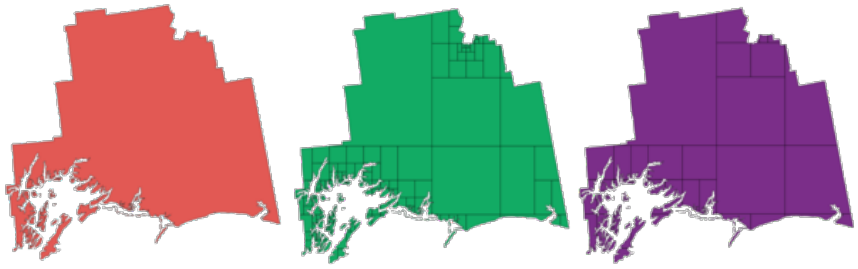


Figura 24.142: A la izquierda de la capa de entrada, el valor medio máximo de los nodos es 100 y el valor máximo derecho es 200

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Nota

Subdividir una geometría puede generar partes de geometría que pueden no ser válidas y pueden contener intersecciones propias.

Ver también

Explotar lineas, *Subcadena de línea*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	La capa vectorial entrante
Máximos nodos en partes	`MAX_NODES`	[numérico: entero] Predeterminado: 256	Número máximo de vértices que cada parte de geometría nueva puede tener. Menos * subpartes * para valores más altos.

continúe en la próxima página

Tabla 24.744 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Subdividido	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida (subdividida). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Subdividido	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente

Código Python

** Algoritmo ID**: `native:subdivide`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Intercmbiar coordenadas X e Y

Intercambia los valores de coordenadas X e Y en geoemtrías entrantes..

Puede emplearse para reparar geometrías que accidentálmente tienen sus valores latitud y longitud invertidos.

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Ver también

Traslado, *Rotar*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	La capa vectorial entrante
Intercambiados	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Intercambiados	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente (intercambiada)

Código Python

** Algoritmo ID**: `native:swapxy`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Zonas de influencia estrechas

Crea zonas de influencia estrechas a lo largo de geometrías lineales, usando un diámetro de la zona de influencia inicial y final.

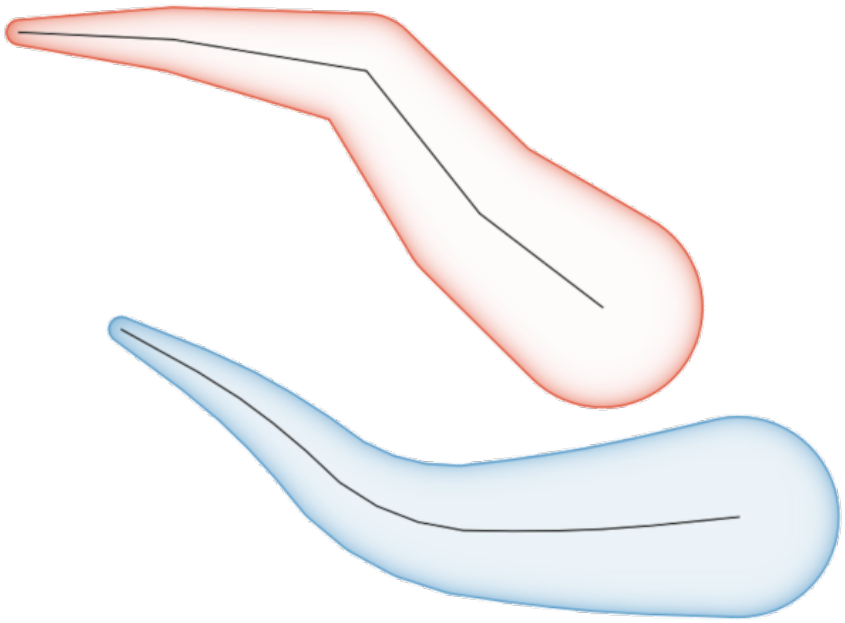


Figura 24.143: Ejemplo de zona de influencia estrecha

Ver también

Ancura de buffer variable (por valor de M), *Buffer*, *Crear buffer de cuñas*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Anchura inicial	`START_WIDTH`	[numérico: doble] Predeterminado: 0.0	Representa el rado del buffer aplicado al punto inicial de la linea de la entidad
Anchura final	`END_WIDTH`	[numérico: doble] Predeterminado: 0.0	Representa el radio del buffer aplicado al punto final de la línea de la entidad.
Segmentos	`SEGMENTS`	[numérico: entero] Preestablecido: 16	Controla el número de segmentos de linea a usar para aproximadamente un cuarto de circulo cuando se crean compensaciones redondeadas.

continúe en la próxima página

Tabla 24.748 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Buffered	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (búfer). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Buffered	`OUTPUT`	[vectorial: poligonal]	Salida capa poligonal (buffer)

Código Python

Algoritmo ID: `native:taperedbuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Teselar

Tesela una capa geométrica poligonal, dividiendo las geoemtrías en componentes triangulares.

La capa saliente consiste en geometrías multipoligonales para cada entidad entrante, con cada multipolígono formado por múltiples polígonos de componentes triangulares.

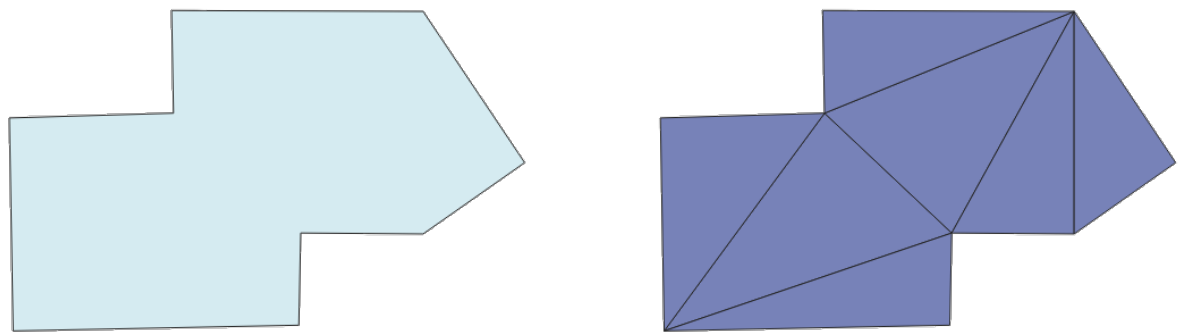


Figura 24.144: polígono teselado (derecha)

Permite *modificación in situ de objetos espaciales.* de objetos espaciales polígonos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: poligonal]	Capa vectorial de polígonos entrante
Teselado	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifica la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Teselado	`OUTPUT`	[vectorial: poligonal]	Capa saliente multipoligonalZ

Código Python

Algoritmo ID: `3d:tessellate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Transecto

Crea transecto en vértices para (multi)linestring.

Un transecto es una línea orientada desde un ángulo (por defecto perpendicular) a la polilínea de entrada (a vértices).

Campo(s) desde entidad(es) son devueltos en el transecto con estos nuevos campos:

- TR_FID: ID de la entidad original
- TR_ID: ID del transecto. Cada transecto tiene una única ID
- TR_SEGMENT: ID del segmento de la cadena lineal
- TR_ANGLE: Ángulo en grados desde la línea original al vértice
- TR_LENGTH: Longitud total devuelta del transecto
- TR_ORIENT: Lado del transecto (sólo a la derecha o a la izquierda de la línea, o ambos lados)

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

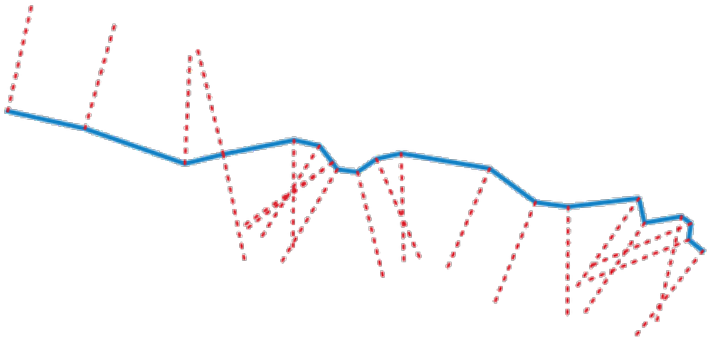


Figura 24.145: Líneas de guión representa el transecto de la capa lineal entrante

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
longitud del transecto	`LENGTH`	[numérico: doble] Preestablecido: 5.0	Longitud en unidades de mapa del transecto
Ángulo en grados desde la línea original a los vértices	`ANGLE`	[numérico: doble] Preestablecido: 90.0	Cambia el ángulo del transecto
Lado a crear el transecto	`SIDE`	[enumeración]	Elegir el lado del transecto. Opciones disponibles son: • 0 – Izquierda • 1 – Derecha • 2 — Ambos
Transecto	`OUTPUT`	[vectorial: lineal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de la línea de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Transecto	`OUTPUT`	[vectorial: lineal]	Capa lineal saliente

Código Python

Algoritmo ID: `native:transect`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Traslado

Mueve las geometrías sin una capa, por compensación con unos desplazamientos predefinidos X e Y.

Los valores Z y M presentes en la geometría también pueden trasladarse.

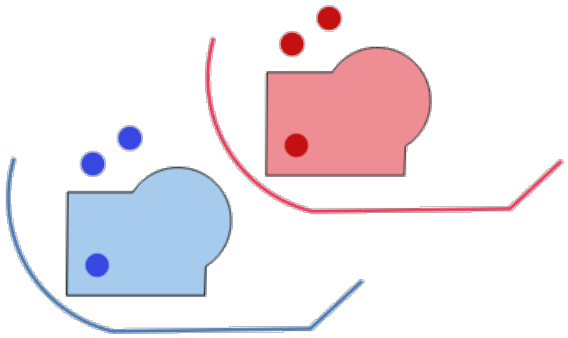


Figura 24.146: Los objetos espaciales en rojo representan la geometría traducida de la capa de entrada (azul).

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Ver también

Colección de objetos trasladados, *Compensado de líneas*, *Rotar*, *Intercmbiar coordenadas X e Y*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Distancia de compensación (eje-x)	`DELTA_X`	[numérico: doble] Predeterminado: 0.0	Desplazamiento a aplicar en el eje X
Distancia de compensación (eje-y)	`DELTA_Y`	[numérico: doble] Predeterminado: 0.0	Desplazamiento a aplicar en el eje Y
Distancia de compensación (eje-Z)	`DELTA_Z`	[numérico: doble] Predeterminado: 0.0	Desplazamiento a aplicar en el eje Z
Distancia de compensación (valores m)	`DELTA_M`	[numérico: doble] Predeterminado: 0.0	Desplazamiento a aplicar en el eje M
Trasladado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Trasladado	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente

Código Python

Algoritmo ID: `native:translategeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Ancura de buffer variable (por valor de M)

Crea búferes de ancho variable a lo largo de las líneas, utilizando el valor M de las geometrías de línea como el diámetro del búfer en cada vértice.

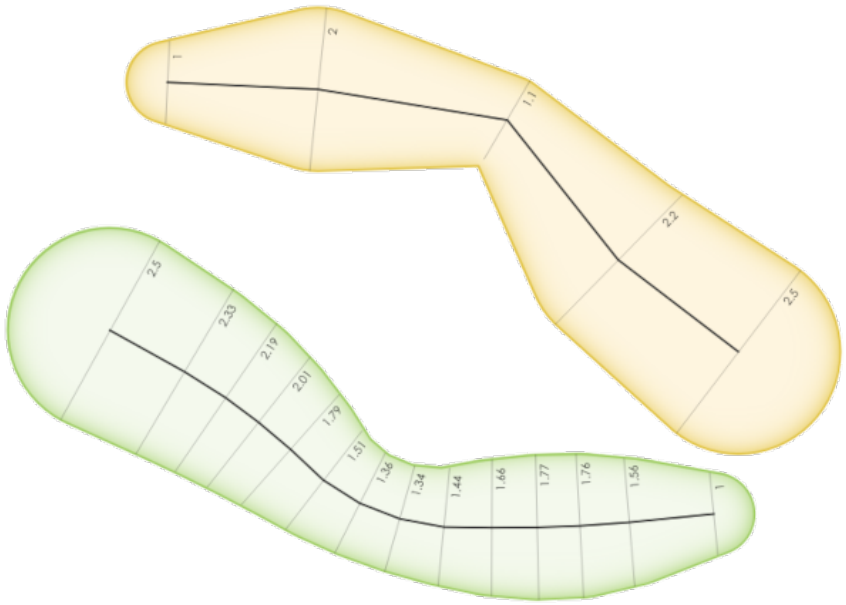


Figura 24.147: Ejemplo de buffer variable

Ver también

Zonas de influencia estrechas, *Buffer*, *Establecer valor M*, *Búfer de distancia variable*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa vectorial entrante de lineas
Segmentos	`SEGMENTS`	[numérico: entero] Preestablecido: 16	Número de segmentos de búfer por cuarto de círculo. Puede ser un valor único (el mismo valor para todas las entidades), o puede tomarse de los datos de las entidades(el valor puede depender de los atributos de la entidad).
Buffered	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (búfer). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Buffered	`OUTPUT`	[vectorial: poligonal]	Capa poligonal de buffer variable

Código Python

Algoritmo ID: `native:bufferbym`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Polígonos Voronoi

Toma una capa de puntos y genera una capa de polígono que contiene los polígonos de Voronoi (también conocidos como polígonos de Thiessen) correspondientes a esos puntos de entrada.

Cualquier ubicación dentro de un polígono Voronoi está más cerca del punto asociado que de cualquier otro punto.

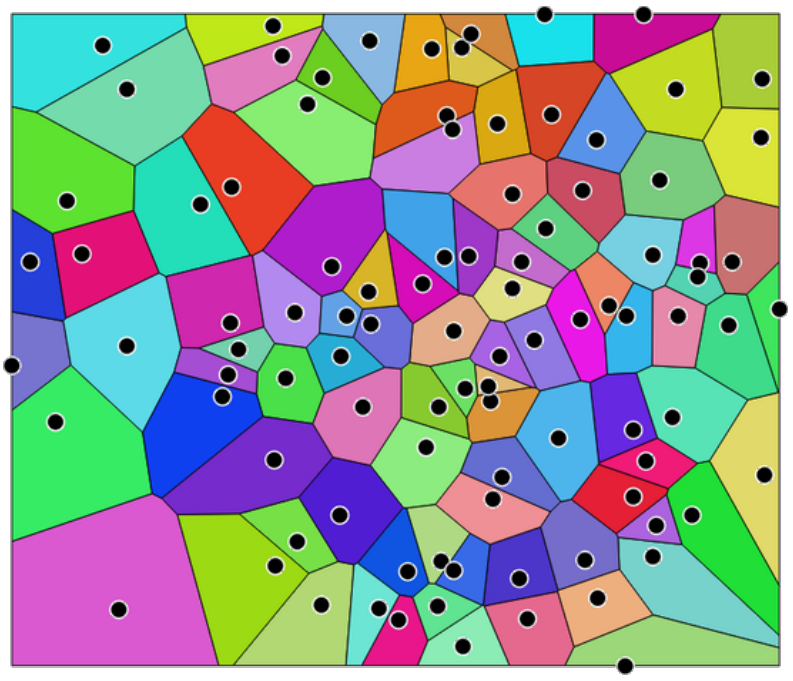


Figura 24.148: Polígonos Voronoi

Menú predeterminado: *Vectorial ► Herramientas de geometría*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Buffer region (% de extensión)	`BUFFER`	[numérico: doble] Predeterminado: 0.0	La extensión de la capa de salida será mucho mayor que la extensión de la capa de entrada.
Tolerancia Opcional	`TOLERANCE`	[numérico: doble] Predeterminado: 0.0	Especifica una tolerancia de ajuste opcional que puede utilizarse para mejorar la robustez de los voronoi.
Copiar atributos de objetos espaciales de entrada	`COPY_ATTRIBUTE`	[booleano] Predeterminado: Verdadero	Especifica si los campos que almacenan el ID de los objetos espaciales de los puntos implicados deben añadirse a la salida. Si es falso, se crea un campo `id` para identificar los polígonos.
Polígonos Voronoi	`OUTPUT`	[vectorial: poligonal] Predeterminado: `[Crear capa temporal]`	Especifique la capa de salida (con los polígonos de Voronoi). *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Polígonos Voronoi	`OUTPUT`	[vectorial: poligonal]	Polígonos de Voronoi de la capa vectorial del punto de entrada

Código Python

ID de Algoritmo: `native:voronoipolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.27 Superposición vectorial

Cortar

Corta una capa vectorial usando las entidades de capa poligonal adicional.

Solo las partes de las entidades en la capa de entrada que se encuentran dentro de los polígonos de la capa de superposición se agregarán a la capa resultante.

> **Advertencia**
>
> **Sólo modificación de la geometría**
>
> Esta operación sólo modifica la geometría de los elementos. Los valores de los atributos de las características **no se modifican**, aunque propiedades como el área o la longitud de las características sí serán modificadas por la operación de superposición. Si tales propiedades se almacenan como atributos, éstos deberán actualizarse manualmente.

Este algoritmo usa índices espaciales en los proveedores, prepara geometrías y aplica una operación de recorte si la geometría no está totalmente contenida en la geometría de la máscara.

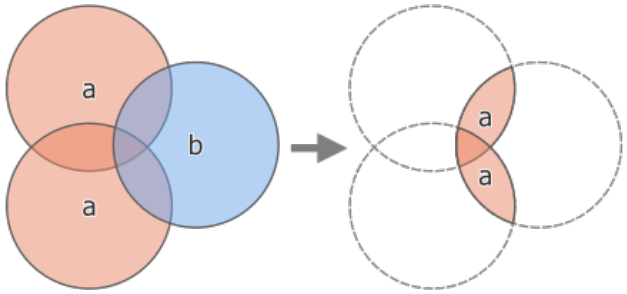


Figura 24.149: Operación de recorte entre una capa de entrada de dos características «a» y una capa superpuesta de una sola característica «b» (izquierda), que da como resultado una nueva capa con las características modificadas de «a» (derecha).

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Menú predeterminado: *Vectorial ► Herramientas de geoprocesamiento*

> **Ver también**
>
> *Intersección*, *Diferencia*

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa contenedora de las entidades a cortar
Capa superpuesta	`OVERLAY`	[vectorial: poligonal]	Capa conteniendo las entidades cortadas

continúe en la próxima página

Tabla 24.760 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Cortadas	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa que contendrá las características de la capa de entrada que se encuentran dentro de la capa de superposición (recorte). *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cortadas	`OUTPUT`	[la misma que la entrada]	Capa contenedora de las entidades de la capa entrante cortada por la capa superpuesta.

Código Python

Algorithm ID: `native:clip`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Diferencia

Extrae entidades de la capa de entrada que no se encuentran dentro de los límites de la capa superpuesta.

Las entidades de la capa de entrada que se superponen parcialmente a las entidades de la capa de superposición se dividen a lo largo del límite de esas entidades y solo se conservan las partes fuera de las entidades de la capa de superposición.

Advertencia

Sólo modificación de la geometría

Esta operación sólo modifica la geometría de los elementos. Los valores de los atributos de las características **no se modifican**, aunque propiedades como el área o la longitud de las características sí serán modificadas por la operación de superposición. Si tales propiedades se almacenan como atributos, éstos deberán actualizarse manualmente.

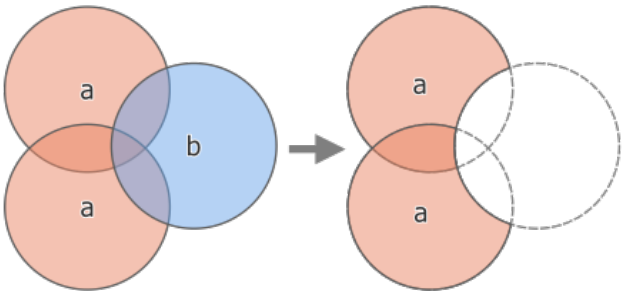


Figura 24.150: Operación de diferencia entre una capa de entrada de dos objetos espaciales «a» y una capa superpuesta de un solo objeto espacial «b» (izquierda), que da como resultado una nueva capa con los objetos espaciales modificados de «a» (derecha).

Permite la modificación in situ de las características de . de objetos espaciales puntos, líneas y polígonos

Menú predeterminado: *Vectorial ► Herramientas de geoprocesamiento*

Ver también

Diferencia (múltiple), *Diferencia simétrica*, *Cortar*

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de la que extraer entidades (partes de)
Capa superpuesta	`OVERLAY`	[vector: geometría]	Capa que contiene las geometrías que se restarán de las geometrías de la capa de entrada. Se espera que tenga al menos tantas dimensiones (punto: 0D, línea: 1D, polígono: 2D, volumen: 3D) como las geometrías de la capa de entrada.
Diferencia	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa que contendrá los (partes de) objetos espaciales de la capa de entrada que no están dentro de la capa de superposición. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Tamaño de la cuadrícula Opcional	`GRID_SIZE`	[numérico: doble] Predeterminado: No establecido	If provided, the input geometries are snapped to a grid of the given size, and the result vertices are computed on that same grid.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Diferencia	`OUTPUT`	[la misma que la entrada]	Capa que contiene (partes de) entidades de la capa de entrada que no se superponen a la capa superpuesta.

Código Python

Algorithm ID: `native:difference`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Diferencia (múltiple)

Extrae características de la capa de entrada que caen completamente fuera o solo se superponen parcialmente las entidades de la(s) capa(s) superpuesta(s).

Por cada capa superpuesta la diferencia se calcula entre el resultado de todas las diferencias de las operaciones anteriores y esta capa superpuesta. Las características de capa de entrada que se superponen parcialmente a caracteristica(s) en la capa superpuesta se dividen a lo largo del límite de esas características y solo se retienen las pociones fuera de las características de la capa superpuesta.

Advertencia

Sólo modificación de la geometría

Esta operación sólo modifica la geometría de los elementos. Los valores de los atributos de las características **no se modifican**, aunque propiedades como el área o la longitud de las características sí serán modificadas por la operación de superposición. Si tales propiedades se almacenan como atributos, éstos deberán actualizarse manualmente.

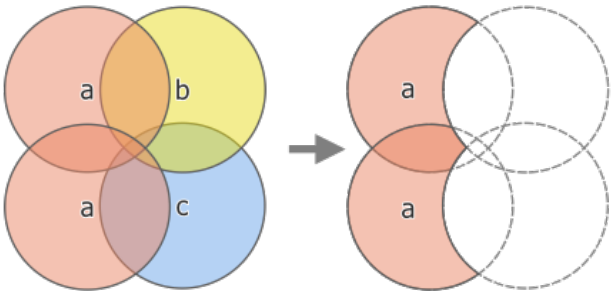


Figura 24.151: Operación de diferencia entre una capa de entrada de dos características “a” y capas superpuestas de una sola característica “b” y “c” (izquierda), lo que da como resultado una nueva capa con las características “a” modificadas (derecha)

Ver también

Diferencia, *Diferencia simétrica*, *Cortar*

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de la que extraer entidades (partes de)
Capas superpuestas	`OVERLAYS`	[vector: geometría] [lista]	Lista de capas que contienen las geometrías que serán restadas de las geometrías de la capa de entrada. Se espera que tengan al menos tantas dimensiones (punto: 0D, línea: 1D, polígono: 2D, volumen: 3D) como las geometrías de capa de entrada.
Diferencia	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa que contendrá (partes de) los objetos espaciales de la capa de entrada que no se solapen con los objetos espaciales de las capas superpuestas. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Diferencia	`OUTPUT`	[la misma que la entrada]	Capa para contener las (partes de) características de la capa de entrada que no superpone características de las capas superpuestas.

Código Python

Algorithm ID: `native:multidifference`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Extraer/cortar por extensión

Crea una nueva capa vectorial que sólo contiene elementos que se encuentran dentro de una extensión especificada. Cualquier elemento que intersecte la extensión se incluirá por defecto.

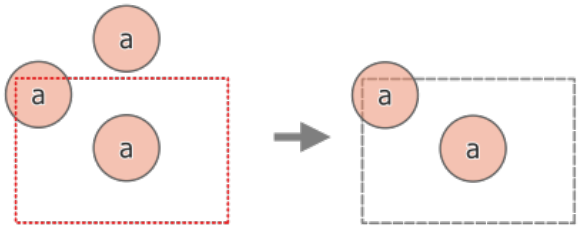


Figura 24.152: Operación de extracción entre una capa de entrada de tres objetos espaciales «a» y una extensión punteada (izquierda) - objetos espaciales resultantes con la extensión punteada como referencia (derecha)

Opcionalmente, las geometrías de los objetos espaciales también pueden recortarse en la medida.

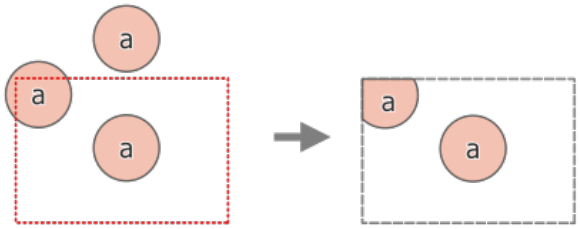


Figura 24.153: Operación de extracción entre una capa de entrada de tres objetos espaciales «a» y una extensión punteada (izquierda) - objetos espaciales resultantes con la extensión punteada como referencia (derecha)

Ver también

Cortar

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de la que extraer entidades (partes de)
Extesión (xmin, xmax, ymin, ymax)	`EXTENT`	[extensión]	Extensión para cortar. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Cortar entidades a extensión	`CLIP`	[booleano] Predeterminado: False	Si está marcada, las geometrías se recortarán en la medida elegida en lugar de tomar toda la geometría como salida. Además, las geometrías de salida se convertirán automáticamente en geometrías múltiples para garantizar tipos de salida uniformes.
Extraido	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa que contendrá los objetos espaciales de la capa de entrada que se encuentran dentro de la extensión del clip. *Una de las siguientes opciones*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extraido	`OUTPUT`	[la misma que la entrada]	Capa que contiene las entidades recortadas.

Código Python

ID del algoritmo: `native:extractbyextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Intersección

Extrae las partes de entidades de la capa de entrada que se superponen a entidades en la capa de superposición.

A las entidades de la capa de intersección se les asignan los atributos de las entidades superpuestas de las capas de entrada y de superposición.

Advertencia

Sólo modificación de la geometría

Esta operación sólo modifica la geometría de los elementos. Los valores de los atributos de las características **no se modifican**, aunque propiedades como el área o la longitud de las características sí serán modificadas por la operación de superposición. Si tales propiedades se almacenan como atributos, éstos deberán actualizarse manualmente.

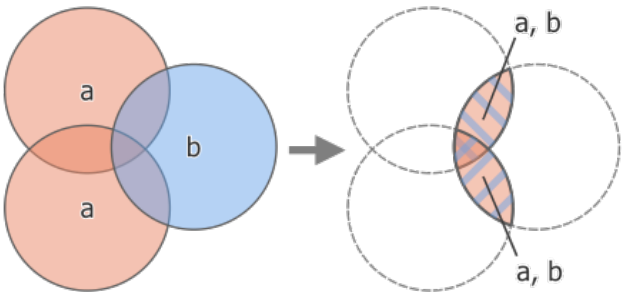


Figura 24.154: Operación de intersección entre una capa de entrada de dos características “a” y una sola capa de superposición de características “b” (izquierda): las áreas superpuestas se convierten en una nueva capa de dos características con los atributos de ambas capas (derecha)

Menú predeterminado: *Vectorial* ► *Herramientas de geoprocesamiento*

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Intersección (múltiple), *Cortar*, *Diferencia*

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de la que extraer entidades (partes de)
Capa superpuesta	`OVERLAY`	[vector: geometría]	Capa que contiene las entidades para comprobar si se superponen. Se espera que la geometría de sus entidades tenga al menos tantas dimensiones (punto: 0D, línea: 1D, polígono: 2D, volumen: 3D) como la capa de entrada.
Campos de entrada a mantener (dejar vacío para mantener todos losa campos) Opcional	`INPUT_FIELDS`	[campo de tabla: cualquiera] [lista] Predeterminado: No establecido	Campo(s) de la capa de entrada para mantener en la salida. Si no se elige ningún campo, se toman todos los campos.
Superponer campos a mantener (dejar en blanco para mantener todos los campos) Opcional	`OVERLAY_FIELDS`	[campo de tabla: cualquiera] [lista] Predeterminado: No establecido	Campo(s) de la capa superpuesta a mantener en la salida. Si no hay campos elegidos se toman todos. A los nombres de campos duplicados se les agregará un sufijo para evitar colisiones.
Intresección	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa que contendrá (las partes de) los objetos espaciales de la capa de entrada que se solapan con uno o más objetos espaciales de la capa de superposición. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Prefijo de campos superpuestos Opcional	`OVERLAY_FIELDS_`	[cadena de texto]	Añade un prefijo para identificar los campos de la capa superpuesta. A los nombres de los campos duplicados se les agregará un sufijo para evitar colisiones.
Tamaño de la cuadrícula Opcional	`GRID_SIZE`	[numérico: doble] Predeterminado: No establecido	If provided, the input geometries are snapped to a grid of the given size, and the result vertices are computed on that same grid.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Intresección	`OUTPUT`	[la misma que la entrada]	Capa que contiene (partes de) entidades de la capa de entrada que se superponen a la capa superpuesta.

Código Python

Algorithm ID: `native:intersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Intersección (múltiple)

Extrae las porciones superpuestas de las características en la entrada y en todas las capas superpuestas.

Las características en la capa de salida tienen asignados los atributos de las características superpuestas tanto en la entrada como las capas superpuestas.

Advertencia

Sólo modificación de la geometría

Esta operación sólo modifica la geometría de los elementos. Los valores de los atributos de las características **no se modifican**, aunque propiedades como el área o la longitud de las características sí serán modificadas por la operación de superposición. Si tales propiedades se almacenan como atributos, éstos deberán actualizarse manualmente.

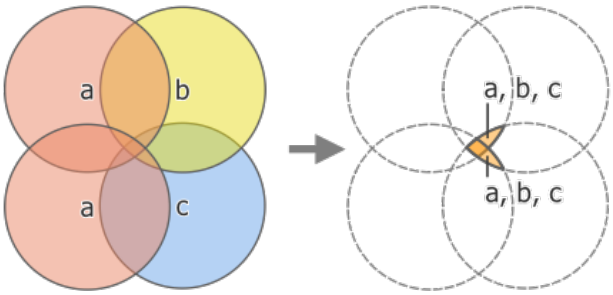


Figura 24.155: Operación de intersección entre una capa de entrada de dos entidades “a” y capas superpuestas de una sola entidad “b” y “c” (izquierda): las áreas superpuestas se convierten en una nueva capa de dos entidades con todos los atributos de las capas (derecha)

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Intersección, *Cortar*, *Diferencia*

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de la que extraer entidades (partes de)
Capas superpuestas	`OVERLAYS`	[vector: geometría] [lista]	Las capas que contienen las características a comprobar para superponer. La geometría de las características se espera que tengan al menos tantas dimensiones (punto: 0D, línea: 1D, polígono: 2D, volumen: 3D) como la capa de entrada.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Prefijo de campos superpuestos Opcional	`OVERLAY_FIELDS_`	[cadena de texto]	Añade un prefijo para identificas las capas superpuestas. A los nombres de campos duplicados se les agrega un sufijo de conteo para evitar colisiones.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Intresección	`OUTPUT`	[la misma que la entrada]	Capa que contiene (partes de) características de la capa de entrada que se superpone a todas las capas superpuestas.

Código Python

Algorithm ID: `native:multiintersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Intersecciones de línea

Crea entidades puntutales donde las líneas de dos capas se intersecan.

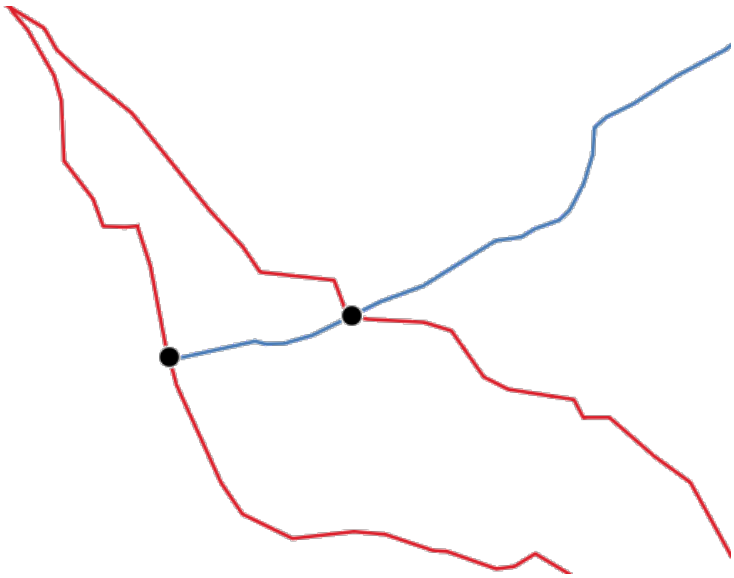


Figura 24.156: Puntos de intersección

Menú predeterminado: *Vectorial ► Herramientas de análisis*

> **Advertencia**
>
> Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	Capa lineal entrante
Capa intersección	`INTERSECT`	[vectorial: lineal]	Capa a usar para encontrar intersecciones de línea.
Campos de entrada a mantener (dejar vacío para mantener todos losa campos) Opcional	`INPUT_FIELDS`	[campo de tabla: cualquiera] [lista] Predeterminado: No establecido	Campo(s) de la capa de entrada para mantener en la salida. Si no se elige ningún campo, se toman todos los campos.
Intersecar campos a mantener (dejar en blanco para mantener todos los campos) Opcional	`INTERSECT_FIEL`	[campo de tabla: cualquiera] [lista] Predeterminado: No establecido	Campo(s) de la capa de intersección a mantener en la salda. Si no se elijen campos se toman todos los campos. A los campos duplicados se les agregará un sufijo de conteo para evitar colisiones.

continúe en la próxima página

Tabla 24.775 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Intresección	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifica la capa que contendrá los puntos de intersección de las líneas de las capas de entrada y superposición. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Prefijo de campos de intersección Opcional	`INTERSECT_FIEL`	[cadena de texto]	Añade un prefijo para identificar los campos en la capa de intersección.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Intersecciones	`OUTPUT`	[vector: punto]	Capa vectorial de puntos de las intersecciones de las líneas, con los atributos de ambas capas.

Código Python

Algorithm ID: `native:lineintersections`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Cortar con líneas

Divide las líneas o polígonos en una capa usando las líneas o anillos de polígonos de otra capa para definir los puntos de ruptura. La intersección entre las geometrías de ambas capas se consideran como puntos de división.

La salida contendrá multigeometrías para entidades de corte.

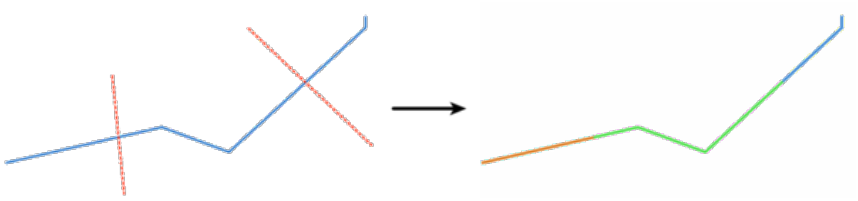


Figura 24.157: Cortar líneas

Permite la modificación in situ de las características de . de objetos espaciales líneas y polígonos

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal, polígonal]	Capa contenedora de las líneas o polígonos a cortar.
Capa de corte	`LINES`	[vectorial: lineal, polígonal]	La capa cuyas líneas o anillos se usan para definir los puntos de ruptura.
Cortar	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa que contendrá los objetos espaciales de línea/polígono divididos (en caso de que sean intersecadas por una línea en la capa dividida) de la capa de entrada. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cortar	`OUTPUT`	[la misma que la entrada]	Capa de vector de salida con líneas de división o polígonos de la capa de entrada.

Código Python

Algorithm ID: `native:splitwithlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Diferencia simétrica

Crea una capa que contiene entidades de las capas de entrada y de superposición, pero con las áreas superpuestas entre las dos capas eliminadas.

La tabla de atributos de la capa de diferencia simétrica contiene atributos y campos de las capas de entrada y superpuesta.

Advertencia

Sólo modificación de la geometría

Esta operación sólo modifica la geometría de los elementos. Los valores de los atributos de las características **no se modifican**, aunque propiedades como el área o la longitud de las características sí serán modificadas por la operación de superposición. Si tales propiedades se almacenan como atributos, éstos deberán actualizarse manualmente.

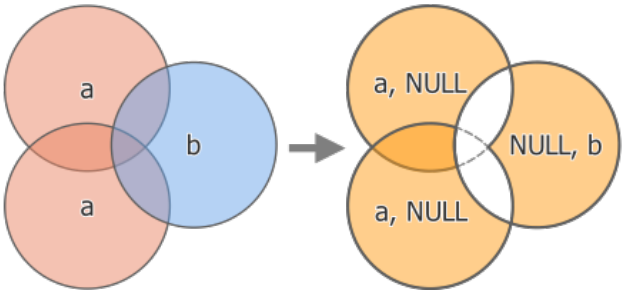


Figura 24.158: La operación de diferencia simétrica entre una capa de entrada de dos características “a” y una única característica de capa superpuesta “b” (izquierda) - resulta una capa de tres características con atributos de ambas capas (derecha)

Menú predeterminado: *Vectorial ► Herramientas de geoprocesamiento*

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Diferencia, *Cortar*, *Intersección*

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Primera capa de la que extraer entidades (partes de)
Capa superpuesta	`OVERLAY`	[vector: geometría]	Segunda capa de la que extraer (partes de) entidades. Idealmente, el tipo de geometría debería ser el mismo que el de la capa de entrada.
Diferencia simétrica	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa que contendrá (las partes de) los objetos espaciales de las capas de entrada y superpuesta que no se solapen con los objetos espaciales de la otra capa. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Prefijo de campos superpuestos Opcional	`OVERLAY_FIELDS_`	[cadena de texto]	Añade un prefijo para identificar los campos de la capa superpuesta. A los nombres de los campos duplicados se les agregará un sufijo para evitar colisiones.
Tamaño de la cuadrícula Opcional	`GRID_SIZE`	[numérico: doble] Predeterminado: No establecido	If provided, the input geometries are snapped to a grid of the given size, and the result vertices are computed on that same grid.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Diferencia simétrica	`OUTPUT`	[la misma que la entrada]	Capa que contiene (partes de) características de cada capa que no se superponen con la otra capa, con los atributos de ambas capas.

Código Python

Algorithm ID: `native:symmetricaldifference`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Unión

Comprueba las superposiciones entre entidades dentro de la capa de entrada y crea entidades independientes para partes superpuestas y no superpuestas. El área de superposición creará tantas entidades superpuestas idénticas como entidades participantes en esa superposición.

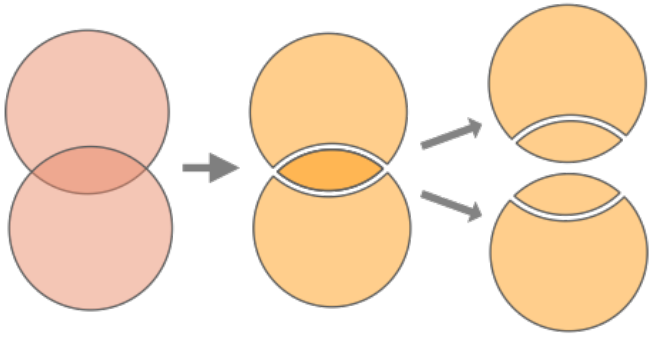


Figura 24.159: Operación de unión con una única capa de entrada con dos características superpuestas (izquierda) - con el resultado de cuatro características (centro) - características movidas por claridad (derecha)

También se puede usar una capa superpuesta, en cuyo caso las características de cada capa se dividen en su superposición con las características de las otras, creando una capa que contiene todas las porciones de tanto las entrada como las capas superpuestas. Las características en la misma capa no se dividen entre sí. La tabla de atributos de la capa de unión se rellena con los valores de los atributos de la respectiva capa original por las características no superpuestas y los atributos de valores de ambas capas para características superpuestas.

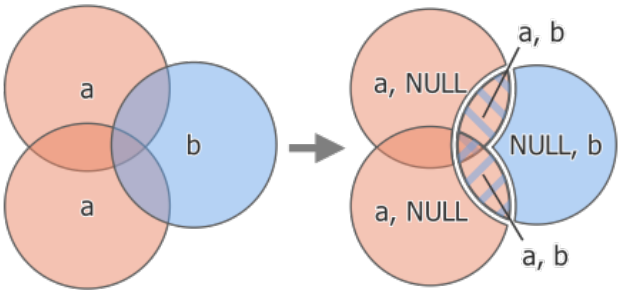


Figura 24.160: La operación de unión entre una capa de entrada de dos características “a” y una capa superpuesta de características única “b” (izquierda) - resulta en una capa de cinco características con los atributos de ambas capas (derecha)

Nota

Con una capa superpuesta, las características de las misma capa no se dividirán entre si. Si desea dividir superposiciones de la misma capa así como de otras capas, primero ejecuta el algoritmo con múltiples capas después corre el algoritmo otra vez con solo la salida anterior.

Menú predeterminado: *Vectorial* ► *Herramientas de geoprocesamiento*

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Unión (múltiple), *Cortar*, *Diferencia*, *Intersección*

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa vectorial entrante a dividir en cualquier intersección.
Capa superpuesta Opcional	`OVERLAY`	[vector: geometría]	Capa que se combinará con la primera. Idealmente, el tipo de geometría debería ser el mismo que el de la capa de entrada.
Unión	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa que contendrá los objetos espaciales (divididos y duplicados) de la capa de entrada y de la capa de superposición. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Prefijo de campos superpuestos Opcional	`OVERLAY_FIELDS_`	[cadena de texto]	Añade un prefijo para identificar los campos de la capa superpuesta. A los nombres de los campos duplicados se les agregará un sufijo para evitar colisiones.
Tamaño de la cuadrícula Opcional	`GRID_SIZE`	[numérico: doble] Predeterminado: No establecido	If provided, the input geometries are snapped to a grid of the given size, and the result vertices are computed on that same grid.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Unión	`OUTPUT`	[la misma que la entrada]	Capa que contiene todas las partes superpuestas y no superpuestas de la capa(s) procesada.

Código Python

Algorithm ID: `native:union`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Unión (múltiple)

Comprueba las superposiciones entre entidades dentro de la capa de entrada y crea entidades independientes para partes superpuestas y no superpuestas. El área de superposición creará tantas entidades superpuestas idénticas como entidades participantes en esa superposición.

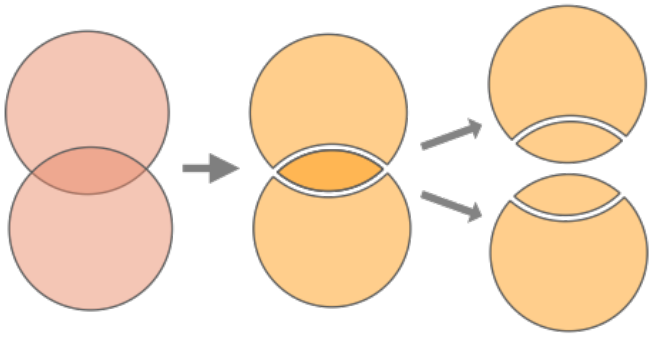


Figura 24.161: Operación de unión con una única capa de entrada con dos características superpuestas (izquierda) - con el resultado de cuatro características (centro) - características movidas por claridad (derecha)

También se puede usar la superposición de múltiples capas, en cuyo caso las características de cada capa se dividen su superposición con características de las demás capas, creando una capa que contiene todas las porciones de la entrada como de las capas superpuestas. Las características de la misma capa no se dividirán entre ellas. La tabla de atributos de la capa Unión se rellena con los valores de capa original respectiva para las características no superpuestas y los valores de los atributos de la capas superpuestas para las características superpuestas.

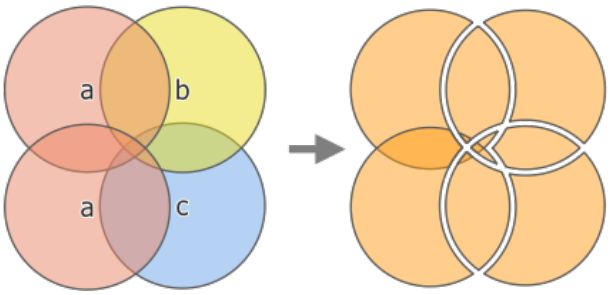


Figura 24.162: La operación de unión entre una capa de entrada de dos características “a” y la capas de característica única superpuesta “b” y “c” (izquierda) - resulta en una capa de once características con los atributos de todas las capas (derecha)

Nota

Con una capa superpuesta, las características de las misma capa no se dividirán entre si. Si desea dividir superposiciones de la misma capa así como de otras capas, primero ejecuta el algoritmo con múltiples capas después corre el algoritmo otra vez con solo la salida anterior.

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Unión, *Cortar*, *Diferencia*, *Intersección*

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa vectorial entrante a dividir en cualquier intersección.
Capas superpuestas Opcional	`OVERLAYS`	[vector: geometría] [lista]	Capas que serán combinadas con la primera. Idealmente el tipo de geometría debería ser el mismo de la capa de entrada.
Unión	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa que contendrá los objetos espaciales (divididos y duplicados) de la capa de entrada y las capas superpuestas. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Prefijo de campos superpuestos Opcional	`OVERLAY_FIELDS_`	[cadena de texto]	Añade un prefijo para identificas las capas superpuestas. A los nombres de campos duplicados se les agrega un sufijo de conteo para evitar colisiones.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Unión	`OUTPUT`	[la misma que la entrada]	Capa que contiene todas las partes superpuestas y no superpuestas de la(s) capa(s) procesada(s), con todos los atributos de la capa.

Código Python

Algorithm ID: `native:multiunion`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.1.28 Selección vectorial

Extraer por atributo

Crea dos capas vectoriales a partir de una capa de entrada: una contendrá solo entidades coincidentes mientras que la segunda contendrá todas las entidades no coincidentes.

Los criterios para agregar entidades a la capa resultante se basan en los valores de un atributo de la capa de entrada.

Ver también

Seleccionar por atributo

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa de la que extraer entidades.
Atributo de selección	`FIELD`	[campo de tabla: cualquiera]	Campo de filtrado de la capa
Operador	`OPERATOR`	[enumeración] Predeterminado: 0	Muchos operadores diferentes están disponibles: • 0 — = • 1 — ≠ • 2 — >> • 3 — >= • 4 — < • 5 — <= • 6 — empieza con • 7 — contiene • 8 — es nulo • 9 — no es nulo • 10 — no contiene
Valor Opcional	`VALUE`	[cadena de texto]	Valor a evaluar
Extraido (atributo)	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida para los objetos espaciales de coincidencia. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.
Extraido (no coincidente)	`FAIL_OUTPUT`	[la misma que la entrada] Preestablecido: `[Omitir salida]`	Especifique la capa de vectores de salida para los objetos espaciale no coincidentes. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extraido (atributo)	`OUTPUT`	[la misma que la entrada]	Capa vectorial con entidades coincidentes de la capa de entrada
Extraido (no coincidente)	`FAIL_OUTPUT`	[la misma que la entrada]	Capa vectorial con entidades no coincidentes de la capa entrante

Código Python

Algorithm ID: `native:extractbyattribute`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extraer por expresión

Crea dos capas vectoriales a partir de una capa de entrada: una contendrá solo entidades coincidentes mientras que la segunda contendrá todas las entidades no coincidentes.

El criterio para añadir entidades a la capa resultante se basa en una expresión de QGIS. Para mas información sobre expresiones ver *Expresiones*.

Ver también

Seleccionar por expresión

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada
Expresión	`EXPRESSION`	[expresión]	Expresión para filtrar la capa vectorial
Entidades coincidentes	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida para los objetos espaciales de coincidencia. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

continúe en la próxima página

Tabla 24.791 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
No coincidentes	`FAIL_OUTPUT`	[la misma que la entrada] Preestablecido: `[Omitir salida]`	Especifique la capa de vectores de salida para los objetos espaciale no coincidentes. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Entidades coincidentes	`OUTPUT`	[la misma que la entrada]	Capa vectorial con entidades coincidentes de la capa de entrada
No coincidentes	`FAIL_OUTPUT`	[la misma que la entrada]	Capa vectorial con entidades no coincidentes de la capa entrante

Código Python

Algorithm ID: `native:extractbyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extraer por localización

Crea una nueva capa vectorial que solo contiene entidades coincidentes de una capa de entrada.

Los criterios para agregar entidades a la capa resultante se basan en la relación espacial entre cada entidad y las entidades en una capa adicional.

Ver también

Seleccionar por ubicacion, *Extraer a distancia*

Explorando las relaciones espaciales

Los predicados geométricos son funciones booleanas que se utilizan para determinar la relación espacial que tiene una característica con otra comparando si sus geometrías comparten una porción de espacio y cómo.

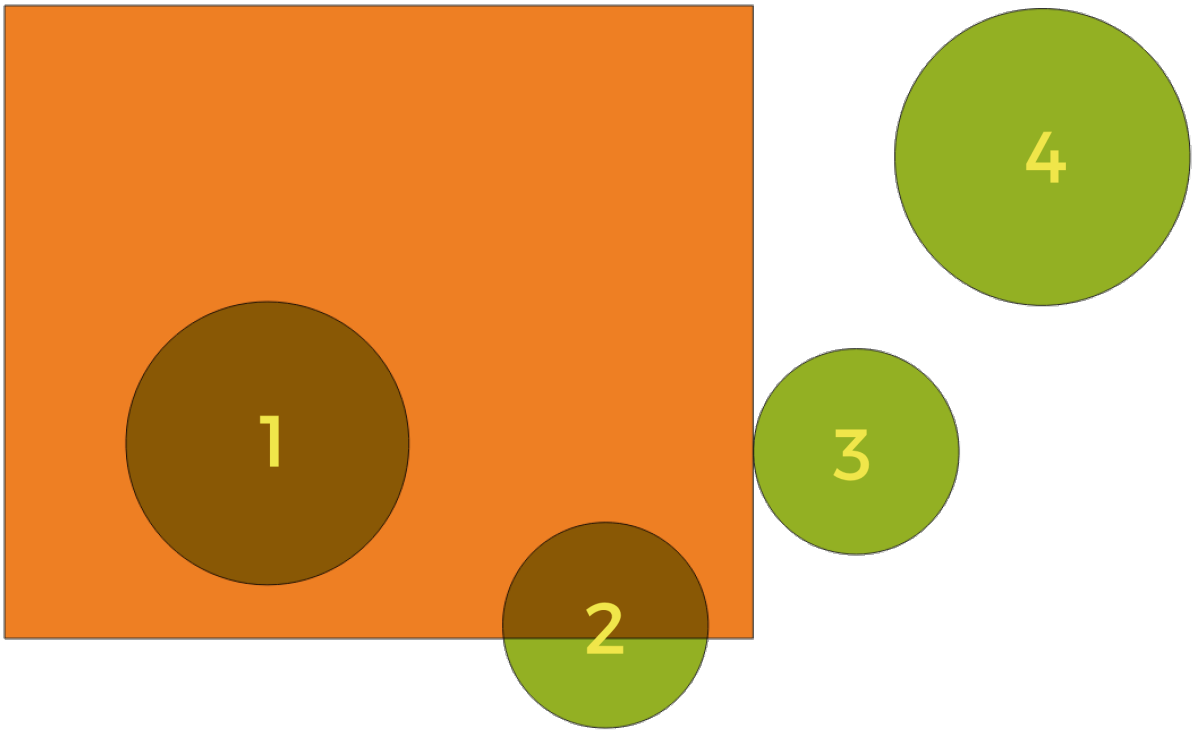


Figura 24.163: Buscando relaciones espaciales entre capas

Usando la figura de arriba, buscamos los círculos verdes comparándolos espacialmente con la característica del rectángulo naranja. Los predicados geométricos disponibles son:

Intersección
: Comprueba si una geometría se cruza con otra. Devuelve 1 (verdadero) si las geometrías se intersecan espacialmente (comparten cualquier parte del espacio, se superponen o se tocan) y 0 en caso contrario. En la imagen de arriba, esto devolverá los círculos 1, 2 y 3.

Contiene
: Devuelve 1 (verdadero) si y solo si ningún punto de b está en el exterior de a, y al menos un punto del interior de b está en el interior de a. En la imagen, no se devuelve ningún círculo, pero el rectángulo sí lo sería si lo buscaras al revés, ya que contiene el círculo 1 por completo. Esto es lo opuesto a *están dentro*.

Dividir
: Devuelve 1 (verdadero) si las geometrías no comparten ninguna parte del espacio (no se superponen, no se tocan). Solo se devuelve el círculo 4.

Igual
: Devuelve 1 (verdadero) si y solo si las geometrías son exactamente iguales. No se devolverán círculos.

Tocar
: Comprueba si una geometría toca otra. Devuelve 1 (verdadero) si las geometrías tienen al menos un punto en común, pero sus interiores no se cruzan. Solo se devuelve el círculo 3.

Superponer
: Comprueba si una geometría se superpone a otra. Devuelve 1 (verdadero) si las geometrías comparten espacio, tienen la misma dimensión, pero no están completamente contenidas entre sí. Solo se devuelve el círculo 2.

Están dentro

Comprueba si una geometría está dentro de otra. Devuelve 1 (verdadero) si la geometría a está completamente dentro de la geometría b. Solo se devuelve el círculo 1.

Cruza

Devuelve 1 (verdadero) si las geometrías proporcionadas tienen algunos puntos interiores en común, pero no todos, y el cruce real tiene una dimensión más baja que la geometría proporcionada más alta. Por ejemplo, una línea que cruza un polígono se cruzará como una línea (verdadero). Dos líneas que se cruzan se cruzarán como un punto (verdadero). Dos polígonos se cruzan como un polígono (falso). En la imagen, no se devolverán círculos.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Extraer entidades de	`INPUT`	[vector: geometría]	Capa de vector de entrada
Donde las entidades (predicado geométrico)	`PREDICATE`	[enumeración] [lista] Predeterminado: [0]	Tipo de relación espacial que el objeto espacial de entrada debe tener con un objeto espacial de intersección para que pueda ser seleccionado. Uno o varios de los siguientes: • 0 — intersecta • 1 — contiene • 2 — dividir • 3 — igual • 4 — toca • 5 — superpone • 6 — esta dentro • 7 — cruza Si se elige más de una condición, al menos una de ellas (operación OR) debe cumplirse para que se extraiga una entidad.
Comparar con las entidades de	`INTERSECT`	[vector: geometría]	Capa vectorial intersección
Extraida (localización)	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida para los objetos espaciales que tengan la relación o relaciones espaciales elegidas con una o más características de la capa de comparación. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extraida (localización)	`OUTPUT`	[la misma que la entrada]	Capa vectorial con entidades de la capa de entrada que tienen las relacion(es) espacial elegida con una o más entidades en la capa de comparación.

Código Python

Algorithm ID: `native:extractbylocation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extraer a distancia

Crea una nueva capa vectorial que solo contiene entidades coincidentes de una capa de entrada. Las entidades se copian dondequiera que estén dentro de la distancia máxima especificada de las entidades en una capa de referencia adicional.

Ver también

Seleccione dentro de la distancia, *Extraer por localización*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Extraer entidades de	`INPUT`	[vector: geometría]	Capa vectorial de entrada para copiar entidades desde
Comparar con las entidades de	`REFERENCE`	[vector: geometría]	Capa vectorial cuya cercanía de entidades se utiliza
Donde están las entidades dentro	`DISTANCE`	[numérico: doble] Predeterminado: 100.0	La distancia máxima alrededor de las entidades de referencia para seleccionar entidades de entrada dentro
Modificar selección actual por	`METHOD`	[enumeración] Predeterminado: 0	Como se debe administrar la selección del algoritmo. Uno de: • 0 — crear nueva selección • 1 — añadir a la selección actual • 2 — seleccionar dentro de la selección actual • 3 — borrar de la selección actual
Extraida (localización)	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa de vectores de salida para los objetos espaciales que estén dentro de la distancia establecida desde los objetos espaciales de referencia. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extraida (localización)	`OUTPUT`	[la misma que la entrada]	Capa vectorial con entidades de la capa de entrada que coinciden con la condición de distancia de las entidades de referencia

Código Python

Algoritmo ID: `native:extractwithindistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Filtrar por tipo de geometría

Filtra entidades por su tipo de geometría. Las entidades entrantes se dirigirán a diferentes salidas en función de si tienen una geometría de punto, línea o polígono.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa a evaluar

Salidas

Etiqueta	Nombre	Tipo	Descripción
Entidades de puntos Opcional	`POINTS`	[vector: punto]	Capa con puntos
Entidades de líneas Opcional	`LINES`	[vectorial: lineal]	Capa con líneas
Entidades de polígonos Opcional	`POLYGONS`	[vectorial: poligonal]	Capa con polígonos
Entidades sin geometría Opcional	`NO_GEOMETRY`	[vector: tabla]	Capa vectorial sin geometría

Código Python

Algoritmo ID: `native:filterbygeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extracción aleatoria

Toma una capa vectorial y genera una nueva que contiene solo un subconjunto de las entidades en la capa de entrada.

El subconjunto se define de forma aleatoria, en función de los ID de entidades, utilizando un porcentaje o valor de recuento para definir el número total de características en el subconjunto.

Advertencia

Este algoritmo elimina las claves primarias o los valores FID existentes y los regenera en las capas de salida.

Ver también

Selección aleatoria

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial fuente de la que seleccionar entidades
Methodo	`METHOD`	[enumeración] Predeterminado: 0	Métodos de selección aleatoria. Uno de: • 0 — Número de entidades seleccionadas • 1 — Porcentaje de entidades seleccionadas
Número/porcentaje de entidades seleccionadas	`NUMBER`	[numérico: entero] Predeterminado: 10	Número o porcentaje de entidades a seleccionar
Extraida (aleatorio)	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida para los objetos espaciales seleccionados aleatoriamente. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extraida (aleatorio)	`OUTPUT`	[la misma que la entrada]	Capa vectorial contenedora de las entidades seleccionadas aleatoriamente a partir de la capa de entrada

Código Python

Algorithm ID: `native:randomextract`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extracto aleatorio dentro de subconjuntos

Toma una capa vectorial y genera una nueva que contiene solo un subconjunto de las entidades en la capa de entrada.

El subconjunto se define de forma aleatoria, en función de los ID de entidades, utilizando un porcentaje o valor de recuento para definir el número total de entidades en el subconjunto. El valor de porcentaje/recuento no se aplica a toda la capa, sino a cada categoría. Las categorías se definen según un atributo determinado.

Ver también

Selección aleatoria con subconjuntos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial de la cuál seleccionar las entidades
Campo ID	`FIELD`	[campo de tabla: cualquiera]	Categoría de la capa vectorial fuente de la que seleccionar las entidades
Methodo	`METHOD`	[enumeración] Predeterminado: 0	Método de selección aleatorio. Uno de: • 0 — Número de entidades seleccionadas • 1 — Porcentaje de entidades seleccionadas
Número/porcentaje de entidades seleccionadas	`NUMBER`	[numérico: entero] Predeterminado: 10	Número o porcentaje de entidades a seleccionar

continúe en la próxima página

Tabla 24.801 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extraido (estratificado aleatorio)	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida para los objetos espaciales seleccionados aleatoriamente. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Extraido (estratificado aleatorio)	`OUTPUT`	[la misma que la entrada]	Capa vectorial contenedora de las entidades seleccionadas aleatoriamente a partir de la capa de entrada

Código Python

Algoritmo ID: `qgis:randomextractwithinsubsets`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Selección aleatoria

Toma una capa vectorial y selecciona un subconjunto de sus entidades. Este algoritmo no genera una nueva capa.

El subconjunto se define de forma aleatoria, en función de los ID de entidades, utilizando un porcentaje o valor de recuento para definir el número total de características en el subconjunto.

menú predeterminado: *Vectorial ► Herramientas de Investigación*

Ver también

Extracción aleatoria

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial para la selección
Methodo	`METHOD`	[enumeración] Predeterminado: 0	Método de selección aleatorio. Uno de: • 0 — Número de entidades seleccionadas • 1 — Porcentaje de entidades seleccionadas
Número/porcentaje de entidades seleccionadas	`NUMBER`	[numérico: entero] Predeterminado: 10	Número o porcentaje de entidades a seleccionar

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[la misma que la entrada]	La capa entrante con las entidades seleccionadas

Código Python

Algoritmo ID: `qgis:randomselection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Selección aleatoria con subconjuntos

Toma una capa vectorial y selecciona un subconjunto de sus entidades. Este algoritmo no genera una nueva capa.

El subconjunto se define de forma aleatoria, en función de los ID de entidades, utilizando un porcentaje o valor de recuento para definir el número total de características en el subconjunto.

El valor porcentaje/recuento no se aplica a toda la capa, sino a cada categoría.

Las categorías se definen de acuerdo con un atributo dado, que también se especifica como un parámetro de entrada para el algoritmo.

No se crean nuevas salidas.

menú predeterminado: *Vectorial ► Herramientas de Investigación*

Ver también

Extracto aleatorio dentro de subconjuntos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial en la que seleccionar entidades
Campo ID	`FIELD`	[campo de tabla: cualquiera]	Categoría de la capa entrante de la que seleccionar entidades
Methodo	`METHOD`	[enumeración] Predeterminado: 0	Método de selección aleatorio. Uno de: • 0 — Número de entidades seleccionadas • 1 — Porcentaje de entidades seleccionadas
Número/porcentaje de entidades seleccionadas	`NUMBER`	[numérico: entero] Predeterminado: 10	Número o porcentaje de entidades a seleccionar

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[la misma que la entrada]	La capa entrante con las entidades seleccionadas

Código Python

Algoritmo ID: `qgis:randomselectionwithinsubsets`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Seleccionar por atributo

Crea una selección en una capa vectorial.

Los criterios para seleccionar entidades se basan en los valores de un atributo de la capa de entrada.

Ver también

Extraer por atributo

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial en la que seleccionar entidades
Atributo de selección	`FIELD`	[campo de tabla: cualquiera]	Campo de filtrado de la capa
Operador	`OPERATOR`	[enumeración] Predeterminado: 0	Muchos operadores diferentes están disponibles: • 0 — = • 1 — ≠ • 2 — >>• 3 — >= • 4 — < • 5 — <= • 6 — empieza con • 7 — contiene • 8 — es nulo • 9 — no es nulo • 10 — no contiene
Valor Opcional	`VALUE`	[cadena de texto]	Valor a evaluar
Modificar selección actual por	`METHOD`	[enumeración] Predeterminado: 0	Como se debe administrar la selección del algoritmo. Uno de: • 0 — crear nueva selección • 1 — añadir a la selección actual • 2 — borrar de la selección actual • 3 — seleccionar dentro de la selección actual

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[la misma que la entrada]	La capa entrante con las entidades seleccionadas

Código Python

Algoritmo ID: `qgis:selectbyattribute`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Seleccionar por expresión

Crea una selección en una capa vectorial.

Los criterios para seleccionar entidades se basan en una expresión QGIS. Para obtener más información sobre las expresiones, consulte *Expresiones*.

Ver también

Extraer por expresión

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada
Expresión	`EXPRESSION`	[expresión]	Expresión para filtrar la capa entrante
Modificar selección actual por	`METHOD`	[enumeración] Predeterminado: 0	Como se debe administrar la selección del algoritmo. Uno de: • 0 — crear nueva selección • 1 — añadir a la selección actual • 2 — borrar de la selección actual • 3 — seleccionar dentro de la selección actual

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[la misma que la entrada]	La capa entrante con las entidades seleccionadas

Código Python

Algoritmo ID: `qgis:selectbyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Seleccionar por ubicacion

Crea una selección en una capa vectorial.

Los criterios para seleccionar entidades se basan en la relación espacial entre cada entidad y las entidades en una capa adicional.

menú predeterminado: *Vectorial* ► *Herramientas de Investigación*

Ver también

Extraer por localización, *Seleccione dentro de la distancia*

Explorando las relaciones espaciales

Los predicados geométricos son funciones booleanas que se utilizan para determinar la relación espacial que tiene una característica con otra comparando si sus geometrías comparten una porción de espacio y cómo.

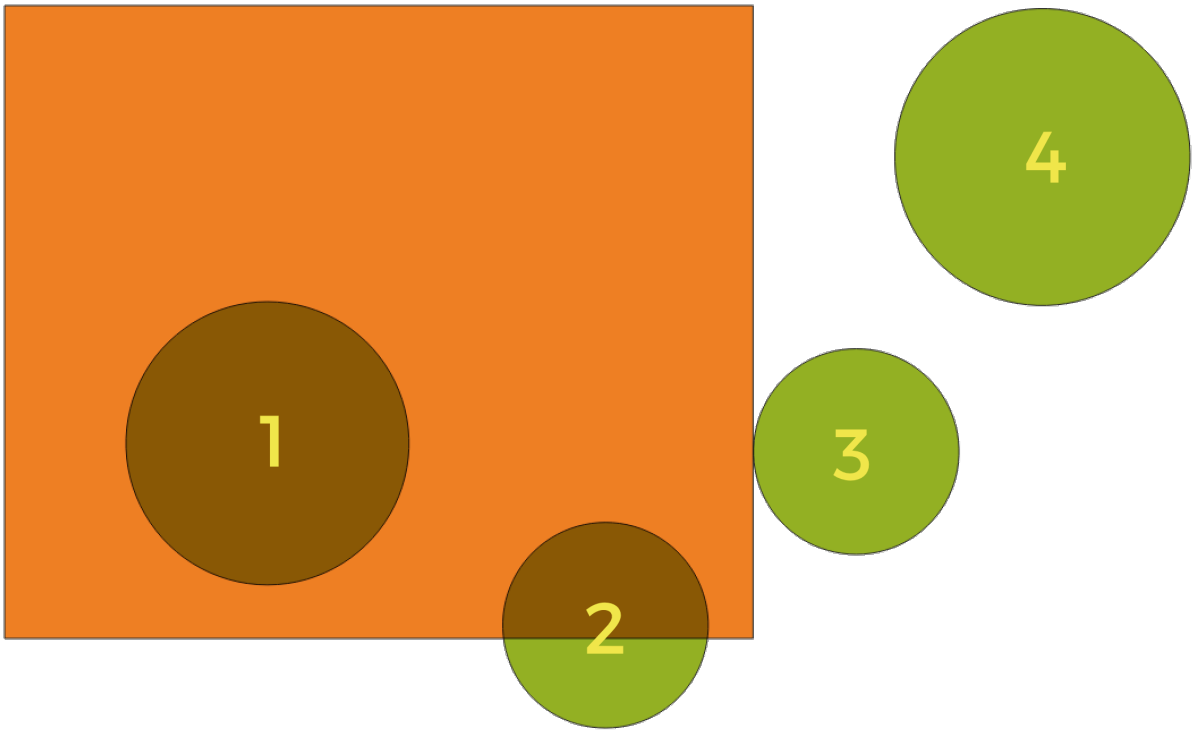


Figura 24.164: Buscando relaciones espaciales entre capas

Usando la figura de arriba, buscamos los círculos verdes comparándolos espacialmente con la característica del rectángulo naranja. Los predicados geométricos disponibles son:

Intersección
: Comprueba si una geometría se cruza con otra. Devuelve 1 (verdadero) si las geometrías se intersecan espacialmente (comparten cualquier parte del espacio, se superponen o se tocan) y 0 en caso contrario. En la imagen de arriba, esto devolverá los círculos 1, 2 y 3.

Contiene
: Devuelve 1 (verdadero) si y solo si ningún punto de b está en el exterior de a, y al menos un punto del interior de b está en el interior de a. En la imagen, no se devuelve ningún círculo, pero el rectángulo sí lo sería si lo buscaras al revés, ya que contiene el círculo 1 por completo. Esto es lo opuesto a *están dentro*.

Dividir
Devuelve 1 (verdadero) si las geometrías no comparten ninguna parte del espacio (no se superponen, no se tocan). Solo se devuelve el círculo 4.

Igual
Devuelve 1 (verdadero) si y solo si las geometrías son exactamente iguales. No se devolverán círculos.

Tocar
Comprueba si una geometría toca otra. Devuelve 1 (verdadero) si las geometrías tienen al menos un punto en común, pero sus interiores no se cruzan. Solo se devuelve el círculo 3.

Superponer
Comprueba si una geometría se superpone a otra. Devuelve 1 (verdadero) si las geometrías comparten espacio, tienen la misma dimensión, pero no están completamente contenidas entre sí. Solo se devuelve el círculo 2.

Están dentro
Comprueba si una geometría está dentro de otra. Devuelve 1 (verdadero) si la geometría a está completamente dentro de la geometría b. Solo se devuelve el círculo 1.

Cruza
Devuelve 1 (verdadero) si las geometrías proporcionadas tienen algunos puntos interiores en común, pero no todos, y el cruce real tiene una dimensión más baja que la geometría proporcionada más alta. Por ejemplo, una línea que cruza un polígono se cruzará como una línea (verdadero). Dos líneas que se cruzan se cruzarán como un punto (verdadero). Dos polígonos se cruzan como un polígono (falso). En la imagen, no se devolverán círculos.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Seleccionar entidades de	`INPUT`	[vector: geometría]	Capa de vector de entrada
Donde las entidades (predicado geométrico)	`PREDICATE`	[enumeración] [lista] Predeterminado: [0]	Tipo de relación espacial que el objeto espacial de entrada debe tener con un objeto espacial de intersección para que pueda ser seleccionado. Uno o varios de los siguientes: • 0 — intersecta • 1 — contiene • 2 — dividir • 3 — igual • 4 — toca • 5 — superpone • 6 — esta dentro • 7 — cruza Si se elige más de una condición, al menos una de ellas (operación OR) debe cumplirse para que se extraiga una entidad.
Comparar con las entidades de	`INTERSECT`	[vector: geometría]	Capa vectorial intersección
Modificar selección actual por	`METHOD`	[enumeración] Predeterminado: 0	Como se debe administrar la selección del algoritmo. Uno de: • 0 — crear nueva selección • 1 — añadir a la selección actual • 2 — seleccionar dentro de la selección actual • 3 — borrar de la selección actual

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[la misma que la entrada]	La capa entrante con las entidades seleccionadas

Código Python

Algorithm ID: `native:selectbylocation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Seleccione dentro de la distancia

crea una selección en una capa vectorial. Las entidades se seleccionan siempre que estén dentro de la distancia máxima especificada de las entidades en una capa de referencia adicional.

Ver también

Extraer a distancia, *Seleccionar por ubicacion*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Seleccionar entidades de	`INPUT`	[vector: geometría]	Capa vectorial de entrada para seleccionar características de
Comparar con las entidades de	`REFERENCE`	[vector: geometría]	Capa vectorial cuya cercanía de entidades se utiliza
Donde están las entidades dentro	`DISTANCE`	[numérico: doble] Predeterminado: 100.0	La distancia máxima alrededor de las entidades de referencia para seleccionar entidades de entrada
Modificar selección actual por	`METHOD`	[enumeración] Predeterminado: 0	Como se debe administrar la selección del algoritmo. Uno de: • 0 — crear nueva selección • 1 — añadir a la selección actual • 2 — seleccionar dentro de la selección actual • 3 — borrar de la selección actual

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[la misma que la entrada]	La capa entrante con las entidades seleccionadas

Código Python

Algoritmo ID: `native:selectwithindistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.29 Tabla vectorial

Añadir campo autoincremental

Agrega un nuevo campo entero a una capa vectorial, con un valor secuencial para cada entidad.

Este campo se puede utilizar como un ID único para las entidades de la capa. El nuevo atributo no se agrega a la capa de entrada, sino que se genera una nueva capa.

Se puede especificar el valor inicial inicial para la serie incremental. Opcionalmente, la serie incremental se puede basar en campos de agrupación y también se puede especificar un orden de clasificación para las entidades.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	La capa vectorial entrante
Nombre de campo	`FIELD_NAME`	[cadena de texto] Predeterminado: “AUTO”	Nombre del campo con valores autoincrementales
Empezar valores en Opcional	`START`	[numérico: entero] Predeterminado: 0	Elegir el número inicial de la cuenta incremental
Valor del módulo Opcional	`MODULUS`	[numérico: entero] Predeterminado: 0	Especificar un valor de módulo opcional reiniciará el conteo para COMENZAR cada vez que el valor del campo alcance el valor del módulo. `0` significa que no hay reinicio.

continúe en la próxima página

Tabla 24.815 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Agrupar valores por Opcional	`GROUP_FIELDS`	[campo de tabla: cualquiera] [lista]	Seleccione el campo(s) de agrupación: en lugar de una sola ejecución de recuento para toda la capa, se procesa un recuento por separado para cada valor devuelto por la combinación de estos campos.
Expresión de ordenar Opcional	`SORT_EXPRESSIO`	[expresión]	Utilice una expresión para ordenar las entidades de la capa de forma global o, si está configurada, según los campos de grupo.
Orden ascendente	`SORT_ASCENDING`	[booleano] Predeterminado: Verdadero	Cuando se establece una `expresión de clasificación`, use esta opción para controlar el orden en el que se asignan valores a las entidades.
Ordenar nulos primero	`SORT_NULLS_FIR`	[booleano] Predeterminado: Falso	Cuando se establece una `expresión de clasificación`, use esta opción para establecer si los valores *Nulos* se cuentan primero o al final.
Incremento	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida con el campo de incremento automático. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Incremento	`OUTPUT`	[la misma que la entrada]	Capa vectorial con campo autoincremental

Código Python

Algoritmo ID: `native:addautoincrementalfield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Añade campo a tabla de atributos

Añade un nuevo campo a la capa vectorial.

El nombre y caracterísitcas del atributo son definidos como parámetros.

El nuevo atributo no se agrega a la capa de entrada, sino que se genera una nueva capa.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	La capa entrante
Nombre de campo	`FIELD_NAME`	[cadena de texto]	Nombre del nuevo campo
Tipo de campo	`FIELD_TYPE`	[enumeración] Predeterminado: 0	Tipo de campo nuevo. Puedes escoger entre: • 0 — Entero (32 bits) • 1 — Decimal (doble) • 2 — Texto (cadena de caracteres) • 3 — Booleano • 4 — Fecha • 5 — Hora • 6 — Fecha y hora • 7 — Objeto binario (BLOB) • 8 — Lista de cadenas • 9 — Lista de enteros • 10 — Lista decimal (doble)
Longitud de campo	`FIELD_LENGTH`	[numérico: entero] Predeterminado: 10	Longitud del campo
Precisión del campo	`FIELD_PRECISIO`	[numérico: entero] Predeterminado: 0	Precisión del campo. Útil con tipo de campo Flotante.
Alias de campo Opcional	`FIELD_ALIAS`	[cadena de texto]	Establece un nombre para usar como alias para el campo. No es compatible con todos los tipos de formato.
Comentario de campo Opcional	`FIELD_COMMENT`	[cadena de texto]	Almacena un comentario que describe el campo. No es compatible con todos los tipos de formato.
Añadido	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa del vector de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Añadido	`OUTPUT`	[la misma que la entrada]	Capa vectorial con nuevo campo añadido

Código Python

*Algoritmo ID**: `native:addfieldtoattributestable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Añadir campo de índice de valor único

Toma una capa vectorial y un atributo y añade un campo numérico.

Los valores de este campo corresponden a los valores del atributo especificado, por lo que las entidades con el mismo valor para el atributo tendrán el mismo valor en el nuevo campo numérico.

Esto crea un equivalente numérico del atributo especificado, que define las mismas clases.

El nuevo atributo no se agrega a la capa de entrada, sino que se genera una nueva capa.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	La capa entrante.
Campo clase	`FIELD`	[campo de tabla: cualquiera]	Las entidades que tienen el mismo valor para este campo obtendrán el mismo índice.
Nombre de campo saliente	`FIELD_NAME`	[cadena de texto] Predeterminado: “NUM_FIELD”	Nombre del nuevo campo que contiene los índices.
Capa con campo de índices	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Capa vectorial con el campo numérico que contiene los índices. *Uno de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

continúe en la próxima página

Tabla 24.819 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Resumen de clase	`SUMMARY_OUTPUT`	[vector: tabla] Preestablecido: `[Omitir salida]`	Especifique la tabla que contendrá el resumen del campo de clase asignado al valor único correspondiente. *Una de*: • Saltar salida • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa con campo de índices	`OUTPUT`	[la misma que la entrada]	Capa de vector con el campo numérico que contiene índices.
Resumen de clase	`SUMMARY_OUTPUT`	[vector: tabla]	Tabla con resumen del campo de clase asignado al valor único correspondiente.

Código Python

Algoritmo ID: `native:adduniquevalueindexfield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Añadir campos X/Y a capa

Agrega campos X e Y (o latitud/longitud) a una capa de puntos. Los campos X/Y se pueden calcular en un SRC diferente a la capa (por ejemplo, creando campos de latitud/longitud para una capa en un SRC proyectado).

Permite la modificación in situ de *características* de objetos espaciales punto

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: punto]	La capa entrante.
Sistema de coordenadas	`CRS`	[src] Predeterminado: «EPSG:4326»	Sistema de referencia de coordenadas que se utilizará para los campos xey generados.

continúe en la próxima página

Tabla 24.821 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Prefijo de campo Opcional	`PREFIX`	[cadena de texto]	Prefijo para agregar a los nuevos nombres de campo para evitar colisiones de nombres con campos en la capa de entrada.
Campos añadidos	`OUTPUT`	[vector: punto] Predeterminado: `[Crear capa temporal]`	Especifica la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Campos añadidos	`OUTPUT`	[vector: punto]	La capa de salida: idéntica a la capa de entrada pero con dos nuevos campos dobles, `x` e `y`.

Código Python

ID del algoritmo: `native:addxyfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Calculadora de campo avanzada de Python

Agrega un nuevo atributo a una capa vectorial, con valores resultantes de aplicar una expresión a cada característica.

La expresión se define como una función de Python.

> **Advertencia**
>
> Este algoritmo es un riesgo potencial para la seguridad si se ejecuta con entradas no verificadas, y puede provocar daños en el sistema o fugas de datos.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada
Nombre de campo resumen	`FIELD_NAME`	[cadena de texto] Predeterminado: “NewField”	Nombre del nuevo campo
Tipo de campo	`FIELD_TYPE`	[enumeración] Predeterminado: 0	Tipo de nuevo campo. Uno de: • 0 — Entero (32 bits) • 1 — Decimal (doble) • 2 — Texto (cadena de caracteres) • 3 — Booleano • 4 — Fecha • 5 — Hora • 6 — Fecha y hora • 7 — Objeto binario (BLOB) • 8 — Lista de cadenas • 9 — Lista de enteros • 10 — Lista decimal (doble)
Longitud de campo	`FIELD_LENGTH`	[numérico: entero] Predeterminado: 10	Longitud del campo
Precisión del campo	`FIELD_PRECISIO`	[numérico: entero] Predeterminado: 3	Precisión del campo. Útil con tipo de campo Flotante.
Expresión global Opcional	`GLOBAL`	[cadena de texto]	El código en la sección de expresión global se ejecutará solo una vez antes de que la calculadora comience a recorrer todas las características de la capa de entrada. Por lo tanto, este es el lugar correcto para importar los módulos necesarios o calcular las variables que se utilizarán en los cálculos posteriores.

Fórmula	`FORMULA`	[cadena de texto]	La fórmula de Python a evaluar. Ejemplo: para calcular el área de una capa de polígono de entrada, puede agregar:
			`value = $geom.area()`
			To access an existing field use its name enclosed in the angle brackets. For example, to uppercase values of the «address» field:
			`value = <address>.upper()`

Calculado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifica la capa vectorial con el nuevo campo calculado. Uno de: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Calculado	`OUTPUT`	[la misma que la entrada]	Capa vectorial con el nuevo campo calculado

Código Python

Algoritmo ID: `qgis:advancedpythonfieldcalculator`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Descartar campo(s)

Toma una capa vectorial y genera una nueva que tiene las mismas entidades pero sin las columnas seleccionadas.

Ver también

Conservar campos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial entrante de la que descartar campo(s)
Campos a descartar	`COLUMN`	[campo de tabla: cualquiera] [lista]	El campo(s) a descartar
Campos restantes	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifique la capa vectorial de salida con los campos restantes. *Uno de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Campos restantes	`OUTPUT`	[la misma que la entrada]	Capa vectorial con los campos restantes

Código Python

Algoritmo ID: `native:deletecolumn`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Explotar campo HStore

Crea una copia de la capa de entrada y agrega un nuevo campo para cada clave única en el campo HStore.

La lista de campos esperados es una lista separada por comas opcional. Si se especifica esta lista, solo se agregan estos campos y se actualiza el campo HStore. De forma predeterminada, se agregan todas las claves únicas.

El HStore de PostgreSQL es un simple almacén clave-valor utilizado en PostgreSQL y GDAL (cuando se lee un archivo OSM con el campo `other_tags`.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Campo HStore	`FIELD`	[campo de tabla: cualquiera]	El campo(s) a descartar
Lista esperada de campos separadas por comas Opcional	`EXPECTED_FIELD`	[cadena de texto] Predeterminado: ""	Lista de campos a extraer separada por comas. El campo HStore se actualizará eliminando estas claves.
Explotada	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifica la capa vectorial saliente. Una de: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Explotada	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente

Código Python

Algoritmo ID: `native:explodehstorefield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Extraer campo binario

Extrae el contenido de un campo binario y lo guarda en archivos individuales. Los nombres de archivo se pueden generar usando valores tomados de un atributo en la tabla fuente o basados en una expresión más compleja.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada que contiene los datos binarios
Campo binario	`FIELD`	[campo de tabla: cualquiera]	Campo contenedor de los datos binarios
Nombre de archivo	`FILENAME`	[expresión]	Campo o texto basado en expresión para nombrar cada archivo saliente
Carpeta destino	`FOLDER`	[carpeta] Predeterminado: `[Guardar en directorio temporal]`	Carpeta en la que se almacenarán los archivos de salida. *Una de*: • Guardar en Directorio Temporal • Guardar en Directorio

Salidas

Etiqueta	Nombre	Tipo	Descripción
Carpeta	`FOLDER`	[carpeta]	La carpeta que contiene los archivos de salida.

Código Python

Algoritmo ID: `native:extractbinary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Calculadora de campo

Abre la calculadora de campos (ver *Expresiones*). Puede utilizar todas las expresiones y funciones admitidas.

Se crea una nueva capa con el resultado de la expresión.

La calculadora de campo es muy útil cuando se usa en *El diseñador de modelos*.

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	La capa en la que calcular
Nombre de campo saliente	`FIELD_NAME`	[cadena de texto]	El nombre del campo para los resultados
Tipo de campo de salida	`FIELD_TYPE`	[enumeración] Predeterminado: 0	El tipo de campo. Uno de: • 0 — Decimal (doble) • 1 — Entero (32 bits) • 2 — Texto (cadena de caracteres) • 3 — Fecha • 4 — Hora • 5 — Fecha y hora • 6 — Booleano • 7 — Objeto binario (BLOB) • 8 — Lista de cadenas • 9 — Lista de enteros • 10 — Lista decimal (doble)
Anchura de campo saliente	`FIELD_LENGTH`	[numérico: entero] Predeterminado: 0	La longitud del campo resultante (mínimo 0)
Precisión del campo	`FIELD_PRECISIO`	[numérico: entero] Predeterminado: 0	La precisión del campo resultante (mínimo 0, máximo 15)
Crear nuevo campo	`NEW_FIELD`	[booleano] Predeterminado: Verdadero	¿El campo de resultado debe ser un campo nuevo?
Fórmula	`FORMULA`	[expresión]	La fórmula a emplear para calcular el resultado

continúe en la próxima página

Tabla 24.831 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Calculado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especificación de la capa saliente. • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Calculado	`OUTPUT`	[la misma que la entrada]	Capa saliente con los valores de campo calculados

Código Python

Algoritmo ID: `native:fieldcalculator`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Refactorizar campos

Permite editar la estructura de la tabla de atributos de una capa vectorial.

Los campos se pueden modificar en su tipo y nombre, utilizando un mapeo de campos.

La capa original no se modifica. Se genera una nueva capa, que contiene una tabla de atributos modificada, de acuerdo con el mapeo de campos proporcionado.

Nota

Cuando se utiliza una capa de plantilla con *constraints* en los campos, la información se muestra en el widget con un fondo de color y una información sobre herramientas. Trate esta información como una sugerencia durante la configuración. No se agregarán restricciones en una capa de salida ni el algoritmo las verificará ni las aplicará.

El algoritmo de campos Refactor permite:

- Cambiar nombres y tipos de campo
- Añadir y borrar campos
- Reordenar campos
- Calcular nuevos campos basados en expresiones

- Cargar lista de campo de otra capa

Nota

Este algoritmo utiliza mediciones basadas en el elipsoide y respeta la configuración actual del elipsoide. <measurements_ellipsoid>`.

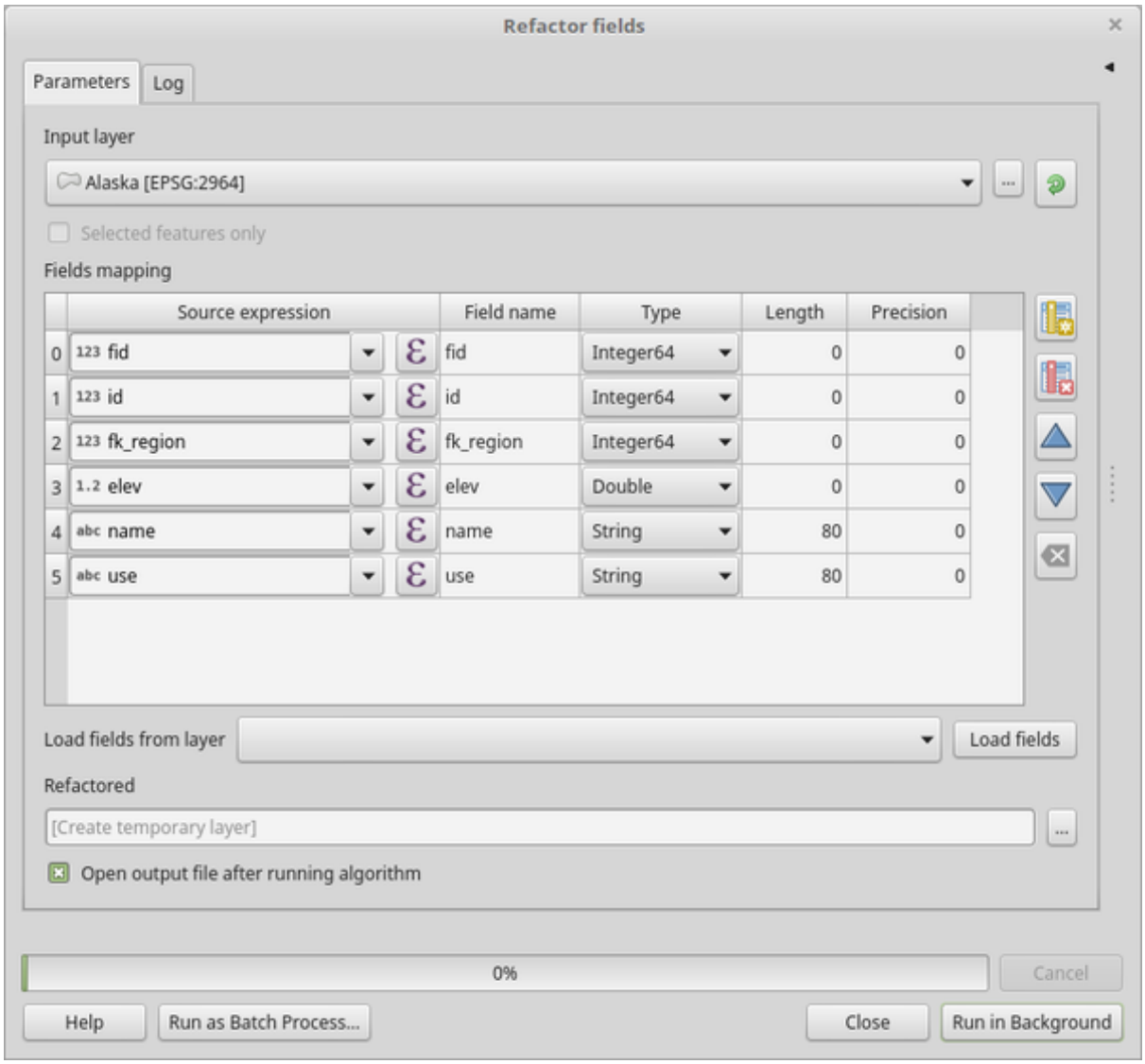


Figura 24.165: Diálogo refactorizar campos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	La capa a modificar

continúe en la próxima página

Tabla 24.833 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Mapeo de campos	`FIELDS_MAPPING`	[lista]	Lista de campos de salida con sus definiciones. La tabla incrustada enumera todos los campos de la capa de origen y le permite editarlos: • Click en  para crear un nuevo campo. • Click en  para borrar un campo. • Usar  para cambiar el orden de los campos seleccionados. • Click en  para restablecer la vista predeterminada. • Haga clic en |invertirSelección| para invertir la selección en la lista de campos. Para cada uno de los campos que desea reutilizar, debe completar las siguientes opciones: ***Expresión de fuente*** **(`expression`) [expression]** Campo o expresión de la capa de entrada. ***Nombre*** **(`nombre`) [cadena]** Nombre del campo en la capa de salida. Por defecto, se mantiene el nombre del campo de entrada. ***Tipo*** **(`type`) [enumeration]** Tipo de datos del campo de salida. Los tipos disponibles pueden no ser compatibles con el proveedor de la capa de salida. Uno de: **Atención** Para ciertos tipos de campo, por ejemplo las listas, un parámetro adicional `sub_type` ayuda a refinar el tipo específico de los datos. Se establece automáticamente en la interfaz gráfica de usuario, pero puede ser necesario si ejecuta el algoritmo en Python o desde la línea de comandos. • 1 — Booleano • 2 — Entero (32 bits) • 4 — Entero (64bit) • 6 — Decimal (doble) • 9 — Lista de números enteros • 9 — Lista de números enteros (64 bits) • 9 — Lista decimal (doble) • 10 — Texto (cadena) • 11 — Lista de cadenas • 12 — Objeto binario (BLOB) • 14 — Fecha • 15 — Tiempo • 16 — Fecha y hora ***Subtipo*** **(`sub_tipo`) [enumeración]** Para ciertos tipos de campos, por ejemplo listas, este parámetro ayuda

Tabla 24.833 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Refactorizado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especificación de la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Refactorizado	`OUTPUT`	[la misma que la entrada]	Capa saliente con campos refactorizados

Código Python

Algoritmo ID: `native:refactorfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Renombrar campo

Renombrar un campo existente en una capa vectorial.

La capa original no se modifica. Una nueva capa es generada donde la tabla de atributos contiene el campo renombrado.

Ver también

Refactorizar campos

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	La capa vectorial entrante
Campo a renombrar	`FIELD`	[campo de tabla: cualquiera]	Campo a ser alterado
Nombre de campo nuevo	`NEW_NAME`	[cadena de texto]	El nombre de campo nuevo
Renombrado	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especificación de la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Renombrado	`OUTPUT`	[la misma que la entrada]	Capa saliente con el campo renombrado

Código Python

Algorithm ID: `native:renametablefield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Conservar campos

Toma una capa vectorial y genera una nueva que conserva solo los campos seleccionados. Todos los demás campos se eliminarán.

Ver también

Descartar campo(s)

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	La capa vectorial entrante
Campos a conservar	`FIELDS`	[campo de tabla: cualquiera] [lista]	Lista de campos a mantener en la capa
Campos retenidos	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especificación de la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Campos retenidos	`OUTPUT`	[la misma que la entrada]	Capa de salida con los campos retenidos

Código Python

Algoritmo ID: `native:retainfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Texto a flotante

Modifica el tipo de un atributo dado en una capa vectorial, convirtiendo un atributo de texto que contiene cadenas numéricas en un atributo numérico (p. Ej. “1” to `1.0`).

El algoritmo crea una nueva capa vectorial para que la fuente no se modifique.

Si la conversión no es posible, la columna seleccionada tendrá valores `NULL`.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	La capa vectorial entrante
Atributo de texto a convertir a flotante	`FIELD`	[campo de tabla: cadena]	El campo de cadena para la capa de entrada que se va a convertir en un campo flotante.
Flotante desde texto	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Crear capa temporal]`	Especifica la capa de salida. *Una de*: • Crear capa temporal (`TEMPORARY_OUTPUT`) • Guardar en archivo… • Guardar en Geopackage… • Guardar en tabla de base de datos… • Agregar a la capa… El fichero codificado también puede ser cambiado aquí.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Flotante desde texto	`OUTPUT`	[la misma que la entrada]	Capa vectorial saliente con el campo de cadena convertido en campo de flotante

Código Python

Algoritmo ID: `qgis:texttofloat`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.1.30 Teselas Vectoriales

Descargar mosaicos vectoriales

Descarga mosaicos vectoriales de una capa de mosaicos vectoriales de entrada y los guarda en un archivo de mosaicos vectoriales local.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[teselas vectoriales]	Una capa de mosaicos vectoriales de la que extraer algunos mosaicos
Extensión	`EXTENT`	[extensión]	Especifica la extensión espacial del área descargada. Se extenderá internamente a un múltiplo del tamaño del mosaico. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Nivel máximo de zoom para descargar	`MAX_ZOOM`	[numérico: entero] Predeterminado: 10	Define hasta qué punto ampliar y obtener datos de los mosaicos.
Límite de mosaico	`TILE_LIMIT`	[numérico: entero] Predeterminado: 100	Número máximo de mosaicos a descargar, teniendo en cuenta los niveles de zoom y la extensión.
Salida	`OUTPUT`	[teselas vectoriales] Predeterminado: [Guardar en archivo temporal]	Especificación del archivo de teselas vectoriales de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Salida	`OUTPUT`	[teselas vectoriales]	Un archivo local de mosaicos vectoriales que almacena los mosaicos descargados.

Código Python

ID algoritmo: `native:downloadvectortiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Escribir teselas vectoriales (MBTiles)

Exporta una o más capas vectoriales a teselas vectoriales, un formato de datos optimizado para la representación rápida de mapas y un tamaño de datos pequeño.

MBTiles es una especificación para almacenar datos de mapas teselados en bases de datos SQLite para uso inmediato y para transferencia. Los archivos MBTiles se conocen como conjuntos de teselas.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[vectorial: cualquiera] [lista]	Una lista de capas a combinar para generar teselas vectoriales
Nivel de zoom mínimo	`MIN_ZOOM`	[numérico: entero] Predeterminado: 0	El menor nivel del zoom para el que el conjunto de teselas suministra datos. Establecer entre 0 y 24.
Nivel de zoom máximo	`MAX_ZOOM`	[numérico: entero] Predeterminado: 3	El mayor nivel del zoom para el que el conjunto de teselas suministra datos. Establecer entre 0 y 24.
Extensión Opcional	`EXTENT`	[extensión] Predeterminado: No establecido	La extensión máxima del área del mapa representada. Los límites deben definir una área cubierta por todos los niveles de zoom.
Metadatos: Nombre Opcional	`META_NAME`	[cadena de texto]	Nombre del conjunto de teselas
Metadatos: Descripción Opcional	`META_DESCRIPTI`	[cadena de texto]	Una descripción del contenido del conjunto de teselas
Metadatos: Atribución Opcional	`META_ATTRIBUTI`	[cadena de texto]	Un texto de atribución, que explique las fuentes de datos y/o el estilo del mapa.

continúe en la próxima página

Tabla 24.843 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Metadatos: Versión Opcional	`META_VERSION`	[cadena de texto]	La versión del conjunto de teselas. Se refiere a la revisión del conjunto de teselas en sí, no a la especificación de MBTiles.
Metadatos: Tipo Opcional	`META_TYPE`	[cadena de texto]	Tipo del conjunto de teselas. Los valores posibles son `superposición` o `capa base`.
Metadatos: Centro Opcional	`META_CENTER`	[cadena de texto]	El centro (texto con números separados por comas: la longitud, latitud, y nivel de zoom) de la vista predeterminada del mapa. Ejemplo: `-122.1906,37.7599,11`
MBTiles destino	`OUTPUT`	[teselas vectoriales] Predeterminado: [Guardar en archivo temporal]	Especificación del archivo MBTiles de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
MBTiles destino	`OUTPUT`	[archivo]	Archivo de salida de teselas vectoriales **:archivo:`.mbtiles`**.

Código Python

ID Algoritmo: `nativo:escribirteselasvectoriales_mbtiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Escribir teselas vectoriales (XYZ)

Exporta una o más capas vectoriales a teselas vectoriales, un formato de datos optimizado para la representación rápida de mapas y un tamaño de datos pequeño.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Plantilla de archivo	`XYZ_TEMPLATE`	[cadena de texto] Predeterminado: "{z}/{x}/{y}.pbf"	Plantilla para generar la url de las teselas vectoriales
Capas de entrada	`INPUT`	[vectorial: cualquiera] [lista]	Una lista de capas a combinar para generar teselas vectoriales

continúe en la próxima página

Tabla 24.845 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Nivel de zoom mínimo	`MIN_ZOOM`	[numérico: entero] Predeterminado: 0	El menor nivel del zoom para el que el conjunto de teselas suministra datos. Establecer entre 0 y 24.
Nivel de zoom máximo	`MAX_ZOOM`	[numérico: entero] Predeterminado: 3	El mayor nivel del zoom para el que el conjunto de teselas suministra datos. Establecer entre 0 y 24.
Extensión Opcional	`EXTENT`	[extensión] Predeterminado: No establecido	La extensión máxima del área del mapa representada. Los límites deben definir una área cubierta por todos los niveles de zoom.
Directorio de salida	`OUTPUT_DIRECTO`	[carpeta] Predeterminado: [Guardar en carpeta temporal]	Especificación de la carpeta de teselas vectoriales de salida. *Una de*: • Guardar en Directorio Temporal • Guardar en Directorio

Salidas

Etiqueta	Nombre	Tipo	Descripción
Directorio de salida	`OUTPUT_DIRECTO`	[carpeta]	Una carpeta conteniendo diferentes subconjuntos de los archivos de teselas vectoriales (`.pbf`) almacenados en subcarpetas correspondientes a los niveles de zoom.

Código Python

ID Algoritmo: `nativo:escribirteselasvectoriales_xyz`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.2 Proveedor de algoritmos GDAL

GDAL (Geospatial Data Abstraction Library) es una biblioteca traductora de formatos de datos geoespaciales ráster y vectoriales. Los algoritmos del marco de procesamiento se derivan de los programas GDAL raster y GDAL vector.

24.2.1 Análisis raster

Aspecto

Genera un mapa de aspecto a partir de cualquier ráster de elevación compatible con GDAL. El aspecto es la dirección de la brújula a la que se enfrenta una pendiente. Los píxeles tendrán un valor de 0-360° medido en grados desde el norte que indica el acimut. En el hemisferio norte, el lado norte de las laderas suele estar sombreado (acimut pequeño de 0°-90°), mientras que el lado sur recibe más radiación solar (acimut más alto de 180°-270°).

Este algoritmo se deriva de la utilidad GDAL DEM.

Menú predeterminado: *Raster* ► *Análisis*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de elevación de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	El número de la banda a usar como elevación
Devuelve el ángulo trigonométrico en lugar del azimut	`TRIG_ANGLE`	[booleano] Predeterminado: Falso	Activar el águlo trigonométrico resulta dá como resultado diferentes categorías: 0° (Este), 90° (Norte), 180° (Oeste), 270° (Sur).
Devuelve 0 para plano en lugar de -9999	`ZERO_FLAT`	[booleano] Predeterminado: Falso	Activando esta opción insertará el valor 0 para el valor -9999 en áreas llanas.
Calcular bordes	`COMPUTE_EDGES`	[booleano] Predeterminado: Falso	Genera bordes a partir del ráster de elevación
Usar la fórmula Zevenbergen&Thorn en lugar de la de Horn's	`ZEVENBERGEN`	[booleano] Predeterminado: Falso	Activa la fórmula Zevenbergen&Thorne para paisajes suavizados
Aspecto	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Capa raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

continúe en la próxima página

Tabla 24.848 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Parámetros adicionales de la línea de comandos Opcional	EXTRA	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Aspecto	OUTPUT	[ráster]	Ráster saliente con valores de ángulo en grados

Código Python

Algoritmo ID: `gdal:aspect`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Color de relieve

Genera un mapa de relieve de color a partir de cualquier ráster de elevación compatible con GDAL. Los relieves de color se pueden utilizar especialmente para representar elevaciones. El algoritmo genera un ráster de 4 bandas con valores calculados a partir de la elevación y un archivo de configuración de color basado en texto. Por defecto, los colores entre los valores de elevación dados se mezclan suavemente y el resultado es un ráster de elevación coloreado agradable.

Este algoritmo se deriva de la utilidad GDAL DEM.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	INPUT	[ráster]	Capa ráster de elevación de entrada
Número de banda	BAND	[banda ráster] Predeterminado: 1	El número de la banda a usar como elevación
Calcular bordes	COMPUTE_EDGES	[booleano] Predeterminado: Falso	Genera bordes a partir del ráster de elevación
Archivo de configuración de color	COLOR_TABLE	[archivo]	Un archivo de configuración de color basado en texto

continúe en la próxima página

Tabla 24.850 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Modo coincidencia	`MATCH_MODE`	[enumeración] Preestablecido: 2	Una de: • 0 — Utilice una combinación de colores estricta • 1 — Utilizar cuádruples RGBA más cercanos • 2 — Usar colores suavemente mezclados
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL
Relieve de color	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Capa raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Relieve de color	`OUTPUT`	[ráster]	Un ráster saliente de 4 bandas

Código Python

Algoritmo ID: `gdal:colorrelief`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Rellenar NoData

Rellena regiones ráster con valores NoData mediante interpolación desde los bordes. Los valores de las regiones NoData se calculan a partir de los valores de los píxeles circundantes utilizando una ponderación de distancia inversa. Después de la interpolación se produce un suavizado de los resultados. La entrada puede ser cualquier capa ráster compatible con GDAL. Este algoritmo suele ser adecuado para interpolar regiones que faltan en rásters que varían de forma bastante continua (como modelos de elevación, por ejemplo). También es adecuado para rellenar pequeños agujeros y grietas en imágenes que varían de forma más irregular (como las aerofotografías). No suele ser tan bueno para interpolar una trama a partir de datos de puntos dispersos.

Este algoritmo se deriva de la utilidad GDAL fillnodata.

Menú predeterminado: *Raster* ► *Análisis*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	La banda sobre la que operar. Los valores NoData deben representarse con el valor 0.
Distancia máxima (en píxeles) para buscar valores para interpolar	`DISTANCE`	[numérico: entero] Predeterminado: 10	El número de pixels a buscar en todas las direcciones para encontrar valores con los que interpolar
Número de iteraciones de suavizado a ejecutar tras la interpolación	`ITERATIONS`	[numérico: entero] Predeterminado: 0	El número de pases de filtro 3x3 a ejecutar (0 o más) para suavizar los resultados de la interpolación.
Mascara de validez	`MASK_LAYER`	[ráster]	Una capa ráster que define las áreas a rellenar.
Rellenado	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la capa raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

continúe en la próxima página

Tabla 24.853 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Rellenado	`OUTPUT`	[ráster]	Ráster saliente

Código Python

Algoritmo ID: `gdal:fillnodata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cuadrícula (datos métricos)

Calcula algunas métricas de datos utilizando la ventana especificada y la geometría de la cuadrícula de salida.

Este algoritmo se deriva de la utilidad GDAL grid.

Menú predeterminado: *Raster* ► *Análisis*

Ver también

Tutorial de grid GDAL

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de puntos	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada

continúe en la próxima página

Tabla 24.855 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Datos métricos a usar	`METRIC`	[enumeración] Predeterminado: 0	Una de: • 0 — Mínimo, valor mínimo encontrado en elipse de búsqueda de nodo de cuadrícula • 1 — Máximo, valor máximo encontrado en la elipse de búsqueda del nodo de la cuadrícula • 2 — Rango, una diferencia entre los valores mínimo y máximo encontrados en la elipse de búsqueda del nodo de la cuadrícula • 3 — Recuento, una cantidad de puntos de datos encontrados en la elipse de búsqueda de nodos de la cuadrícula • 4 — Distancia promedio, una distancia promedio entre el nodo de la cuadrícula (centro de la elipse de búsqueda) y todos los puntos de datos encontrados en la elipse de búsqueda del nodo de la cuadrícula • 5 — Distancia promedio entre puntos, una distancia promedio entre los puntos de datos encontrados en la elipse de búsqueda de nodos de cuadrícula. Se calcula la distancia entre cada par de puntos dentro de la elipse y el promedio de todas las distancias se establece como un valor de nodo de cuadrícula
El primer radio de búsqueda de elipse	`RADIUS_1`	[numérico: doble] Predeterminado: 0.0	El primer radio (eje X si el ángulo de rotación es 0) de la elipse de búsqueda
El segundo radio de la elipse de búsqueda	`RADIUS_2`	[numérico: doble] Predeterminado: 0.0	El segundo radio (eje Y si el ángulo de rotación es 0) de la elipse de búsqueda
Ángulo de rotación en grados de la elipse de búsqueda (sentido antihorario)	`ANGLE`	[numérico: doble] Predeterminado: 0.0	Ángulo de rotación de elipse en grados. Elipse rotada en sentido antihorario
Mínimo número de puntos de datos a usar	`MIN_POINTS`	[numérico: doble] Predeterminado: 0.0	Número mínimo de puntos de datos a promediar. Si se encuentran menos puntos, el nodo de la cuadrícula se considerará vacío y se rellenará con el marcador NoData.
NoData	`NODATA`	[numérico: doble] Predeterminado: 0.0	Marcador NoData para rellenar puntos vacíos
Interpolación(métri de datos)	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida con valores interpolados. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor Z de campo Opcional	`Z_FIELD`	[campo de tabla: numérico]	Campo para la interpolación
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Interpolación(métri de datos)	`OUTPUT`	[ráster]	Ráster de salida con valores interpolados

Código Python

Algoritmo ID: `gdal:griddatametrics`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cuadrícula (IDW con búsqueda de vecino mas próximo)

Calcula la distancia inversa a una cuadrícula de energía combinada con el método del vecino más cercano. Ideal cuando se requiere un número máximo de puntos de datos para usar.

Este algoritmo se deriva de la utilidad GDAL grid.

Ver también

Tutorial de grid GDAL

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de puntos	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Poder de ponderación	`POWER`	[numérico: doble] Preestablecido: 2.0	Poder de ponderación
Suavizado	`SMOOTHING`	[numérico: doble] Predeterminado: 0.0	Parámetro de suavizado
El radio del círculo de búsqueda	`RADIUS`	[numérico: doble] Predeterminado: 1.0	El radio de la búsqueda circular
Número máximo de datos puntuales a usar	`MAX_POINTS`	[numérico: entero] Predeterminado: 12	No buscar para mas puntos que este número.
Mínimo número de puntos de datos a usar	`MIN_POINTS`	[numérico: entero] Predeterminado: 0	Número mínimo de puntos de datos a promediar. Si se encuentran menos puntos, el nodo de la cuadrícula se considerará vacío y se rellenará con el marcador NoData.
NoData	`NODATA`	[numérico: doble] Predeterminado: 0.0	Marcador NoData para rellenar puntos vacíos
Interpolado (IDW con búsqueda NN)	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida con valores interpolados. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor Z de campo Opcional	`Z_FIELD`	[campo de tabla: numérico]	Campo para la interpolación
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Interpolado (IDW con búsqueda NN)	`OUTPUT`	[ráster]	Ráster de salida con valores interpolados

Código Python

Algoritmo ID: `gdal:gridinversedistancenearestneighbor`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cuadrícula (distancia inversa a una fuente)

El método de cuadrícula de distancia inversa a una fuente es un interpolador promedio ponderado.

Debe proporcionar los arreglos de entrada con los valores de datos dispersos, incluidas las coordenadas de cada punto de datos y la geometría de la cuadrícula de salida. La función calculará el valor interpolado para la posición dada en la cuadrícula de salida.

Este algoritmo se deriva de la utilidad GDAL grid.

Menú predeterminado: *Raster ► Análisis*

Ver también

Tutorial de grid GDAL

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de puntos	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Poder de ponderación	`POWER`	[numérico: doble] Preestablecido: 2.0	Poder de ponderación
Suavizada	`SMOOTHING`	[numérico: doble] Predeterminado: 0.0	Parámetro de suavizado
El primer radio de búsqueda de elipse	`RADIUS_1`	[numérico: doble] Predeterminado: 0.0	El primer radio (eje X si el ángulo de rotación es 0) de la elipse de búsqueda
El segundo radio de la elipse de búsqueda	`RADIUS_2`	[numérico: doble] Predeterminado: 0.0	El segundo radio (eje Y si el ángulo de rotación es 0) de la elipse de búsqueda
Ángulo de rotación en grados de la elipse de búsqueda (sentido antihorario)	`ANGLE`	[numérico: doble] Predeterminado: 0.0	Ángulo de rotación de elipse en grados. Elipse rotada en sentido antihorario
Número máximo de datos puntuales a usar	`MAX_POINTS`	[numérico: entero] Predeterminado: 0	No buscar para mas puntos que este número.

continúe en la próxima página

Tabla 24.861 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Mínimo número de puntos de datos a usar	`MIN_POINTS`	[numérico: entero] Predeterminado: 0	Número mínimo de puntos de datos a promediar. Si se encuentran menos puntos, el nodo de la cuadrícula se considerará vacío y se rellenará con el marcador NoData.
NoData	`NODATA`	[numérico: doble] Predeterminado: 0.0	Marcador NoData para rellenar puntos vacíos
Interpolada (IDW)	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida con valores interpolados. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor Z de campo Opcional	`Z_FIELD`	[campo de tabla: numérico]	Campo para la interpolación
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

continúe en la próxima página

Tabla 24.862 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Interpolada (IDW)	`OUTPUT`	[ráster]	Ráster de salida con valores interpolados

Código Python

Algoritmo ID: `gdal:gridinversedistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cuadrícula (lineal)

El método Lineal realiza la interpolación lineal calculando una triangulación de Delaunay de la nube de puntos, encontrando en qué triángulo de la triangulación se encuentra el punto, y realizando la interpolación lineal a partir de sus coordenadas baricéntricas dentro del triángulo. Si el punto no está en ningún triángulo, dependiendo del radio, el algoritmo utilizará el valor del punto más cercano o el valor NoData.

Este algoritmo se deriva de la utilidad GDAL grid.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de puntos	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
Distancia de búsqueda	`RADIUS`	[numérico: doble] Predeterminado: -1.0	En caso de que el punto a interpolar no encaje en un triángulo de la triangulación de Delaunay, utiliza esa distancia máxima para buscar un vecino más cercano, o utiliza NoData en caso contrario. Si se establece en `-1`, la distancia de búsqueda es infinita. Si se establece en `0`, se utilizará el valor NoData.
NoData	`NODATA`	[numérico: doble] Predeterminado: 0.0	Marcador NoData para rellenar puntos vacíos
Interpolación (Lineal)	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida con valores interpolados. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor Z de campo Opcional	`Z_FIELD`	[campo de tabla: numérico]	Campo para la interpolación
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (`|`).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

continúe en la próxima página

Tabla 24.865 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Interpolación (Lineal)	`OUTPUT`	[ráster]	Ráster de salida con valores interpolados

Código Python

Algoritmo ID: `gdal:gridlinear`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cuadrícula(promedio móvil)

La media móvil es un algoritmo sencillo de promediación de datos. Utiliza una ventana móvil de forma elíptica para buscar valores y promedia todos los puntos de datos dentro de la ventana. La elipse de búsqueda se puede girar un ángulo especificado, el centro de la elipse se encuentra en el nodo de la cuadrícula. También se puede establecer el número mínimo de puntos de datos a promediar, si no hay suficientes puntos en la ventana, el nodo de cuadrícula se considera vacío y se rellenará con el valor NoData especificado.

Este algoritmo se deriva de la utilidad GDAL grid.

Menú predeterminado: *Raster ► Análisis*

Ver también

Tutorial de grid GDAL

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de puntos	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
El primer radio de búsqueda de elipse	`RADIUS_1`	[numérico: doble] Predeterminado: 0.0	El primer radio (eje X si el ángulo de rotación es 0) de la elipse de búsqueda
El segundo radio de la elipse de búsqueda	`RADIUS_2`	[numérico: doble] Predeterminado: 0.0	El segundo radio (eje Y si el ángulo de rotación es 0) de la elipse de búsqueda
Ángulo de rotación en grados de la elipse de búsqueda (sentido antihorario)	`ANGLE`	[numérico: doble] Predeterminado: 0.0	Ángulo de rotación de elipse en grados. Elipse rotada en sentido antihorario
Mínimo número de puntos de datos a usar	`MIN_POINTS`	[numérico: entero] Predeterminado: 0	Número mínimo de puntos de datos a promediar. Si se encuentran menos puntos, el nodo de la cuadrícula se considerará vacío y se rellenará con el marcador NoData.
NoData	`NODATA`	[numérico: doble] Predeterminado: 0.0	Marcador NoData para rellenar puntos vacíos
**Interpolación (promedio móvil)	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor Z de campo Opcional	`Z_FIELD`	[campo de tabla: numérico]	Campo para la interpolación
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

Salidas

Etiqueta	Nombre	Tipo	Descripción
**Interpolación (promedio móvil)	`OUTPUT`	[ráster]	Ráster de salida con valores interpolados

Código Python

Algoritmo ID: `gdal:gridaverage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cuadrícula (Vecino mas próximo)

El método Vecino más próximo no realiza ninguna interpolación ni suavizado, sólo toma el valor del punto más próximo encontrado en la elipse de búsqueda de nodos de la cuadrícula y lo devuelve como resultado. Si no se encuentra ningún punto, se devolverá el valor NoData especificado.

Este algoritmo se deriva de la utilidad GDAL grid.

Menú predeterminado: *Raster* ► *Análisis*

Ver también

Tutorial de grid GDAL

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de puntos	`INPUT`	[vector: punto]	Capa vectorial de puntos de entrada
El primer radio de búsqueda de elipse	`RADIUS_1`	[numérico: doble] Predeterminado: 0.0	El primer radio (eje X si el ángulo de rotación es 0) de la elipse de búsqueda
El segundo radio de la elipse de búsqueda	`RADIUS_2`	[numérico: doble] Predeterminado: 0.0	El segundo radio (eje Y si el ángulo de rotación es 0) de la elipse de búsqueda
Ángulo de rotación en grados de la elipse de búsqueda (sentido antihorario)	`ANGLE`	[numérico: doble] Predeterminado: 0.0	Ángulo de rotación de elipse en grados. Elipse rotada en sentido antihorario
NoData	`NODATA`	[numérico: doble] Predeterminado: 0.0	Marcador NoData para rellenar puntos vacíos
Interpolado (vecino mas cercano)	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida con valores interpolados. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor Z de campo Opcional	`Z_FIELD`	[campo de tabla: numérico]	Campo para la interpolación
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Interpolado (vecino mas cercano)	`OUTPUT`	[ráster]	Ráster de salida con valores interpolados

Código Python

Algoritmo ID: `gdal:gridnearestneighbor`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Sombreado

Genera un ráster con un agradable efecto de relieve sombreado. Es muy útil para visualizar el terreno. Opcionalmente, puede especificar el acimut y la altitud de la fuente de luz, un factor de exageración vertical y un factor de escala para tener en cuenta las diferencias entre las unidades verticales y horizontales.

Este algoritmo se deriva de la utilidad GDAL DEM .

Menú predeterminado: *Raster* ► *Análisis*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster Elevación de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	Banda contenedora de la información de elevación
Factor Z (exageración vertical)	`Z_FACTOR`	[numérico: doble] Predeterminado: 1.0	El factor exagera la altura del ráster de elevación saliente
Escala (relación entre unidades verticales y horizontales)	`SCALE`	[numérico: doble] Predeterminado: 1.0	El ratio de unidades verticales fente a unidades horizontales
Azimut de la luz	`AZIMUTH`	[numérico: doble] Predeterminado: 315.0	Define el acimut de la luz que brilla sobre el ráster de elevación en grados. Si viene de la parte superior del ráster el valor es 0, si viene del este es 90 a.s.o.
Altitud de la luz	`ALTITUDE`	[numérico: doble] Preestablecido: 45.0	Define la altitud de la luz, en grados. 90 si la luz proviene de arriba del ráster de elevación, 0 si es luz rasante.
Calcular bordes	`COMPUTE_EDGES`	[booleano] Predeterminado: Falso	Genera bordes a partir del ráster de elevación
Usar la fórmula Zevenbergen&Thorn (en lugar de la de Horn's)	`ZEVENBERGEN`	[booleano] Predeterminado: Falso	Activa la fórmula Zevenbergen&Thorne para paisajes suavizados
Sombreado combinado	`COMBINED`	[booleano] Predeterminado: Falso	

continúe en la próxima página

Tabla 24.873 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Sombreado multidireccional	`MULTIDIRECTION`	[booleano] Predeterminado: Falso	
Sombreado	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida con valores interpolados. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Sombreado	`OUTPUT`	[ráster]	Ráster de salida con valores interpolados

Código Python

Algoritmo ID: `gdal:hillshade`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Negro cercano

Convierte los bordes cercanos negros/blancos a negro.

Este algoritmo escaneará una imagen e intentará configurar todos los píxeles que son casi o exactamente negros, blancos o uno o más colores personalizados alrededor del cuello en blanco o negro. Esto se usa a menudo para «arreglar» fotos aéreas comprimidas con pérdida de modo que los píxeles de color puedan tratarse como transparentes al hacer mosaicos.

Este algoritmo se deriva de la utilidad GDAL nearblack.

Menú predeterminado: *Raster ► Análisis*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster Elevación de entrada
Como de lejos de negro (blanco)	`NEAR`	[numérico: entero] Preestablecido: 15	Seleccione como de lejos de los colores negro, blanco o personalizados pueden estar los valores de píxeles que aún se consideran cercanos al negro, blanco o color personalizado.
Buscar pixeles blancos cercanos en lugar de negros cercanos	`WHITE`	[booleano] Predeterminado: Falso	Buscar pixeles blancos cercanos (255) en lugar de pixeles negros cercanos
NegroCercano	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
NegroCercano	`OUTPUT`	[ráster]	Ráster saliente

Código Python

Algoritmo ID: `gdal:nearblack`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Proximidad (distancia ráster)

Genera un mapa de proximidad ráster que indica la distancia desde el centro de cada píxel al centro del píxel más cercano identificado como píxel de destino. Los píxeles de destino son aquellos en el ráster de origen para los que el valor de píxel de ráster está en el conjunto de valores de píxeles de destino.

Este algoritmo se deriva de la utilidad de proximidad GDAL.

Menú predeterminado: *Raster ► Análisis*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster Elevación de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	Banda contenedora de la información de elevación
List of target pixels Opcional	`VALUES`	[cadena de texto] Predeterminado: ""	Comma-separated list of pixel values in the source image to consider as target pixels. If not specified, all non-zero pixels will be considered target pixels.
Unidades de distancia	`UNITS`	[enumeración] Predeterminado: 1	Indique si las distancias generadas deben ser en píxeles o coordenadas georreferenciadas. Uno de: • 0 — Coordenadas Georreferenciadas • 1 — Coordenadas de pixel
The maximum distance to generate Opcional	`MAX_DISTANCE`	[numérico: doble] Predeterminado: 0.0	The maximum distance to be generated. The NoData value will be used for pixels beyond this distance. Distance is interpreted according to the value of the *Distance units* parameter.

continúe en la próxima página

Tabla 24.879 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Value to apply to pixels within the maximum distance Opcional	`REPLACE`	[numérico: doble] Predeterminado: 0.0	Specify a value to apply to all pixels that are within the maximum distance from target pixels (including the target pixels) instead of a distance value.
**Valor de NoData a utilizar para la trama de proximidad de destino. Opcional	`NODATA`	[numérico: doble] Predeterminado: 0.0	Specify the NoData value to use for the pixels beyond the maximum distance. If a NoData value is not provided, the output band will be queried for its NoData value. If the output band does not have a NoData value, then the value `65535` will be used.
Mapa de proximidad	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

continúe en la próxima página

Tabla 24.880 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Mapa de proximidad	`OUTPUT`	[ráster]	Ráster saliente

Código Python

Algoritmo ID: `gdal:proximity`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Rugosidad

Genera un ráster de banda única con valores calculados a partir de la elevación. La rugosidad es el grado de irregularidad de la superficie. Se calcula mediante la mayor diferencia entre celdas de un píxel central y su celda circundante. La determinación de la rugosidad juega un papel en el análisis de los datos de elevación del terreno, es útil para los cálculos de la morfología del río, en climatología y geografía física en general.

Este algoritmo se deriva de la utilidad GDAL DEM.

Menú predeterminado: *Raster* ► *Análisis*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de elevación de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	El número de la banda a usar como elevación
Calcular bordes	`COMPUTE_EDGES`	[booleano] Predeterminado: Falso	Genera bordes a partir del ráster de elevación
Rugosidad	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Rugosidad	`OUTPUT`	[ráster]	Raster de rugosidad de salida de una sola banda. El valor -9999 se utiliza como valor NoData.

Código Python

Algoritmo ID: `gdal:roughness`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Tamiz

Elimina los polígonos ráster más pequeños que un tamaño de umbral proporcionado (en píxeles) y los reemplaza con el valor de píxel del polígono vecino más grande. Es útil si tiene una gran cantidad de áreas pequeñas en su mapa ráster.

Este algoritmo se deriva de la utilidad GDAL sieve.

Menú predeterminado: *Raster* ► *Análisis*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de elevación de entrada
Umbral	`THRESHOLD`	[numérico: entero] Predeterminado: 10	Sólo los polígonos ráster menores que este tamaño serán eliminados
Usar conectividad 8	`EIGHT_CONNECTE`	[booleano] Predeterminado: Falso	Si es Falso, se consideran vecinos solo los (4) píxeles que conectan los bordes horizontal o verticalmente. Si es True, se considera vecino cada píxel que toca un borde o una esquina.
No usar la máscara de validez predeterminada para la banda de entrada	`NO_MASK`	[booleano] Predeterminado: Falso	
Mascara de validez Opcional	`MASK_LAYER`	[ráster]	Máscara de validez a utilizar en lugar de la predeterminada
Tamizado	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Tamizado	`OUTPUT`	[ráster]	Capa ráster saliente.

Código Python

Algoritmo ID: `gdal:sieve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Pendiente

Genera un mapa de pendientes a partir de cualquier ráster de elevación compatible con GDAL. La pendiente es el ángulo de inclinación con respecto a la horizontal. Tiene la opción de especificar el tipo de valor de pendiente que desea: grados o porcentaje de pendiente.

Este algoritmo se deriva de la utilidad GDAL DEM.

Menú predeterminado: *Raster ► Análisis*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster Elevación de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	Banda contenedora de la información de elevación
Ratio de unidades verticales a horizontales	`SCALE`	[numérico: doble] Predeterminado: 1.0	El ratio de unidades verticales fente a unidades horizontales

continúe en la próxima página

Tabla 24.888 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Express slope as percent (instead of degrees)	`AS_PERCENT`	[booleano] Predeterminado: Falso	Expresa la pendiente como porcentaje en lugar de grados
Calcular bordes	`COMPUTE_EDGES`	[booleano] Predeterminado: Falso	Genera bordes a partir del ráster de elevación
Usar la fórmula Zevenbergen&Thorn (en lugar de la de Horn's)	`ZEVENBERGEN`	[booleano] Predeterminado: Falso	Activa la fórmula Zevenbergen&Thorne para paisajes suavizados
Pendiente	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Pendiente	`OUTPUT`	[ráster]	Ráster saliente

Código Python

Algoritmo ID: `gdal:slope`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Indice de Rugosidad del Terreno (TRI)

Genera un ráster de banda única con valores calculados a partir de la elevación. TRI son las siglas de Terrain Ruggedness Index, que se define como la diferencia media entre un píxel central y las celdas circundantes.

Este algoritmo se deriva de la utilidad GDAL DEM.

Menú predeterminado: *Raster* ► *Análisis*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de elevación de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	El número de la banda a usar como elevación
Calcular bordes	`COMPUTE_EDGES`	[booleano] Predeterminado: Falso	Genera bordes a partir del ráster de elevación
Índice de rugosidad del Terreno	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Índice de rugosidad del Terreno	`OUTPUT`	[ráster]	Salida de la trama de robustez. El valor -9999 se utiliza como valor NoData.

Código Python

Algoritmo ID: `gdal:triterrainruggednessindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Índice de Posición Topográfica (TPI)

Genera un ráster de banda única con valores calculados a partir de la elevación. TPI significa Índice de posición topográfica, que se define como la diferencia entre un píxel central y la media de las celdas circundantes.

Este algoritmo se deriva de la utilidad GDAL DEM.

Menú predeterminado: *Raster* ► *Análisis*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de elevación de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	El número de la banda a usar para valores de elevación
Calcular bordes	`COMPUTE_EDGES`	[booleano] Predeterminado: Falso	Genera bordes a partir del ráster de elevación
Índice de rugosidad del Terreno	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa ráster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

Salidas

Etiqueta	Nombre	Tipo	Descripción
Índice de rugosidad del Terreno	`OUTPUT`	[ráster]	Raster saliente.

Código Python

Algoritmo ID: `gdal:tpitopographicpositionindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.2.2 Conversión raster

gdal2xyz

Convierte datos ráster a archivo de formato XYZ ASCII.

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster a convertir
Número de banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa entrante	Si el ráster es multibanda, elige la banda que quieras convertir

continúe en la próxima página

Tabla 24.897 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Valor del píxel de entrada a tratar como NoData Opcional	`NODATA_INPUT`	[numérico: entero] Predeterminado: No establecido	Input pixel value to treat as «NoData» **Atención** Using this parameter requires QGIS installed with GDAL >= 3.7 (see *Help* ► *About* menu).
Destino NoData Opcional	`NODATA_OUTPUT`	[numérico: entero] Predeterminado: No establecido	Assign specified «NoData» value to output. **Atención** Using this parameter requires QGIS installed with GDAL >= 3.7 (see *Help* ► *About* menu).
No mostrar valores NoData	`SKIP_NODATA`	[booleano] Predeterminado: Falso	Do not output «NoData» values. **Atención** Using this parameter requires QGIS installed with GDAL >= 3.3 (see *Help* ► *About* menu).
Valores separacos por comas salientes	`CSV`	[booleano] Predeterminado: Falso	Establece si el archivo de salida debe ser de tipo valores separados por comas (csv).
Archivo XYZ ASCII	`OUTPUT`	[archivo] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo XYZ ASCII	`INPUT`	[tabla]	Archivo de taba contenedor de los valores exportados desde la banda ráster.

Código Python

Algoritmo ID: `gdal:gdal2xyz`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

PCT a RGB

Convierte una imagen con paleta de 8 bits en un RGB de 24 bits. Convertirá una banda de pseudocolor del archivo de entrada a un archivo RGB del formato deseado.

Este algoritmo se deriva de la utilidad GDAL pct2rgb.

Menú predeterminado: *Raster ► Conversion*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Imagen ráster de 8 bits entrante
Número de banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa entrante	Si el ráster es multibanda, elige la banda que quieras convertir
Generar archivo RGBA	`RGBA`	[booleano] Predeterminado: Falso	Establece si el archivo de salida debe ser de tipo RGBA.
PCT a RGB	`OUTPUT`	[archivo] Predeterinado: `[Save to temporary file]`	Especificación del archivo de salida. *Uno de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
PCT a RGB	`OUTPUT`	[ráster]	Imagen ráster RGB de 24 bit

Código Python

Algoritmo ID: `gdal:pcttorgb`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Poligonizar (ráster a vectorial)

Crea polígonos vectoriales para todas las regiones conectadas de píxeles en el ráster que comparten un valor de píxel común. Cada polígono se crea con un atributo que indica el valor de píxel de ese polígono.

Este algoritmo se deriva de la utilidad GDAL polygonize.

Menú predeterminado: *Raster* ► *Conversion*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: La primera banda de la capa entrante	Si el ráster es multibanda, elige la banda que deseas usar
Nombre del campo a crear	`FIELD`	[cadena de texto] Predeterminado: "DN"	Especifica el nombre de campo para los atributos de las regiones conectadas.
Usar conectividad 8	`EIGHT_CONNECTE`	[booleano] Predeterminado: Falso	Si no se establece, las celdas ráster deben tener un borde común para que se consideren conectadas (*conectividad 4*). Si se establece, las celdas ráster en contacto también se consideran conectadas (*conectividad 8*).
Vectorizado	`OUTPUT`	[vectorial: poligonal] Predeterinado: `[Save to temporary file]`	Especificación de la capa vectorial de salida (polígono). *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Vectorizado	`OUTPUT`	[vectorial: poligonal]	Capa vectorial saliente

Código Python

Algoritmo ID: `gdal:polygonize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Reorganizar bandas

Crea un nuevo ráster utilizando las bandas seleccionadas de una capa ráster determinada. El algoritmo también hace posible reordenar las bandas para el ráster recién creado.

Este algoritmo se deriva de la utilidad GDAL translate.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de entrada
Banda(s) seleccionadas	`BANDS`	[banda ráster] [lista] Predeterminado: No establecido	Lista ordenada de las bandas a usar para crear el nuevo ráster
Convertido	`OUTPUT`	[ráster] Predeterminado: `[Guardar en capa temporal]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 0	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Utilizar tipo de datos de la capa de entrada • 1 — Byte (Enteros de ocho bit no firmado (quint8)) • 2 — Int16 (Entero de dieciséis bit firmado (qint16)) • 3 — UInt16 (Entero de dieciséis bit no firmado (quint16)) • 4 — UInt32 (Entero de treinta y dos bit no firmado (quint32)) • 5 — Int32 (Entero de treinta y dos bit firmado (qint32)) • 6 — Float32 (Punto flotante de treinta y dos bit (float)) • 7 — Float64 (Punto flotante de sesenta y cuatro bit (double)) • 8 — CInt16 (Complejo Int16) • 9 — CInt32 (Complejo Int32) • 10 — CFloat32 (Complejo Float32) • 11 — CFloat64 (Complejo Float64) • 12 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Convertido	`OUTPUT`	[ráster]	Capa ráster saliente con bandas reorganizadas.

Código Python

Algoritmo ID: `gdal:rearrange_bands`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

RGB a PCT

Convierte una imagen RGB de 24 bits en una paleta de 8 bits. Calcula una tabla de pseudo-color óptima para la imagen RGB dada utilizando un algoritmo de corte medio en un histograma RGB con muestreo reducido. Luego, convierte la imagen en una imagen pseudo-coloreada usando la tabla de colores. Esta conversión utiliza difuminado Floyd-Steinberg (difusión de errores) para maximizar la calidad visual de la imagen de salida.

Si desea clasificar un mapa ráster y desea reducir el número de clases, puede ser útil reducir la resolución de su imagen con este algoritmo antes.

Este algoritmo se deriva de la utilidad GDAL rgb2pct.

Menú predeterminado: *Raster* ► *Conversion*

Parametros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster entrante (RGB)
Número de colores	`NCOLORS`	[numérico: entero] Preestablecido: 2	El número de colores que contendrá la imagen resultante. Es posible un valor entre 2-256.
RGB a PCT	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
RGB a PCT	`OUTPUT`	[ráster]	Capa ráster saliente.

Código Python

Algoritmo ID: `gdal:rgbtopct`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Traducir (convertir formato)

Convierte datos ráster entre diferentes formatos.

Este algoritmo se deriva de la utilidad GDAL translate.

Menú predeterminado: *Raster ► Conversion*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de entrada
Anular la proyección del archivo de salida Opcional	`TARGET_CRS`	[src]	Especifica una proyección para el archivo saliente
Asignar un valor NoData especificado a las bandas de salida Opcional	`NODATA`	[numérico: doble] Predeterminado: No establecido	Define el valor que se utilizará para NoData en el ráster de salida
Copie todos los subconjuntos de datos de este archivo en archivos de salida individuales	`COPY_SUBDATASE`	[booleano] Predeterminado: Falso	Crear archivos individuales para subconjuntos de datos
Convertido	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la capa ráster de salida (traducida). *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 0	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Utilizar tipo de datos de la capa de entrada • 1 — Byte (Enteros de ocho bit no firmado (quint8)) • 2 — Int16 (Entero de dieciséis bit firmado (qint16)) • 3 — UInt16 (Entero de dieciséis bit no firmado (quint16)) • 4 — UInt32 (Entero de treinta y dos bit no firmado (quint32)) • 5 — Int32 (Entero de treinta y dos bit firmado (qint32)) • 6 — Float32 (Punto flotante de treinta y dos bit (float)) • 7 — Float64 (Punto flotante de sesenta y cuatro bit (double)) • 8 — CInt16 (Complejo Int16) • 9 — CInt32 (Complejo Int32) • 10 — CFloat32 (Complejo Float32) • 11 — CFloat64 (Complejo Float64) • 12 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Convertido	`OUTPUT`	[ráster]	Capa ráster de salida (traducida).

Código Python

Algoritmo ID: `gdal:translate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.2.3 Extracción ráster

Cortar ráster por extensión

Recorta cualquier archivo ráster compatible con GDAL en una medida determinada.

Este algoritmo se deriva de la utilidad GDAL translate.

Menu Predeterminado: *Raster* ► *Extracción*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	El ráster entrante
Cortando extensión	`EXTENT`	[extensión]	Extensión que debe usarse para el ráster de salida. Solo los píxeles dentro del cuadro delimitador especificado se incluirán en la salida. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
**Anular la proyección del fichero de salida.	`OVERCRS`	[booleano] Predeterminado: Falso	Si está marcado, al archivo de salida se le asigna el SRC de la capa de entrada.

continúe en la próxima página

Tabla 24.912 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Asignar un valor NoData especificado a las bandas de salida Opcional	`NODATA`	[numérico: doble] Predeterminado: No establecido	Define un valor que debe insertarse para los valores NoData en el ráster de salida
Cortado (extensión)	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la capa raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 0	Define el formato del archivo ráster saliente Opciones: • 0 — Utilizar tipo de datos de la capa de entrada • 1 — Byte (Enteros de ocho bit no firmado (quint8)) • 2 — Int16 (Entero de dieciséis bit firmado (qint16)) • 3 — UInt16 (Entero de dieciséis bit no firmado (quint16)) • 4 — UInt32 (Entero de treinta y dos bit no firmado (quint32)) • 5 — Int32 (Entero de treinta y dos bit firmado (qint32)) • 6 — Float32 (Punto flotante de treinta y dos bit (float)) • 7 — Float64 (Punto flotante de sesenta y cuatro bit (double)) • 8 — CInt16 (Complejo Int16) • 9 — CInt32 (Complejo Int32) • 10 — CFloat32 (Complejo Float32) • 11 — CFloat64 (Complejo Float64) • 12 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

continúe en la próxima página

Tabla 24.913 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cortado (extensión)	`OUTPUT`	[ráster]	Capa ráster de salida recortada por la extensión dada

Código Python

Algoritmo ID: `gdal:cliprasterbyextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cortar ráster por capa de máscara

Corta cualquier ráster GDAL soportado por una capa vectorial de máscara.

Este algoritmo se deriva de la utilidad GDAL warp.

Menu Predeterminado: *Raster ► Extracción*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	El ráster entrante
Capa de máscara	`MASK`	[vectorial: poligonal]	Mascara vectorial para cortar el ráster
SRC de fuente	`SOURCE_CRS`	[src]	Establece la coordenada de referencia a usar para el ráster entrante
SRC destino	`TARGET_CRS`	[src]	Establece la coordenada de referencia a usar para la capa de máscara

continúe en la próxima página

Tabla 24.915 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extensión del objetivo Opcional	`TARGET_EXTENT`	[extensión]	Extensión del archivo de salida que se va a crear Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Asignar un valor NoData especificado a las bandas de salida Opcional	`NODATA`	[numérico: doble] Predeterminado: No establecido	Define un valor que debe insertarse para los valores NoData en el ráster de salida
Crear una banda alfa saliente	`ALPHA_BAND`	[booleano] Predeterminado: Falso	Crea una banda alfa para el resultado. La banda alfa luego incluye los valores de transparencia de los píxeles.
Hacer coincidir la extensión del ráster recortado con la extensión de la capa de máscara	`CROP_TO_CUTLIN`	[booleano] Predeterminado: Verdadero	Aplica la extensión de la capa vectorial al ráster de salida si está marcado.
Mantener resolución de ráster de entrada	`KEEP_RESOLUTIO`	[booleano] Predeterminado: Falso	La resolución del ráster saliente no será cambiada
Configura la resolución del archivo de salida	`SET_RESOLUTION`	[booleano] Predeterminado: Falso	Se debe especificar la resolución de salida (tamaño de celda)
****Resolución X para bandas salientes* Opcional	`X_RESOLUTION`	[numérico: doble] Predeterminado: No establecido	La anchura de las celdas en el ráster saliente
Resolución Y para la banda saliente Opcional	`Y_RESOLUTION`	[numérico: doble] Predeterminado: No establecido	La altura de las celdas en el ráster saliente

continúe en la próxima página

Tabla 24.915 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Usar una implementación de deformación multiproceso	`MULTITHREADING`	[booleano] Predeterminado: Falso	Se utilizarán dos subprocesos para procesar fragmentos de imagen y realizar operaciones de entrada/salida simultáneamente. Tenga en cuenta que el cálculo no es multiproceso en sí mismo.
Cortado (mascara)	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la capa raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

continúe en la próxima página

Tabla 24.916 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 0	Define el formato del archivo ráster saliente Opciones: • 0 — Utilizar tipo de datos de la capa de entrada • 1 — Byte (Enteros de ocho bit no firmado (quint8)) • 2 — Int16 (Entero de dieciséis bit firmado (qint16)) • 3 — UInt16 (Entero de dieciséis bit no firmado (quint16)) • 4 — UInt32 (Entero de treinta y dos bit no firmado (quint32)) • 5 — Int32 (Entero de treinta y dos bit firmado (qint32)) • 6 — Float32 (Punto flotante de treinta y dos bit (float)) • 7 — Float64 (Punto flotante de sesenta y cuatro bit (double)) • 8 — CInt16 (Complejo Int16) • 9 — CInt32 (Complejo Int32) • 10 — CFloat32 (Complejo Float32) • 11 — CFloat64 (Complejo Float64) • 12 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cortado (mascara)	`OUTPUT`	[ráster]	Capa ráster saliente cortada por la capa vectorial

Código Python

Algoritmo ID: `gdal:cliprasterbymasklayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Curvas de nivel

Extrae líneas de contorno a partir de cualquier ráster de elevaciones compatible con GDAL.

Este algoritmo se deriva de la herramienta de contorno GDAL <https://gdal.org/en/latest/programs/gdal_contour.html>_.

Menu Predeterminado: *Raster* ► *Extracción*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Ráster de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	Banda ráster de la cuál crear los contornos
Intervalo entre curvas de nivel	`INTERVAL`	[numérico: doble] Predeterminado: 10.0	Define el intervalo enter las líneas de contorno en las undades dadas del ráster de elevación (valor mínimo 0)
Nombre de atributo (si no se configura, no se adjunta ningún atributo de elevación) Opcional	`FIELD_NAME`	[cadena de texto] Predeterminado: "ELEV"	Proporciona un nombre para el atributo en el cuál poner la elevación.
Desplazamiento desde cero con respecto al cual interpretar los intervalos Opcional	`OFFSET`	[numérico: doble] Predeterminado: 0.0	
Curvas de nivel	`OUTPUT`	[vectorial: lineal] Predeterinado: `[Save to temporary file]`	Especificación de la capa vectorial de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Producir vectorial 3D	`CREATE_3D`	[booleano] Predeterminado: Falso	Fuerza la producción de vectoriales 3D en lugar de 2D. Incluye la elevación de todos los vértices.
Tratar todos los valores ráster como válidos	`IGNORE_NODATA`	[booleano] Predeterminado: Falso	Ignora cualquier valor NoData en el conjunto de datos.
Valor del píxel de entrada a tratar como «NoData « Opcional	`NODATA`	[numérico: doble] Predeterminado: No establecido	Define un valor que debe insertarse para los valores NoData en el ráster de salida

continúe en la próxima página

Tabla 24.919 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Agregar opciones de línea de comando GDAL adicionales. Consulte la documentación de la utilidad GDAL correspondiente.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Curvas de nivel	`OUTPUT`	[vectorial: lineal]	Capa vectorial saliente con las curvas de nivel

Código Python

Algoritmo ID: `gdal:contour`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Polígonos de Contorno

Extráe ploígonos de contorno de cualquier ráster de elevación con soporte GDAL.

Este algoritmo se deriva de la herramienta de contorno GDAL <https://gdal.org/en/latest/programs/gdal_contour.html>_.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Ráster de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	Banda ráster de la cuál crear los contornos
Intervalo entre curvas de nivel	`INTERVAL`	[numérico: doble] Predeterminado: 10.0	Define el intervalo enter las líneas de contorno en las undades dadas del ráster de elevación (valor mínimo 0)
Desplazamiento desde cero con respecto al cual interpretar los intervalos Opcional	`OFFSET`	[numérico: doble] Predeterminado: 0.0	

continúe en la próxima página

Tabla 24.921 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Nombre de atributo para la elevación mínima de polígono de contorno Opcional	`FIELD_NAME_MIN`	[cadena de texto] Predeterminado: "ELEV_MIN"	Proporciona un nombre para el atributo en el que colocar la elevación mínima del polígono de contorno. Si no se proporciona, no se adjunta ningún atributo de elevación mínima.
Nombre de atributo para la elevación máxima de polígono de contorno Opcional	`FIELD_NAME_MAX`	[cadena de texto] Predeterminado: "ELEV_MAX"	Proporciona un nombre para el atributo en el que colocar la elevación máxima del polígono de contorno. Si no se proporciona, no se adjunta ningún atributo de elevación máxima.
Curvas de nivel	`OUTPUT`	[vectorial: poligonal] Predeterinado: `[Save to temporary file]`	Especificación de la capa vectorial de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Producir vectorial 3D	`CREATE_3D`	[booleano] Predeterminado: Falso	Fuerza la producción de vectoriales 3D en lugar de 2D. Incluye la elevación de todos los vértices.
Tratar todos los valores ráster como válidos	`IGNORE_NODATA`	[booleano] Predeterminado: Falso	Ignora cualquier valor NoData en el conjunto de datos.
Valor del píxel de entrada a tratar como «NoData « Opcional	`NODATA`	[numérico: doble] Predeterminado: No establecido	Define un valor que debe insertarse para los valores NoData en el ráster de salida
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Agregar opciones de línea de comando GDAL adicionales. Consulte la documentación de la utilidad GDAL correspondiente.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Curvas de nivel	`OUTPUT`	[vectorial: poligonal]	Capa vectorial saliente con polígonos de contorno

Código Python

ID Algoritmo: `gdal:contour_polygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.2.4 Miscelánea Ráster

Construir vista general (pirámides)

Para acelerar el tiempo de renderizado de capas ráster, se pueden crear vistas generales (pirámides). Las vistas generales son copias de menor resolución de los datos que utiliza QGIS según el nivel de zoom.

Este algoritmo se deriva de la utilidad GDAL addo.

Menú predeterminado: *Raster ► Misceláneo*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de entrada
Borrar todas las vistas generales existentes	`CLEAN`	[booleano] Predeterminado: False	Elimina las vistas generales existentes del ráster. Por defecto, estos no se eliminan.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Niveles de vista general	`LEVELS`	[cadena de texto] Predeterminado: “2 4 8 16”	Define el número de niveles de vista general calculados por la resolución original de la capa ráster de entrada. Por defecto se tendrán en cuenta 4 niveles.

continúe en la próxima página

Tabla 24.925 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Método de remuestreo Opcional	`RESAMPLING`	[enumeración] Predeterminado: 0	Calcula las vistas generales con un método de remuestreo definido. Los posibles métodos de remuestreo son: • 0 – Vecino mas próximo (`nearest`) • 1 – Promedio (`average`) • 2 – Gaussiano (`gauss`) • 3 – Cúbico (núcleo 4x4) (`cubic`) • 4 – B-Spline cúbica (núcleo 4x4) (`cubicspline`) • 5 – Lanczos (núcleo 6x6) (`lanczos`) • 6 – Promedio MP (`average_mp`) • 7 – Promedio en Mag/Phase Space (`average_magphase`) • 8 – Modo (`mode`)
Formato de vistas generales Opcional	`FORMAT`	[enumeración] Predeterminado: 0	Las vistas generales se pueden almacenar internamente o externamente como un archivo GTiff o ERDAS Imagine. De forma predeterminada, las vistas generales se almacenan en el ráster de salida. Los posibles métodos de formatos son: • 0 – Interna (si es posible) • 1 – Externa (GTiff .ovr) • 2 – Externa (ERDAS Imagine .aux)
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Piramizado	`OUTPUT`	[ráster]	Capa ráster saliente con vistas generales

Código Python

Algoritmo ID: `gdal:overviews`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Construir ráster virtual

Crea un VRT (conjunto de datos virtuales) que es un mosaico de la lista de rásteres de entrada compatibles con GDAL. Con un mosaico puede combinar varios archivos ráster.

Este algoritmo se deriva de la utilidad GDAL buildvrt.

Menú predeterminado: *Raster* ► *Misceláneo*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Capas ráster con compatibilidad GDAL.
Resolución	`RESOLUTION`	[enumeración] Predeterminado: 0	La resolución de salida del mosaico. De forma predeterminada, se elegirá la resolución media de los archivos ráster. Opciones: • 0 — Promedio (`average`) • 1 — Mas alto (`highest`) • 2 — Mas bajo (`lowest`)
Ubicar cada archivo entrante en una banda separada	`SEPARATE`	[booleano] Predeterminado: False	Con “Verdadero” puede definir que cada archivo ráster vaya a una banda apilada separada en la banda VRT.
Permitir diferencia de proyección	`PROJ_DIFFERENC`	[booleano] Predeterminado: False	Permite que las bandas de salida tengan diferentes proyecciones derivadas de la proyección de las capas ráster de entrada.
Virtual	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la capa raster de salida. *Una de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Agregar una banda de máscara alfa a VRT cuando el ráster de origen no tenga ninguna	`ADD_ALPHA`	[booleano] Predeterminado: False	Agregar una banda de máscara alfa al VRT cuando el ráster de origen no tiene ninguna.
Suplantar la proyección del archivo de salida Opcional	`ASSIGN_CRS`	[src] Predeterminado: No establecido	Anula la proyección del archivo de salida. No se hace ninguna reprobación.

continúe en la próxima página

Tabla 24.928 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Algoritmo de remuestreo	`RESAMPLING`	[enumeración] Predeterminado: 0	El algoritmo de remuestreo a utilizar. Opciones: • 0 — Vecino mas próximo (`nearest`) • 1 — Bilineal (núcleo 2x2) (`bilinear`) • 2 — Cúbico (núcleo 4x4) (`cubic`) • 3 — B-Spline cúbica (núcleo 4x4) (`cubicspline`) • 4 — Lanczos (núcleo 6x6) (`lanczos`) • 5 — Promedio (`average`) • 6 — Modo (`mode`)
Valor(es) de NoData para las bandas de entrada (separados por espacios) Opcional	`SRC_NODATA`	[cadena de texto] Predeterminado: No establecido	Valor(es) NoData separado(s) por espacios para la(s) banda(s) de entrada
Parámetros adicionales de la línea de comandos	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Virtual	`OUTPUT`	[ráster]	Capa ráster saliente

Código Python

Algoritmo ID: `gdal:buildvirtualraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

gdal2tiles

Genera un directorio con pequeñas teselas y metadatos, siguiendo la OSGeo Tile Map Service Specification. Ver también el OpenGIS Web Map Tile Service Implementation Standard. También se generan páginas web sencillas con visores basados en Google Maps, OpenLayers y Leaflet. Para explorar sus mapas en línea en el navegador web, sólo tiene que cargar el directorio generado en un servidor web.

Este algoritmo también crea los metadatos necesarios para Google Earth (KML SuperOverlay), en caso de que el mapa suministrado utilice la proyección `EPSG:4326`.

Los archivos de mundo de ESRI y la georreferenciación incrustada se utilizan durante la generación de mosaicos, pero también puede publicar una imagen sin la georreferenciación adecuada.

Este algoritmo se deriva de la utilidad GDAL gdal2tiles.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster compatible con GDAL.
Perfil de corte de teselas	`PROFILE`	[enumeración] Predeterminado: 0	Una de: • 0 — Mercator (`mercator`) • 1 — Geodésico (`geodetic`) • 2 — Ráster (`raster`)
Niveles de Zoom a representar Opcional	`ZOOM`	[cadena de texto] Predeterminado: “”	
Visor Web para generar	`VIEWER`	[enumeración] Predeterminado: 0	Una de: • 0 — Todo (`all`) • 1 — GoogleMaps (`google`) • 2 — OpenLayers (`openlayers`) • 3 — Leaflet (`leaflet`) • 4 — Ninguno (`none`)
Título del mapa Opcional	`TITLE`	[cadena de texto] Predeterminado: “”	
Copyright del mapa	`COPYRIGHT`	[cadena de texto] Predeterminado: “”	
Directorio de salida	`OUTPUT`	[carpeta] Predeterminado: `[Guardar en directorio temporal]`	Especifique la carpeta de salida para las teselas. *Una de*: • Guardar en Directorio Temporal • Guardar en Directorio

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Método de remuestreo	`RESAMPLING`	[enumeración] Predeterminado: 0	El algoritmo de remuestreo a utilizar. Opciones: • 0 — Promedio (`average`) • 1 — Vecino más próximo (`near`) • 2 — Bilineal (núcleo 2x2) (`bilinear`) • 3 — Cúbico (núcleo 4x4) (`cubic`) • 4 — B-Spline cúbica (núcleo 4x4) (`cubicspline`) • 5 — Lanczos (núcleo 6x6) (`lanczos`) • 6 — Antialias (`antialias`)
El sistema de referencia espacial utilizado para los datos de entrada de origen Opcional	`SOURCE_CRS`	[src] Predeterminado: No establecido	
Valor de transparencia a asignar a los datos entrantes Opcional	`NODATA`	[numérico: doble] Predeterminado: 0.0	
Dirección URL donde se publicarán los mosaicos generados Opcional	`URL`	[cadena de texto] Predeterminado: “”	
Clave de Google Maps API (http://code.google.c Opcional	`GOOGLE_KEY`	[cadena de texto] Predeterminado: “”	Tu clave API de Google Maps.
Clave de Bing Maps API (https://www.bingm Opcional	`BING_KEY`	[cadena de texto] Predeterminado: “”	Tu clave de la API de Bing maps
Generar solo archivos faltantes	`RESUME`	[booleano] Predeterminado: False	
Generar KML para Google Earth	`KML`	[booleano] Predeterminado: False	
Evitar la generación automática de archivos KML para EPSG: 4326	`NO_KML`	[booleano] Predeterminado: False	

Salidas

Etiqueta	Nombre	Tipo	Descripción
Directorio de salida	`OUTPUT`	[carpeta]	La carpeta saliente (para las teselas)

Código Python

Algoritmo ID: `gdal:gdal2tiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Mezclar

Fusiona archivos ráster de forma sencilla. Aquí puede utilizar una tabla de pseudocolor de un ráster de entrada y definir el tipo de ráster de salida. Todas las imágenes deben estar en el mismo sistema de coordenadas.

Este algoritmo se deriva de la utilidad de fusión GDAL.

Menú predeterminado: *Raster ► Misceláneo*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capas de entrada	`INPUT`	[ráster] [lista]	Capas ráster entrantes
Coge la tabla de pseudocolor de la primera capa	`PCT`	[booleano] Predeterminado: False	La tabla de pseudocolor de la primera capa se utilizará para colorear
Ubicar cada archivo entrante en una banda separada	`SEPARATE`	[booleano] Predeterminado: False	Colocar cada archivo de entrada en una banda separada

continúe en la próxima página

Tabla 24.933 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el formato del fichero raster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)
Mezclado	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la capa raster de salida. *Una de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Valor del píxel de entrada a tratar como «NoData « Opcional	`NODATA_INPUT`	[numérico: doble] Predeterminado: No establecido	Ignora los píxeles de los archivos que se fusionan con este valor de píxel
Asignar valor especificado «NoData» a la salida Opcional	`NODATA_OUTPUT`	[numérico: doble] Predeterminado: No establecido	Asigna el valor NoData especificado a las bandas de salida.
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).

continúe en la próxima página

Tabla 24.934 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Parámetros adicionales de la línea de comandos	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Mezclado	`OUTPUT`	[ráster]	Capa ráster saliente

Código Python

Algoritmo ID: `gdal:merge`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Pansharpening

Realiza una operación de afilado panorámico. Puede crear un conjunto de datos de salida «clásico» (como GeoTIFF) o un conjunto de datos VRT que describa la operación de pan-sharpening.

Véase GDAL Pansharpen.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Conjunto de datos espectrales	`SPECTRAL`	[ráster]	capa vectorial entrante (espectral)
Conjunto de datos pancromático	`PANCHROMATIC`	[ráster]	Capa ráster entrante (pancromática)
Salida	`OUTPUT`	[ráster] Predeterminado: `[Save to temporary file]`	Especifique la capa raster de salida (mejorada). *Una de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Algoritmo de remuestreo	`RESAMPLING`	[enumeración] Preestablecido: 2	El algoritmo de remuestreo a utilizar. Opciones: • 0 — Vecino mas próximo (`nearest`) • 1 — Bilineal (núcleo 2x2) (`bilinear`) • 2 — Cúbico (núcleo 4x4) (`cubic`) • 3 — B-Spline cúbica (núcleo 4x4) (`cubicspline`) • 4 — Lanczos (núcleo 6x6) (`lanczos`) • 5 — Promedio (`average`)
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Salida	`OUTPUT`	[ráster]	Cpa ráster saliente (sombreada)

Código Python

Algoritmo ID: `gdal:pansharp`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Calculadora ráster

Calculadora de ráster de línea de comando con sintaxis numpy. Utilice cualquier aritmética básica compatible con matrices numpy, como +, -, * y / junto con operadores lógicos, como>. Tenga en cuenta que todos los rásteres de entrada deben tener las mismas dimensiones, pero no se realiza ninguna comprobación de proyección.

Véase la documentación de la utilidad GDAL Raster Calculator.

Ver también

Calculadora ráster, *Calculadora Ráster (virtual)*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa entrante A	`INPUT_A`	[ráster]	Primera capa ráster entrante (obligatoria)
Número de bandas ráster para A	`BAND_A`	[banda ráster]	Banda para capa entrante A (obligatoria)
Capa entrante B Opcional	`INPUT_B`	[ráster] Predeterminado: No establecido	Segunda capa ráster entrante
Número de banda ráster para B Opcional	`BAND_B`	[banda ráster]	Banda para capa entrante B
Capa entrante C Opcional	`INPUT_C`	[ráster] Predeterminado: No establecido	Tercera capa ráster entrante
Número de banda ráster para C Opcional	`BAND_C`	[banda ráster]	Banda para la capa entrante C
Capa entrante D Opcional	`INPUT_D`	[ráster] Predeterminado: No establecido	Cuarta capa ráster entrante
Número de banda ráster para D Opcional	`BAND_D`	[banda ráster]	Banda para capa entrante D
Capa entrante E Opcional	`INPUT_E`	[ráster] Predeterminado: No establecido	Quinta capa ráster entrante
Número de banda ráster para E Opcional	`BAND_E`	[banda ráster]	Banda para capa E
Capa entrante F Opcional	`INPUT_F`	[ráster]	Sexta capa ráster entrante
Número de banda ráster para F Opcional	`BAND_F`	[banda ráster] Predeterminado: No establecido	Banda para capa entrante F

continúe en la próxima página

Tabla 24.939 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Cálculo en sintaxis gdalnumérica usando + - / * o cualquier función de matriz numérica (es decir, logic_and())	`FORMULA`	[cadena de texto] Predeterminado: ""	La fórmula de cálculo. Ejemplos: • `A*(A>0)` — genera el valor del ráster A si el valor de A es mayor que 0. Si no, genera 0. • `A*(A>0 and A>B)` — genera el valor de A si ese valor es mayor que 0 y mayor que el valor de B. Si no, genera 0. • `A*logical_or(A<=177, A>=185)` — emite el valor de A si A <= 177 o A> = 185. Si no, emite 0. • `sqrt(A*A+B*B)` — Da como resultado la raíz cuadrada de la suma del valor de A al cuadrado y el valor de B al cuadrado.
Ajustar valor de salida NoData Opcional	`NO_DATA`	[numérico: doble] Predeterminado: No establecido	Valor a utilizar para NoData
Tratamiento de las diferencias de extensión	`EXTENT_OPT`	[enumeración] Predeterminado: 0	Determines how to handle rasters with different extents. Supported options are: • 0 — Ignora • 1 — Falla • 2 — Unión • 3 — Intersecta **Atención** Using this parameter requires QGIS installed with GDAL >= 3.3 (see *Help* ► *About* menu).

continúe en la próxima página

Tabla 24.939 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extensión de salida Opcional	`INPUT`	[extensión]	Custom extent of the output raster. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax` **Atención** Using this parameter requires QGIS installed with GDAL >= 3.3 (see *Help* ► *About* menu).
Tipo de ráster saliente	`RTYPE`	[enumeración] Predeterminado: 5	Define el tipo de datos del archivo ráster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)
Calculado	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especifique la capa raster de salida (calculada). *Una de*: • Guardar en Archivo Temporal • Guardar en archivo…

continúe en la próxima página

Tabla 24.939 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: ""	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Calculado	`OUTPUT`	[ráster]	Capa ráster saliente (calculada)

Código Python

Algoritmo ID: `gdal:rastercalculator`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Información ráster

El programa gdalinfo lista diversa información sobre un dataset ráster compatible con GDAL.

Este algoritmo se deriva de la utilidad GDAL info.

Menú predeterminado: *Raster ► Misceláneo*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de entrada
Forzar el cálculo de los valores mínimos/máximos reales para cada banda	`MIN_MAX`	[booleano] Predeterminado: False	Fuerza el cálculo de los valores mínimos/máximos reales para cada banda en el conjunto de datos
Leer y mostrar estadísticas de la imagen (forzar cálculo si es necesario)	`STATS`	[booleano] Predeterminado: False	Lee y muestra estadísticas de imágenes. Fuerza el cálculo si no se almacenan estadísticas en una imagen.
Suprimir información de GCP	`NO_GCP`	[booleano] Predeterminado: False	Suprime la impresión de la lista de puntos de control terrestre. Puede ser útil para conjuntos de datos con una gran cantidad de GCP, como L1B AVHRR o HDF4 MODIS, que contienen miles de ellos.
Suprimir información de metadatos	`NO_METADATA`	[booleano] Predeterminado: False	Suprime la impresión de metadatos. Algunos conjuntos de datos pueden contener muchas cadenas de metadatos.
Información de la capa	`OUTPUT`	[html] Predeterinado: `[Save to temporary file]`	Especifique el archivo HTML de salida. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Información de la capa	`OUTPUT`	[html]	El archivo HTML contenedor de información sobre la capa ráster entrante

Código Python

Algoritmo ID: `gdal:gdalinfo`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Retesela

Retesela un conjunto de teselas de entrada. Todos los mosaicos de entrada deben estar georreferenciados en el mismo sistema de coordenadas y tener un número coincidente de bandas. Opcionalmente se generan niveles piramidales.

Este algoritmo se deriva de la utilidad GDAL Retile.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Archivos entrantes	`INPUT`	[ráster] [lista]	Los archivos ráster entrantes
Anchura de tesela	`TILE_SIZE_X`	[numérico: entero] Predeterminado: 256	Anchura de las teselas en pixels (mínimo 0)
Altura de tesela	`TILE_SIZE_Y`	[numérico: entero] Predeterminado: 256	Altura de teselas en pixeles (mínimo 0)
Superposición en píxeles entre teselas consecutivas	`OVERLAP`	[numérico: entero] Predeterminado: 0	
Número de niveles piramidales a construir	`LEVELS`	[numérico: entero] Predeterminado: 1	Mínimo: 0
Directorio de salida	`OUTPUT`	[carpeta] Predeterminado: `[Guardar en directorio temporal]`	Especifique la carpeta de salida para las teselas. *Una de*: • Guardar en Directorio Temporal • Guardar en Directorio
Archivo CSV contenedor de la información de georreferenciación de tesela(s)	`OUTPUT_CSV`	[archivo] Preestablecido: `[Omitir salida]`	Especifique el archivo de salida para las teselas. *Uno de*: • Saltar salida • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Sistema de coordenadas de referencia fuente Opcional	SOURCE_CRS	[src] Predeterminado: No establecido	
Método de remuestreo	RESAMPLING	[enumeración] Predeterminado: 0	El algoritmo de remuestreo a utilizar. Opciones: • 0 — Vecino mas próximo (nearest) • 1 — Bilineal (núcleo 2x2) (bilinear) • 2 — Cúbico (núcleo 4x4) (cubic) • 3 — B-Spline cúbica (núcleo 4x4) (cubicspline) • 4 — Lanczos (núcleo 6x6) (lanczos)
Delimitador de columna usado en archivo CSV Opcional	DELIMITER	[cadena de texto] Predeterminado: ";"	Delimitador a usar en archivo CSV contenedor de la información georreferenciada de tesela(s)
Opciones de creación adicional Opcional	CREATION_OPTIO (for QGIS <= 3.42, this was OPTIONS)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Parámetros adicionales de la línea de comandos Opcional	EXTRA	[cadena de texto] Predeterminado: ""	Añadir opciones extra de línea de comando GDAL

continúe en la próxima página

Tabla 24.946 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de datos salientes	DATA_TYPE	[enumeración] Predeterminado: 5	Define el formato del fichero raster de salida. Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)
Construir solo las pirámides	ONLY_PYRAMIDS	[booleano] Predeterminado: False	
Usar directorio separado para cada fila de tesela	DIR_FOR_ROW	[booleano] Predeterminado: False	

Salidas

Etiqueta	Nombre	Tipo	Descripción
Directorio de salida	OUTPUT	[carpeta]	La carpeta de salida para las teselas.
Archivo CSV contenedor de la información de georreferenciación de tesela(s)	OUTPUT_CSV	[archivo]	El archiv CSV con información de georreferenciación para las teselas.

Código Python

Algoritmo ID: `gdal:retile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

El índice

Crea una capa vectorial con un registro para cada archivo ráster de entrada, un atributo que contiene el nombre del archivo y una geometría poligonal que describe el ráster. Esta salida es adecuada para su uso con MapServer como índice de mosaico ráster.

Este algoritmo se deriva de la utilidad GDAL Tile Index.

Menú predeterminado: *Raster ► Misceláneo*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Archivos entrantes	`LAYERS`	[ráster] [lista]	Los archivos ráster entrantes. Pueden ser archivos múltiples.
Nombre de campo para contener la ruta del archivo a los rásteres indexados	`PATH_FIELD_NAM` Opcional	[cadena de texto] Predeterminado: “location”	El nombre del campo de salida para contener la ruta/ubicación del archivo a los rásteres indexados.
Almacenar la ruta absoluta a los rásteres indexados	`ABSOLUTE_PATH`	[booleano] Predeterminado: False	Establezca si la ruta absoluta a los archivos ráster se almacena en el archivo de índice de mosaico. De forma predeterminada, los nombres de archivo ráster se incluirán en el archivo exactamente como se especifican en el comando.
Omitir archivos con referencias de proyección diferentes	`PROJ_DIFFERENC`	[booleano] Predeterminado: False	Solo se insertarán los archivos con la misma proyección que los archivos ya insertados en el índice de teselas. El valor predeterminado no verifica la proyección y acepta todas las entradas.
Índice de teselas	`OUTPUT`	[vectorial: poligonal] Predeterinado: `[Save to temporary file]`	Especifique la capa vectorial de polígonos en la que se escribirá el índice. *Una de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Transforme geometrías al SRC dado Opcional	`TARGET_CRS`	[src]	Las geometrías de los archivos de entrada se transformarán en el sistema de referencia de coordenadas de destino especificado. Por defecto crea polígonos rectangulares simples en el mismo sistema de referencia de coordenadas que los rásteres de entrada.
El nombre del campo para almacenar el SRC de cada tesela Opcional	`CRS_FIELD_NAME`	[cadena de texto]	El nombre del campo para almacenar el SRC de cada mosaico
El formato en el que se debe escribir el CRS de cada tesela	`CRS_FORMAT`	[enumeración] Predeterminado: 0	Formato para el SRC. Uno de: • 0 – Auto (`AUTO`) • 1 – Texto bien conocido (`WKT`) • 2 – EPSG (`EPSG`) • 3 – Proj.4 (`PROJ`)

Salidas

Etiqueta	Nombre	Tipo	Descripción
Índice de teselas	`OUTPUT`	[vectorial: poligonal]	La capa vectorial poligonal con el índice de teselas.

Código Python

Algoritmo ID: `gdal:tileindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Viewshed

Calcula un ráster de vista a partir de un ráster DEM de entrada utilizando el método definido en Wang2000 para un punto definido por el usuario.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[ráster]	Capa ráster de elevación de entrada
Número de banda	`BAND`	[banda ráster] Predeterminado: 1	El número de la banda a usar como elevación
Ubicación del observador	`OBSERVER`	[coordinates]	La ubicación del observador
Altura del observador	`OBSERVER_HEIGH`	[numérico: doble] Predeterminado: 1.0	La altitud del observador, en unidades MDE
Altura objetivo	`TARGET_HEIGHT`	[numérico: doble] Predeterminado: 1.0	La altitud del elemento objetivo, en unidades MDE
Distancia máxima del observador para calcular la visibilidad	`MAX_DISTANCE`	[numérico: doble] Predeterminado: 100.0	Distancia máxima desde el observador para calcular la visibilidad, en unidades DEM
Salida	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Capa raster de salida. *Una de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (`|`).
Parámetros adicionales de la línea de comandos	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Salida	`OUTPUT`	[ráster]	La capa ráster que muestra la cuenca visual.

Código Python

Algoritmo ID: `gdal:viewshed`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.2.5 Proyecciones ráster

Asignar proyección

Aplica un sistema de coordenadas a un dataset ráster.

Este algoritmo se deriva de la utilidad de edición GDAL.

Menú predeterminado: *Raster* ► *Proyecciones*

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT_LAYER`	[raster]	Capa ráster de entrada
SRC Deseado	`CRS`	[src]	La proyección (SRC) de la capa saliente

Salidas

Etiqueta	Nombre	Tipo	Descripción
Capa con proyección	`OUTPUT`	[raster]	La capa ráster saliente (con la nueva información de proyección)

Código Python

Algoritmo ID: `gdal:assignprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Extraer poyección

Extrae la proyección de un archivo ráster y lo escribe en un archivo *mundo* con extensión `.wld`.

Este algoritmo se deriva de la utilidad GDAL srsinfo.

Menú predeterminado: *Raster ► Proyecciones*

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Archivo de entrada	`INPUT_LAYER`	[raster]	Ráster de entrada La capa ráster debe basarse en un archivo, ya que el algoritmo usa la ruta al archivo ráster como la ubicación del archivo generado `.wld`. El uso de una capa ráster que no sea de archivo provocará un error.
Crear también archivo .prj	`PRJ_FILE_CREAT`	[booleano] Predeterminado: False	Si está activado, también se crea un archivo `.prj` que contiene la información de la proyección.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Archivo mundo	`WORLD_FILE`	[archivo]	Archivo de texto con extensión `.wld` conteniendo los parámetros de transformación para el archivo ráster.
Archivo prj Shapefile de ESRI	`PRJ_FILE`	[archivo]	Archivo de texto con: archivo: extensión *.prj* que describe el SRC. Será `None` si *Create also .prj file* es Falso.

Código Python

Algoritmo ID: `gdal:extractprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Deformación (reproyectar)

Reproyecta una capa ráster en otro Sistema de referencia de coordenadas (SRC). Se puede elegir la resolución del archivo de salida y el método de remuestreo.

Este algoritmo se deriva de la utilidad GDAL warp.

Menú predeterminado: *Raster ► Proyecciones*

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[raster]	Capa ráster entrante a reproyectar
SRC de fuente Opcional	`SOURCE_CRS`	[src]	Define el SRC de la capa ráster entrante
SRC destino Opcional	`TARGET_CRS`	[src] Predeterminado: `EPSG:4326`	El SRC de la capa saliente
Método de remuestreo a usar	`RESAMPLING`	[enumeración] Predeterminado: 0	Método de remuestreo de valor de pixel a emplear. Opciones: • 0 — Vecino más próximo • 1 — Bilineal (núcleo 2x2) • 2 — Cúbico (núcleo 4x4) • 3 — B-Spline cúbica (núcleo 4x4) • 4 — Lanczos (núcleo 6x6) • 5 — Promedio • 6 — Modo • 7 — Máximo • 8 — Mínimo • 9 — Mediana • 10 — Primer cuartil (Q1) • 11 — Tercer cuartil (Q3)
Valor NoData para las bandas de salida Opcional	`NODATA`	[numérico: doble] Predeterminado: No establecido	Establece el valor NoData para las bandas de salida. Si no se proporciona, los valores NoData se copiarán del conjunto de datos de origen.

continúe en la próxima página

Tabla 24.958 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Resolución del archivo de salida en unidades georreferenciadas de destino Opcional	`TARGET_RESOLUT`	[numérico: doble] Predeterminado: No establecido	Define la resolución del archivo saliente de la reproyección resultante
Reproyectado	`OUTPUT`	[raster] Predeterminado: `[Save to temporary file]`	Especificación de la capa raster de salida. *Una de*: • Guardar en Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 0	Define el formato del fichero raster de salida. Opciones: • 0 — Utilizar tipo de datos de la capa de entrada • 1 — Byte (Enteros de ocho bit no firmado (quint8)) • 2 — Int16 (Entero de dieciséis bit firmado (qint16)) • 3 — UInt16 (Entero de dieciséis bit no firmado (quint16)) • 4 — UInt32 (Entero de treinta y dos bit no firmado (quint32)) • 5 — Int32 (Entero de treinta y dos bit firmado (qint32)) • 6 — Float32 (Punto flotante de treinta y dos bit (float)) • 7 — Float64 (Punto flotante de sesenta y cuatro bit (double)) • 8 — CInt16 (Complejo Int16) • 9 — CInt32 (Complejo Int32) • 10 — CFloat32 (Complejo Float32) • 11 — CFloat64 (Complejo Float64) • 12 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)

continúe en la próxima página

Tabla 24.959 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Extensión georreferenciada del archivo de salida que se creará Opcional	`TARGET_EXTENT`	[extensión]	Establece la extensión georreferenciada del archivo de salida que se creará (en *SRC Objetivo* de forma predeterminada. En *SRC de la extensión del ráster objetivo*, si se especifica). Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
SRC de la extensión de ráster de destino Opcional	`TARGET_EXTENT_`	[src]	Especifica el SRC en el que interpretar las coordenadas dadas para la extensión del archivo de salida. Esto no debe confundirse con el SRC de destino del conjunto de datos de salida. En cambio, es una conveniencia, p. cuando se conocen las coordenadas de salida en un SRC geodésico de longitud/latitud, pero se desea obtener un resultado en un sistema de coordenadas proyectadas.
Usar una implementación de deformación multiproceso	`MULTITHREADING`	[booleano] Predeterminado: False	Se utilizarán dos subprocesos para procesar fragmentos de la imagen y realizar operaciones de entrada/salida simultáneamente. Tenga en cuenta que el cálculo en sí no es multiproceso.
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: No establecido	Agregar opciones adicionales a la línea de comandos de GDAL.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Reproyectado	`OUTPUT`	[raster] Predeterminado: `[Save to temporary file]`	Capa ráster saliente reproyectada

Código Python

Algoritmo ID: `gdal:warpreproject`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

24.2.6 Conversión vectorial

Convertir formato

Convierte cualquier capa vectorial compatible con OGR a otro formato compatible OGR.

Este algoritmo se deriva de la utilidad ogr2ogr.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa de vector de entrada
Convertir todas las capas del conjunto de datos	`CONVERT_ALL_LA`	[booleano] Predeterminado: Falso	Convierte todo el conjunto de datos. Los formatos de salida admitidos para esta opción son `GPKG` y `GML`.
Convertido	`OUTPUT`	[la misma que la entrada]	Especificación de la capa vectorial de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo… Para `Guardar en archivo`, se debe especificar el formato de salida. Se admiten todos los formatos vectoriales GDAL. Para `Guardar en un archivo temporal` se utilizará el formato vectorial predeterminado de QGIS.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: "" (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Convertido	`OUTPUT`	[la misma que la entrada]	La capa vectorial saliente

Código Python

Algoritmo ID: `gdal:convertformat`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Rasterizar (sobreescribir con atributo)

Sobrescribe una capa ráster con valores de una capa vectorial. Los nuevos valores se asignan en función del valor del atributo de la característica vectorial superpuesta.

Este algoritmo se deriva de la utilidad GDAL rasterize.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Capa ráster entrante	`INPUT_RASTER`	[ráster]	Capa ráster de entrada
Campo a usar para un valor de quemado Opcional	`FIELD`	[campo de tabla: numérico]	Define el campo de atributo a usar para establecer los valores de los pixels

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Añade quemadura en valores a valores ráster existentes	`ADD`	[booleano] Predeterminado: Falso	Si es Falso, a los píxeles se les asigna el valor del campo seleccionado. Si es Verdadero, el valor del campo seleccionado se agrega al valor de la capa ráster de entrada.

continúe en la próxima página

Tabla 24.965 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: ""	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Rasterizado	`OUTPUT`	[ráster]	La capa ráster entrante sobreescrita

Código Python

Algoritmo ID: `gdal:rasterize_over`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Rasterizar (sobreescribir con valores fijados)

Sobrescribe partes de una capa ráster con un valor fijo. Los píxeles que se van a sobrescribir se eligen en función de la capa vectorial proporcionada (superpuesta).

Este algoritmo se deriva de la utilidad GDAL rasterize.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Capa ráster entrante	`INPUT_RASTER`	[ráster]	Capa ráster de entrada
Un valor fijado para quemar	`BURN`	[numérico: doble] Predeterminado: 0.0	El valor para quemar

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Añade quemadura en valores a valores ráster existentes	`ADD`	[booleano] Predeterminado: Falso	Si es False, a los píxeles se les asigna el valor fijo. Si es Verdadero, el valor fijo se agrega al valor de la capa ráster de entrada.
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: ""	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Rasterizado	`OUTPUT`	[ráster]	La capa ráster entrante sobreescrita

Código Python

Algoritmo ID: `gdal:rasterize_over_fixed_value`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Rasterizar (vectorial a ráster)

Convierte geometrías vectoriales (puntos, líneas y polígonos) en una imagen ráster.

Este algoritmo se deriva de la utilidad GDAL rasterize.

Menú predeterminado: *Raster ► Conversion*

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Campo a usar para un valor de quemado Opcional	`FIELD`	[campo de tabla: numérico]	Define el atributo de campo a partir del cual se escogeran los atributos de los pixeles

continúe en la próxima página

Tabla 24.970 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Un valor fijado para quemar Opcional	`BURN`	[numérico: doble] Predeterminado: 0.0	Un valor fijo para quemar para todas las entidades en una banda.
Valor de quemado extraído de los valores «Z» del objeto espacial	`USE_Z`	[booleano] Predeterminado: Falso	Indica que se debe extraer un valor de quemado de los valores «Z» de la función. Trabaja con puntos y líneas (interpolación lineal a lo largo de cada segmento). Para polígonos, funciona correctamente solo si son planos (mismo valor Z para todos los vértices)
Unidades de tamaño de ráster saliente	`UNITS`	[enumeración] Predeterminado: 0	Unidades a usar al definir la resolución/tamaño del ráster saliente. Uno de: • 0 — Pixeles • 1 — Unidades georreferenciadas
Resolución de anchura/horizontal	`WIDTH`	[numérico: doble] Predeterminado: 0.0	Establece el ancho (si las unidades de tamaño son «Píxeles») o la resolución horizontal (si las unidades de tamaño son «Unidades georreferenciadas») del ráster de salida. Valor mínimo: 0.0.
Resolución de altura/vertical	`HEIGHT`	[numérico: doble] Predeterminado: 0.0	Establece la altura (si las unidades de tamaño son «Píxeles») o la resolución vertical (si las unidades de tamaño son «Unidades georreferenciadas») del ráster de salida.
Extensión de salida Opcional	`EXTENT`	[extensión]	Extensión de la capa ráster de salida. Si no se especifica la extensión, se utilizará la extensión mínima que cubre la capa(s) de referencia seleccionadas. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Asignar un valor NoData especificado a las bandas de salida Opcional	`NODATA`	[numérico: doble] Predeterminado: 0.0	Asigna un valor NoData especificado a las bandas de salida

continúe en la próxima página

Tabla 24.970 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Rasterizado	`OUTPUT`	[ráster] Predeterinado: `[Save to temporary file]`	Especificación de la capa raster de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo… Para `Save to File`, se debe especificar el formato de salida. Se admiten todos los formatos de trama GDAL. Para `Save to a Temporary File` se utilizará el formato de trama predeterminado de QGIS.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: ""	Para agregar una o más opciones de creación que controlen el ráster a crear (colores, tamaño de bloque, compresión de archivos …). Para mayor comodidad, puede confiar en perfiles predefinidos (consultar *GDAL driver options section*). Proceso por lotes y Diseñador de modelos: separe varias opciones con un carácter de tubo (|).
Tipo de datos salientes	`DATA_TYPE`	[enumeración] Predeterminado: 5	Define el formato del archivo ráster saliente Opciones: • 0 — Byte (entero de ocho bits sin signo (quint8)) • 1 — Int16 (Entero con signo de dieciséis bits (qint16)) • 2 — UInt16 (Entero sin signo de dieciséis bits (quint16)) • 3 — UInt32 (Entero sin signo de treinta y dos bits (quint32)) • 4 — Int32 (Entero con signo de treinta y dos bits (qint32)) • 5 — Float32 (Punto flotante de treinta y dos bits (float)) • 6 — Float64 (Punto flotante de sesenta y cuatro bits (doble)) • 7 — CInt16 (Int16 complejo) • 8 — CInt32 (Int32 complejo) • 9 — CFloat32 (Float32 complejo) • 10 — CFloat64 (Complejo Float64) • 11 — Int8 (entero con signo de ocho bits (qint8)) Las opciones disponibles dependen de la versión de GDAL compilada con QGIS (vea l menú *Ayuda* ► *Sobre*)
Preinicialice la imagen de salida con valor Opcional	`INIT`	[numérico: doble]	Preinicializa las bandas de imagen de salida con este valor. No se marca como valor NoData en el archivo de salida. Se utiliza el mismo valor en todas las bandas.

continúe en la próxima página

Tabla 24.971 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Invertir rasterización	`INVERT`	[booleano] Predeterminado: Falso	Graba el valor de quemado fijo o el valor de quemado asociado con la primera entidad en todas las partes de la imagen que no están dentro del polígono proporcionado.
Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: ""	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Rasterizado	`OUTPUT`	[ráster]	Capa ráster saliente

Código Python

Algoritmo ID: `gdal:rasterize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.2.7 Geoprocesamiento vectorial

Buffers vectoriales

Crea buffers alrededor de entidades de una capa vectorial.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	La capa vectorial entrante
Nombre de la columna geometría	`GEOMETRY`	[cadena de texto] Predeterminado: "geometry"	El nombre de la columna de geometría de la capa entrante a usar
Distancia de buffer	`DISTANCE`	[numérico: doble] Predeterminado: 10.0	Mínimo: 0.0

continúe en la próxima página

Tabla 24.973 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Disolver por atributo Opcional	`FIELD`	[campo de tabla: cualquiera] Predeterminado: No establecido	Campo a usar par disolver
Resultados de disolución	`DISSOLVE`	[booleano] Predeterminado: Falso	Si se establece, el resultado se disuelve. Si no se establece ningún campo para la disolución, todos los búferes se disuelven en una sola característica.
Produce una entidad para cada geometría en cualquier tipo de colección de geometría en el archivo de origen	`EXPLODE_COLLEC`	[booleano] Predeterminado: Falso	
Buffer	`OUTPUT`	[vectorial: poligonal] Predeterinado: `[Save to temporary file]`	Especifique la capa del búfer de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: “” (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Buffer	`OUTPUT`	[vectorial: poligonal]	La capa bufer saliente

Código Python

Algoritmo ID: `gdal:buffervectors`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cortar vectorial por extensión

Corta cualquier archivo vectorial compatible con OGR a una extensión dada.

Este algoritmo se deriva de la utilidad GDAL ogr2ogr.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	La capa vectorial entrante
Cortar extensión	`EXTENT`	[extensión]	Define el recuadro delimitador a usar para el archivo saliente vectorial. Tiene que ser definido en las coordenadas del SRC destinatario. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Cortado (extensión)	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especifique la capa de salida (recortada). *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: “” (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cortado (extensión)	`OUTPUT`	[la misma que la entrada]	La capa saliente (cortada). El formato predeterminado es «archivo de forma ESRI».

Código Python

Algoritmo ID: `gdal:clipvectorbyextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Cortar vectorial por capa de máscara

Corta cualquier capa vectorial compatible OGR por una capa poligonal de máscara.

Este algoritmo se deriva de la utilidad GDAL ogr2ogr.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	La capa vectorial entrante
Capa de máscara	`MASK`	[vectorial: poligonal]	Capa a usar como extensión de corte para la capa vectorial entrante.
Cortado (mascara)	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	La capa de salida (enmascarada). *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: "" (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Cortado (mascara)	`OUTPUT`	[la misma que la entrada]	La capa saliente (mascarada). El formato predeterminado es «Archivo de forma ESRI».

Código Python

ID Algoritmo: `gdal:clipvectorbypolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Disolver

Disolver (combinar) geometrías que tengan el mismo valor para un atributo/campo dado. Las geometrías de salida son multiparte.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	La capa entrante a disolver
Disolver campo Opcional	`FIELD`	[campo de tabla: cualquiera]	El campo de la capa entrante a usar para la disolución
Nombre de la columna geometría	`GEOMETRY`	[cadena de texto] Predeterminado: “geometry”	El nombre de la columna de geometría de la capa entrante a usar para la disolución.
Disuelto	`OUTPUT`	[la misma que la entrada] Predeterinado: `[Save to temporary file]`	Especifica la capa de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Produce una entidad para cada geometría en cualquier tipo de colección de geometría en el archivo de origen	`EXPLODE_COLLEC`	[booleano] Predeterminado: Falso	Producir una entidad para cada geometría en cualquier tipo de colección de geometría en el archivo fuente
Mantener atributos de entrada	`KEEP_ATTRIBUTE`	[booleano] Predeterminado: Falso	Mantener todos los atributos de la capa entrante
Contar entidades disueltas	`COUNT_FEATURES`	[booleano] Predeterminado: Falso	Contar las entidades disueltas e incluirlas en la capa saliente.
Calcular área y perímetro de entidades disueltas	`COMPUTE_AREA`	[booleano] Predeterminado: Falso	Calcular el área y el perímetro de los objetos espaciales disueltos e incluirlos en la capa de salida.
Calcular min/max/sum/media para atributo	`COMPUTE_STATIS`	[booleano] Predeterminado: Falso	Calcular estadísticas (min, max, suma y media) para los atributos nnuméricos especificados e incluirlas en la capa saliente
Atributo numérico donde calcular estadísticas Opcional	`STATISTICS_ATT`	[campo de tabla: numérico]	El atributo numérico para calcular las estadísticas
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: "" (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Disuelto	`OUTPUT`	[la misma que la entrada]	La capa de geometría multiparte de salida (con geometrías disueltas)

Código Python

Algoritmo ID: `gdal:dissolve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Desplazamiento de curva

Compensa las líneas por una distancia especificada. Las distancias positivas compensarán las líneas a la izquierda, y las distancias negativas las compensarán a la derecha.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	La capa lineal entrante
Nombre de la columna geometría	`GEOMETRY`	[cadena de texto] Predeterminado: “geometry”	El nombre de la columna de geometría de la capa entrante a usar
Distancia de desplazamiento (lado izquierdo: positiva, lado derecho: negativa)	`DISTANCE`	[numérico: doble] Predeterminado: 10.0	
Desplazamiento de curva	`OUTPUT`	[vectorial: lineal] Predeterinado: `[Save to temporary file]`	Especifique la capa de la línea de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: “” (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Desplazamiento de curva	`OUTPUT`	[vectorial: lineal]	La capa con la curva desplazada saliente

Código Python

Algoritmo ID: `gdal:offsetcurve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Bufer unilateral

Crea un bufer en un lado (derecho o izquierdo) de las líneas en una capa vectorial lineal.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	La capa lineal entrante
Nombre de la columna geometría	`GEOMETRY`	[cadena de texto] Predeterminado: “geometry”	El nombre de la columna de geometría de la capa entrante a usar
Distancia de buffer	`DISTANCE`	[numérico: doble] Predeterminado: 10.0	
Bufer lateral	`BUFFER_SIDE`	[enumeración] Predeterminado: 0	Una de: • 0 — Derecha • 1 — Izquierda
Disolver por atributo Opcional	`FIELD`	[campo de tabla: cualquiera] Predeterminado: No establecido	Campo a usar par disolver
Disolver todos los resultados	`DISSOLVE`	[booleano] Predeterminado: Falso	**Si se establece, el resultado se disuelve** Si no se establece ningún campo para disolver, todos los búferes se disuelven en una entidad.
Produce una entidad para cada geometría en cualquier tipo de colección de geometría en el archivo de origen	`EXPLODE_COLLEC`	[booleano] Predeterminado: Falso	

continúe en la próxima página

Tabla 24.988 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Bufer unilateral	`OUTPUT`	[vectorial: poligonal] Predeterinado: `[Save to temporary file]`	Especifique la capa del búfer de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: "" (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Bufer unilateral	`OUTPUT`	[vectorial: poligonal]	La capa bufer saliente

Código Python

Algoritmo ID: `gdal:onesidebuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

Puntos a lo largo de líneas

Genera un punto en cada línea de una capa de vector de línea a una distancia desde el inicio. La distancia se proporciona como una fracción de la longitud de la línea.

Parametros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: lineal]	La capa lineal entrante
Nombre de la columna geometría	`GEOMETRY`	[cadena de texto] Predeterminado: “geometry”	El nombre de la columna de geometría de la capa entrante a usar
Distancia desde el inicio de la línea representada como una fracción de la longitud de la línea	`DISTANCE`	[numérico: doble] Predeterminado: 0.5	
Puntos a lo largo de las líneas	`OUTPUT`	[vector: punto] Predeterinado: `[Save to temporary file]`	Especifique la capa de puntos de salida. *Una de*: • Guardar a un Archivo Temporal • Guardar en archivo…

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: “” (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Puntos a lo largo de línea	`OUTPUT`	[vector: punto]	La capa puntual saliente

Código Python

Algoritmo ID: `gdal:pointsalonglines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra cuando pasa el cursor sobre el algoritmo en la caja de herramientas de procesos. El *diccionario de parámetros* proporciona los NOMBRES y valores de los parámetros. Consulte *Utilizar algoritmos de procesamiento desde la consola* para obtener detalles sobre cómo ejecutar algoritmos de procesamiento desde la consola de Python.

24.2.8 Vector misceláneo

Construir vectorial virtual

Crea una capa vectorial virtual que contiene un conjunto de capas vectoriales.

Este algoritmo es especialmente útil en caso de que otro algoritmo necesite varias capas pero acepte solo una `vrt` en la que se especifican las capas.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Fuentes de datos entrantes	`INPUT`	[vectorial: cualquiera] [lista]	Seleccione las capas vectoriales que desea usar para construir el vector virtual
Crear VRT «unido»	`UNIONED`	[booleano] Predeterminado: False	Marque si quiere unir todos los vectoriales en un único archivo `vrt`
Vectorial Virtual	`OUTPUT`	[la misma que la entrada] Predeterminado: `[Save to temporary file]`	Especifique la capa de salida que contiene sólo los duplicados. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo…

Salidas

Etiqueta	Nombre	Tipo	Descripción
Vectorial Virtual	`OUTPUT`	[vectorial: cualquiera]	La capa vectorial virtual saliente hecha de las fuentes elegidas

Código Python

ID Algoritmo: `gdal:buildvirtualvector`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Ejecutar SQL

Ejecuta una consulta simple o compleja con sintaxis SQL en la capa de origen. El resultado de la consulta se agregará como una nueva capa.

Este algoritmo se deriva de la utilidad GDAL ogr2ogr.

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial de entrada compatible OGR
Expresión SQL	`SQL`	[cadena de texto]	Define la consulta SQL, por ejemplo, `SELECT * FROM my_table WHERE name is not null`.
Dialecto SQL	`DIALECT`	[enumeración] Predeterminado: 0	Dialecto SQL a utilizar. Uno de: • 0 — Ninguno • 1 — OGR SQL • 2 — SQLite
Resultado SQL	`OUTPUT`	[vectorial: cualquiera]	Especificación de la capa de salida. *Una de*: • Guardar en Archivo Temporal • Guardar en archivo… Para `Guardar en archivo`, se debe especificar el formato de salida. Se admiten todos los formatos vectoriales GDAL. Para `Guardar en un archivo temporal` se utilizará el formato de capa de vector de salida predeterminado.

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
Opciones de creación adicional Opcional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[cadena de texto] Predeterminado: "" (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Etiqueta	Nombre	Tipo	Descripción
Resultado SQL	`OUTPUT`	[vectorial: cualquiera]	Capa vectorial creada por la consulta

Código Python

Algoritmo ID: `gdal:executesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Exportar a PostgreSQL (conexiones disponibles)

Importa capas vectoriales dentro de una base de datos PostgreSQL sobre la base de una conexión disponible. La conexión tiene que *ser definida apropiadamente* de antemano. Tenga en cuenta que las casillas de verificación "Guardar nombre de usuario" y "Guardar contraseña" están activadas. Entonces puedes usar el algoritmo.

Este algoritmo se deriva de la utilidad GDAL ogr2ogr.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Base de datos (nombre de conexión)	`DATABASE`	[cadena de texto]	La base de datos de PostgreSQL a la que conectar
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial compatible OGR a exportar a la base de datos
Codificación de forma Opcional	`SHAPE_ENCODING`	[cadena de texto] Predeterminado: ""	Establece la codificación a aplicar a los datos

continúe en la próxima página

Tabla 24.999 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Tipo de geometría saliente	GTYPE	[enumeración] Predeterminado: 0	Define el tipo de geometría saliente. Una de: • 0 — • 1 — NINGUNA • 2 — GEOMETRIA • 3 — PUNTO • 4 — CADENA LINEAL • 5 — POLIGONAL • 6 — COLECCIÓN DE GEOMETRÍAS • 7 — MULTIPUNTO • 8 — MULTIPOLÍGONO • 9 — MULTI CADENA DE LÍNEAS • 10 — CIRCULARSTRING • 11 — COUMPOUNDCURVE • 12 — CURVEPOLYGON • 13 — MULTICURVE • 14 — MULTISURFACE • 15 — CONVERT_TO_LINEAR • 16 — CONVERT_TO_CURVE
Asignar un SRC saliente Opcional	A_SRS	[src] Predeterminado: No establecido	Define el SRC saliente de la tabla de base de datos
Refporyectar a este SRC en la salida Opcional	T_SRS	[src] Predeterminado: No establecido	Reproyectar/transformar a este SRC en salida
Suplantar SRC de fuente Opcional	S_SRS	[src] Predeterminado: No establecido	Suplanta el SRC de la capa entrante
Esquema (nombre de esquema) Opcional	SCHEMA	[cadena de texto] Predeterminado: “public”	Define el esquema para la tabla de base de datos
Tabla para la que exportar (dejar en blanco para usar el nombre de capa) Opcional	TABLE	[cadena de texto] Predeterminado: “”	Define un nombre para la tabla que se importará a la base de datos. Por defecto, el nombre de la tabla es el nombre del archivo de vector de entrada.
Clave primaria (nuevo campo) Opcional	PK	[cadena de texto] Predeterminado: “id”	Define qué campo de atributo será la clave principal de la tabla de la base de datos
Clave principal (campo existente, utilizado si la opción anterior se deja vacía) Opcional	PRIMARY_KEY	[campo de tabla: cualquiera] Predeterminado: No establecido	Define qué campo de atributo en la capa exportada será la clave principal de la tabla de la base de datos
Nombre de la columna geometría Opcional	GEOCOLUMN	[cadena de texto] Predeterminado: “geom”	Define en qué campo de atributo de la base de datos estará la información de geometría

continúe en la próxima página

Tabla 24.999 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Dimensiones vectoriales Opcional	`DIM`	[enumeración] Predeterminado: 0 (2D)	Define las dimensiones de las coordenadas de los datos vectoriales importados. Una de: • 0 — 2 (`XY`) • 1 — 3 (`XYZ`) • 2 — 4 (`XYZM`)
Tolerancia de distancia para simplificación Opcional	`SIMPLIFY`	[cadena de texto] Predeterminado: ""	Define una tolerancia de distancia para la simplificación de las geometrías vectoriales que se importarán. Por defecto no hay simplificación.
Distancia máxima entre 2 nodos (densificación) Opcional	`SEGMENTIZE`	[cadena de texto] Predeterminado: ""	La distancia máxima entre dos nodos. Se utiliza para crear puntos intermedios. Por defecto no hay densificación.
Seleccionar entidades por extensión (definida en SRC de capa entrante) Opcional	`SPAT`	[extensión] Predeterminado: No establecido	Puede seleccionar entidades de una extensión determinada que estarán en la tabla de salida. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Cortar la capa entrante usando la extensión de encima (rectángulo)	`CLIP`	[booleano] Predeterminado: False	La capa de entrada será recortada por la extensión que definiste antes.
Seleccionar entidades usando una declaración SQL «WHERE» (Ej: column=»value») Opcional	`WHERE`	[cadena de texto] Predeterminado: ""	Define con una sentencia SQL «WHERE» que entidades deben ser seleccionadas de la capa entrante
Agrupar N entidades mediante transacción (Predeterminado: 2000) Opcional	`GT`	[cadena de texto] Predeterminado: ""	Puede agrupar las entidades de entrada en transacciones donde N define el tamaño. Por defecto, N limita el tamaño de la transacción a 20000 entidades.

continúe en la próxima página

Tabla 24.999 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Sobreescribir tabla existente	OVERWRITE	[booleano] Predeterminado: Verdadero	Si hay una tabla con el mismo nombre en la base de datos y si esta opción se establece en Verdadero, la tabla se sobrescribirá.
Agregar a tabla existente	APPEND	[booleano] Predeterminado: False	Si está marcado / Verdadero, los datos vectoriales se agregarán a una tabla existente. Los nuevos campos que se encuentran en la capa de entrada se ignoran. De forma predeterminada, se creará una nueva tabla.
Agregar y añadir nuevos campos a tabla existente	ADDFIELDS	[booleano] Predeterminado: False	Si se activa, los datos vectoriales se agregarán a una tabla existente, no se creará una nueva tabla. Los nuevos campos que se encuentran en la capa de entrada se agregan a la tabla. De forma predeterminada, se creará una nueva tabla.
No blanquear nombres de columnas/tabla	LAUNDER	[booleano] Predeterminado: False	Con esta opción marcada, puede evitar el comportamiento predeterminado (convertir los nombres de las columnas a minúsculas, eliminar espacios y otros caracteres no válidos).
No crear Índice Espacial	INDEX	[booleano] Predeterminado: False	Impide que se cree un índice espacial para la tabla de salida. De forma predeterminada, se agrega un índice espacial.
Continuar tras un fallo, saltando la entidad errónea	SKIPFAILURES	[booleano] Predeterminado: False	Continuar después de un fallo, omitiendo la función fallida.
Validar geometrías basadas en la especificación Simple Features	MAKEVALID	[booleano] Predeterminado: False	Aplica operaciones de limpieza a las geometrías para garantizar que son válidas con respecto a las reglas de la especificación Simple Features.
Ascender a multiparte	PROMOTETOMULTI	[booleano] Predeterminado: Verdadero	Convierte el tipo de geometría de entidades a multiparte en la tabla de salida
Mantener anchura y precisión de los atributos de entrada	PRECISION	[booleano] Predeterminado: Verdadero	Evita modificar los atributos de la columna para cumplir con los datos de entrada
Opciones de creación adicional Opcional	CREATION_OPTIO (for QGIS <= 3.42, this was OPTIONS)	[cadena de texto] Predeterminado: "" (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Este algoritmo no conlleva una salida.

Código Python

Algoritmo ID: `gdal:importvectorintopostgisdatabaseavailableconnections`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Exportar a PostgreSQL (nueva conexión)

Imports vector layers inside a PostgreSQL database. A new connection to the PostgreSQL database must be created.

Este algoritmo se deriva de la utilidad GDAL ogr2ogr.

Parámetros

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vectorial: cualquiera]	Capa vectorial compatible OGR a exportar a la base de datos
Codificación de forma Opcional	`SHAPE_ENCODING`	[cadena de texto] Predeterminado: “”	Establece la codificación a aplicar a los datos
Tipo de geometría saliente	`GTYPE`	[enumeración] Predeterminado: 0	Define el tipo de geometría saliente. Una de: • 0 — • 1 — NINGUNA • 2 — GEOMETRIA • 3 — PUNTO • 4 — CADENA LINEAL • 5 — POLIGONAL • 6 — COLECCIÓN DE GEOMETRÍAS • 7 — MULTIPUNTO • 8 — MULTIPOLÍGONO • 9 — MULTI CADENA DE LÍNEAS • 10 — CIRCULARSTRING • 11 — COUMPOUNDCURVE • 12 — CURVEPOLYGON • 13 — MULTICURVE • 14 — MULTISURFACE • 15 — CONVERT_TO_LINEAR • 16 — CONVERT_TO_CURVE
Asignar un SRC saliente Opcional	`A_SRS`	[src] Predeterminado: No establecido	Define el SRC saliente de la tabla de base de datos

continúe en la próxima página

Tabla 24.1000 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Refporyectar a este SRC en la salida Opcional	`T_SRS`	[src] Predeterminado: No establecido	Reproyectar/transformar a este SRC en salida
Suplantar SRC de fuente Opcional	`S_SRS`	[src] Predeterminado: No establecido	Suplanta el SRC de la capa entrante
Host Opcional	`HOST`	[cadena de texto] Predeterminado: "localhost"	Nombre del host de la base de datos
Puerto Opcional	`PORT`	[cadena de texto] Predeterminado: "5432"	Número de puerto en el que escucha el servidor de base de datos PostgreSQL
Nombre de usuario Opcional	`USER`	[cadena de texto] Predeterminado: ""	Nombre de usuario a registrar en la base de datos
Nombre de la base de datos Opcional	`DBNAME`	[cadena de texto] Predeterminado: ""	Nombre de la base de datos
Contraseña Opcional	`PASSWORD`	[cadena de texto] Predeterminado: ""	Contraseña usada con el nombre de usuario para conectar a la base de datos
Esquema (nombre de esquema) Opcional	`SCHEMA`	[cadena de texto] Predeterminado: "public"	Define el esquema para la tabla de base de datos
Nombre de tabla, dejar vacío para usar el nombre entrante Opcional	`TABLE`	[cadena de texto] Predeterminado: ""	Define un nombre para la tabla que se importará a la base de datos. Por defecto, el nombre de la tabla es el nombre del archivo de vector de entrada.
Clave primaria (nuevo campo) Opcional	`PK`	[cadena de texto] Predeterminado: "id"	Define qué campo de atributo será la clave principal de la tabla de la base de datos
Clave principal (campo existente, utilizado si la opción anterior se deja vacía) Opcional	`PRIMARY_KEY`	[campo de tabla: cualquiera] Predeterminado: No establecido	Define qué campo de atributo en la capa exportada será la clave principal de la tabla de la base de datos
Nombre de la columna geometría Opcional	`GEOCOLUMN`	[cadena de texto] Predeterminado: "geom"	Define en que atributo de campo almacenar la información de geometría
Dimensiones vectoriales Opcional	`DIM`	[enumeración] Predeterminado: 0 (2D)	Define las dimensiones de las coordenadas de los datos vectoriales importados. Una de: • 0 — 2 (`XY`) • 1 — 3 (`XYZ`) • 2 — 4 (`XYZM`)
Tolerancia de distancia para simplificación Opcional	`SIMPLIFY`	[cadena de texto] Predeterminado: ""	Define una tolerancia de distancia para la simplificación de las geometrías vectoriales que se importarán. Por defecto no hay simplificación.

continúe en la próxima página

Tabla 24.1000 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Distancia máxima entre 2 nodos (densificación) Opcional	`SEGMENTIZE`	[cadena de texto] Predeterminado: ""	La distancia máxima entre dos nodos. Se utiliza para crear puntos intermedios. Por defecto no hay densificación.
Seleccionar entidades por extensión (definida en SRC de capa entrante) Opcional	`SPAT`	[extensión] Predeterminado: No establecido	Puede seleccionar entidades de una extensión determinada que estarán en la tabla de salida. Los métodos disponibles son: • Calcular desde capa…: usa la extensión de una capa cargada en el proyecto actual • Calcular desde el mapa de diseño…: usa la extensión de un es extent of a *elemento del mapa de diseño* en el proyecto activo • Calcular desde el marcador…: utiliza la extensión de un marcador <sec_bookmarks> guardado • Usar la extensión del lienzo del mapa • Dibujar sobre el lienzo: pulsar y arrastrar un rectángulo que delimite el área a tener en cuenta • Ingresar las coordenadas como `xmin, xmax, ymin, ymax`
Cortar la capa entrante usando la extensión de encima (rectángulo)	`CLIP`	[booleano] Predeterminado: False	La capa de entrada será recortada por la extensión que definiste antes.
Campos a incluir (dejar vacío para usar todos los campos) Opcional	`FIELDS`	[cadena] [lista] Predeterminado: []	Define los campos que se guardarán del archivo vectorial importado. Si no se selecciona ninguno, se importan todos los campos.
Seleccionar entidades usando una declaración SQL «WHERE» (Ej: column=»value») Opcional	`WHERE`	[cadena de texto] Predeterminado: ""	Define con una declaración SQL «WHERE» qué entidades deben seleccionarse para la tabla de salida
Agrupar N entidades mediante transacción (Predeterminado: 2000) Opcional	`GT`	[cadena de texto] Predeterminado: ""	Puede agrupar las entidades de entrada en transacciones donde N define el tamaño. Por defecto, N limita el tamaño de la transacción a 20000 entidades.
Sobreescribir tabla existente	`OVERWRITE`	[booleano] Predeterminado: Verdadero	Si hay una tabla con el mismo nombre en la base de datos y si esta opción se establece en Verdadero, la tabla se sobrescribirá.

continúe en la próxima página

Tabla 24.1000 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Agregar a tabla existente	APPEND	[booleano] Predeterminado: False	Si está marcado / Verdadero, los datos vectoriales se agregarán a una tabla existente. Los nuevos campos que se encuentran en la capa de entrada se ignoran. De forma predeterminada, se creará una nueva tabla.
Agregar y añadir nuevos campos a tabla existente	ADDFIELDS	[booleano] Predeterminado: False	Si está activado, los datos vectoriales se agregarán a una tabla existente, no se creará una nueva tabla. Los nuevos campos que se encuentran en la capa de entrada se agregan a la tabla. De forma predeterminada, se creará una nueva tabla.
No blanquear nombres de columnas/tabla	LAUNDER	[booleano] Predeterminado: False	Con esta opción marcada, puede evitar el comportamiento predeterminado (convertir los nombres de las columnas a minúsculas, eliminar espacios y otros caracteres no válidos).
No crear Índice Espacial	INDEX	[booleano] Predeterminado: False	Impide que se cree un índice espacial para la tabla de salida. De forma predeterminada, se agrega un índice espacial.
Continuar tras un fallo, saltando la entidad errónea	SKIPFAILURES	[booleano] Predeterminado: False	Continuar después de un fallo, omitiendo la función fallida.
Validar geometrías basadas en la especificación Simple Features	MAKEVALID	[booleano] Predeterminado: False	Aplica operaciones de limpieza a las geometrías para garantizar que son válidas con respecto a las reglas de la especificación Simple Features.
Ascender a multiparte	PROMOTETOMULTI	[booleano] Predeterminado: Verdadero	Convierte el tipo de geometría de entidades a multiparte en la tabla de salida
Mantener anchura y precisión de los atributos de entrada	PRECISION	[booleano] Predeterminado: Verdadero	Evita modificar los atributos de la columna para cumplir con los datos de entrada
Opciones de creación adicional Opcional	CREATION_OPTIO (for QGIS <= 3.42, this was OPTIONS)	[cadena de texto] Predeterminado: "" (sin opciones adicionales)	Opciones adicionales de creación GDAL.

Salidas

Este algoritmo no conlleva una salida.

Código Python

Algoritmo ID: `gdal:importvectorintopostgisdatabasenewconnection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Información vectorial

Crea un archivo de información que enumera información sobre una fuente de datos compatible con OGR. La salida se mostrará en una ventana «Resultado» y puede escribirse en un archivo HTML. La información incluye el tipo de geometría, el recuento de características, la extensión espacial, la información de proyección, la lista de atributos y el tipo relacionado, la lista de relaciones y dominios de campo, y mucho más.

Este algoritmo se deriva de la utilidad GDAL ogrinfo.

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	`INPUT`	[vector: geometría]	Capa de vector de entrada
Habilitar el listado de todas las capas del conjunto de datos	`ALL_LAYERS`	[booleano] Predeterminado: False	Si se marca, QGIS mostrará la información de todas las capas del conjunto de datos de entrada en lugar de la información de la primera capa solamente. Conveniente cuando se utiliza como entrada un formato multicapa (GeoPackage, GML, KML, SpatiaLite, SQLite, …).
Solo salida de resumen	`SUMMARY_ONLY`	[booleano] Predeterminado: Verdadero	Suprima el listado de objetos espaciales individuales y muestre sólo información resumida como proyección, esquema, recuento y extensiones de objetos espaciales.
Suprimir información de metadatos	`NO_METADATA`	[booleano] Predeterminado: False	Suprimir la impresión de metadatos. Algunos conjuntos de datos pueden contener muchas cadenas de metadatos.
Información de la capa	`OUTPUT`	[html] Predeterminado: `[Save to temporary file]`	Especifique el archivo HTML de salida que incluye la información del archivo. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo… Si no se define un archivo HTML, la salida se escribirá en un archivo temporal

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
**Parámetros adicionales de la línea de comandos Opcional	EXTRA	[cadena de texto] Predeterminado: "" (sin opciones adicionales)	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Información de la capa	OUTPUT	[html]	El archivo HTML de salida que incluye la información del archivo.

Código Python

Algoritmo ID: `gdal:ogrinfo`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

Información vectorial (JSON)

Crea un archivo de información que enumera información sobre una fuente de datos compatible con OGR. La salida se mostrará en una ventana «Resultado» y puede escribirse en un archivo `.JSON`. La información incluye el tipo de geometría, el recuento de características, la extensión espacial, la información de proyección, la lista de atributos y el tipo relacionado, la lista de relaciones y dominios de campo, y mucho más.

Este algoritmo se deriva de la utilidad GDAL ogrinfo.

> **Atención**
>
> Running this algorithm requires QGIS installed with GDAL >= 3.7 (see *Help* ► *About* menu).

Parámetros

Parámetros básicos

Etiqueta	Nombre	Tipo	Descripción
Capa de entrada	INPUT	[vector: geometría]	Capa de vector de entrada

continúe en la próxima página

Tabla 24.1004 – proviene de la página anterior

Etiqueta	Nombre	Tipo	Descripción
Habilitar el listado de todas las capas del conjunto de datos	`ALL_LAYERS`	[booleano] Predeterminado: False	Si se marca, QGIS mostrará la información de todas las capas del conjunto de datos de entrada en lugar de la información de la primera capa solamente. Conveniente cuando se utiliza como entrada un formato multicapa (GeoPackage, GML, KML, SpatiaLite, SQLite, …).
Habilitar listado de objetos espaciales	`FEATURES`	[booleano] Predeterminado: False	
Suprimir información de metadatos	`NO_METADATA`	[booleano] Predeterminado: False	Suprimir la impresión de metadatos. Algunos conjuntos de datos pueden contener muchas cadenas de metadatos.
Información de la capa	`OUTPUT`	[archivo] Predeterminado: `[Save to temporary file]`	Especifique el archivo JSON de salida que incluye la información del archivo. *Uno de*: • Guardar en Archivo Temporal • Guardar en archivo… Si no se define ningún archivo JSON, la salida se escribirá en un archivo temporal

Parámetros avanzados

Etiqueta	Nombre	Tipo	Descripción
**Parámetros adicionales de la línea de comandos Opcional	`EXTRA`	[cadena de texto] Predeterminado: “” (sin opciones adicionales)	Añadir opciones extra de línea de comando GDAL

Salidas

Etiqueta	Nombre	Tipo	Descripción
Información de la capa	`OUTPUT`	[archivo]	El archivo de salida *.JSON* que incluye la información del archivo.

Código Python

ID del algoritmo: `gdal:ogrinfojson`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

El *algoritmo id* se muestra al desplazarse sobre el algoritmo en la Caja de Herramientas de Procesamiento. El *parámetro diccionario* suministra los NOMBREs y valores de los parámetros. Ver consola_procesamiento para detalles sobre como ejecutar algoritmos desde la consola Python.

CAPÍTULO 25

Complementos

25.1 Complementos de QGIS

QGIS ha sido diseñado con una arquitectura de complementos. Esto permite que sea fácil añadir muchas características y funciones nuevas a la aplicación. Muchas de las características de QGIS están en realidad implementadas como complementos.

25.1.1 Complementos base y externos

Los complementos de QGIS se implementan como **Complementos principales** o **Complementos externos**.

Los *Complementos principales* son mantenidos por el Equipo de Desarrollo de QGIS y son automáticamente parte de cada distribución de QGIS. Están escritos en uno de dos idiomas: **C++** o **Python**.

La mayoría de los complementos externos están escritos actualmente en Python. Están almacenados en el repositorio «oficial» de QGIS en https://plugins.qgis.org/plugins/ o en repositorios externos y son mantenidos por sus autores. Para los complementos del repositorio oficial se proporciona documentación detallada sobre su uso, versión mínima de QGIS, página de inicio, autores y otra información importante. Para otros repositorios externos, la documentación puede estar disponible con los propios complementos externos. La documentación de los complementos externos no se incluye en este manual.

Para instalar o activar el complemento, vaya al menú *Complementos* y seleccionar  *Administrar e Instalar complementos…*. los complementos de Python externos instalados se colocan en la carpeta `python/plugins` de la ruta activa *perfil de usuario*.

También se pueden agregar rutas a bibliotecas de complementos personalizados de C ++ en *Configuración* ► *Opciones* ► *Sistema*.

25.1.2 El diálogo de complementos

La pestaña Configuración

En la parte inferior del panel izquierdo, la pestaña *Settings* es el lugar principal donde puede configurar qué plugins pueden mostrarse en su aplicación. Puede utilizar las siguientes opciones:

- *Check for Updates on Startup*. QGIS will automatically check for plugin updates at startup, at most once every 3 days. Whenever an installed plugin has an update available, QGIS will inform you.
- *Mostrar también Complementos Experimentales*. QGIS le mostrará complementos en fases tempranas de desarrollo, que generalmente no son adecuados para su uso en producción. Para estos complementos, puede instalar la versión estable o la experimental, y en cualquier momento cambiar de una a otra.
- *Mostrar también complementos obsoletos*. Estos complementos no suelen estar mantenidos porque tienen funciones de sustitución en QGIS, falta de mantenedores, dependen de funciones que ya no están disponibles en QGIS… Generalmente no son adecuados para su uso en producción y aparecen en gris en la lista de complementos.

De modo predeterminado en la sección guilabel:*Repositorios de Complementos* QGIS le proporciona su repositorio de complementos oficial con la URL `https://plugins.qgis.org/plugins/plugins.xml?qgis=version` (donde `<version>` representa la versión exacta de QGIS que está corriendo). Para añadir repositorios de autores externos, pulse *Añadir…* y rellene el formato *Detalles del Repositorio* con el nombre y la URL. La URL puede ser de tipo de protocolo `http://` o `file://`.

El repositorio predeterminado de QGIS es un repositorio abierto y no necesita ninguna autenticación para acceder a él. Sin embargo, puede implementar su propio repositorio de complementos y requerir una autenticación (autenticación básica, PKI). Puede obtener más información sobre el soporte de autenticación de QGIS en el capítulo *Autenticación*.

Si no desea uno o varios de los repositorios añadidos, puede desactivarlos desde la pestaña Configuración mediante el botón *Editar…*, o eliminarlos completamente con el botón *Borrar*.

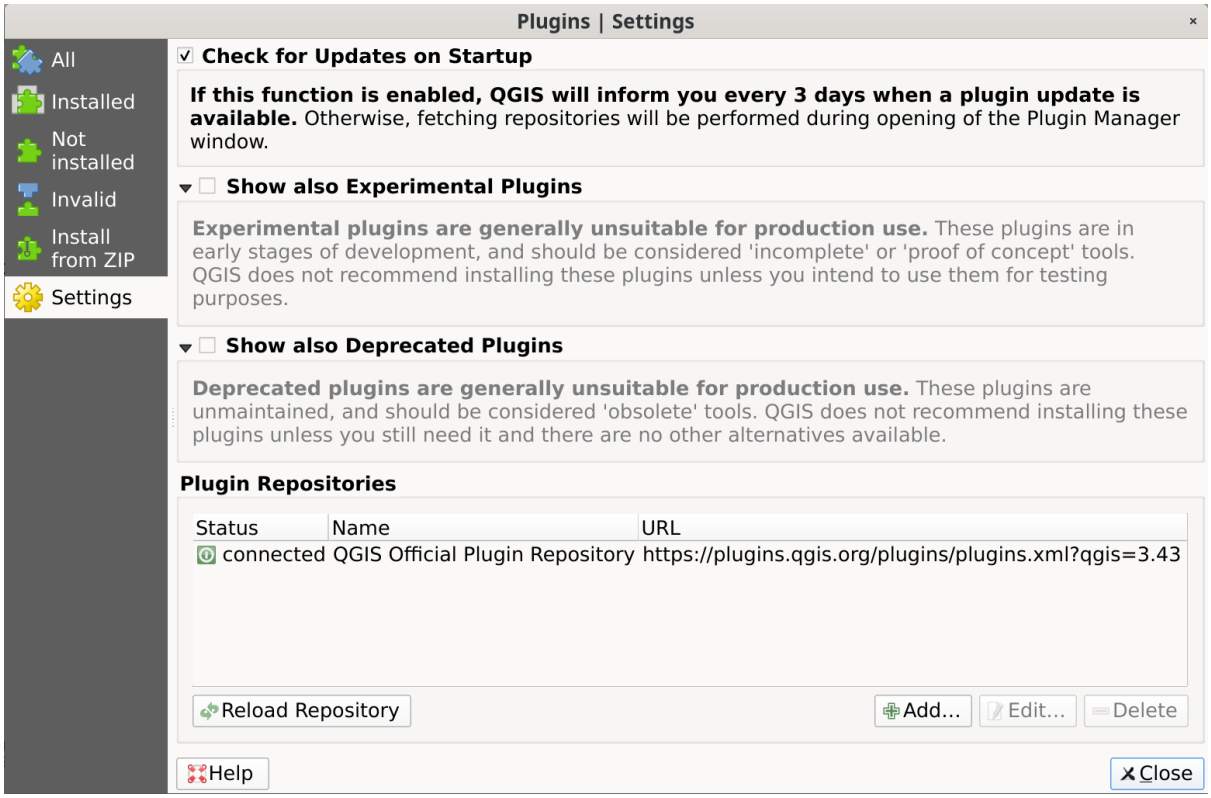


Figura 25.1: La pestaña *Configuración*

Navegar por los complementos

Las pestañas

Las pestañas superiores del cuadro de diálogo *Complementos* le proporcionan listas de complementos en función de su estado de instalación, creación o actualización. Dependiendo de la configuración de los complementos, las pestañas disponibles pueden ser:

- *Todo*: muestra todos los complementos disponibles en los repositorios habilitados.
- *Instalado*: muestra tanto los complementos que has instalado como los complementos núcleo que se instalan por defecto y no puedes desinstalar
- *No instalado*: muestra los complementos desinstalados o aún no instalados en los repositorios habilitados.
- *Nuevo*: muestra los complementos lanzados desde la última *Buscar actualizaciones al iniciar*.
- *Actualizable*: muestra los complementos instalados que han publicado una versión más reciente en el repositorio
- *Inválido*: muestra todos los complementos instalados que están actualmente rotos por cualquier razón (falta de dependencia, errores al cargar, funciones incompatibles con la versión de QGIS…)

En la parte superior de las pestañas, una función *Buscar* le ayuda a encontrar cualquier plugin utilizando información de metadatos (autor, nombre, descripción, etiqueta,…).

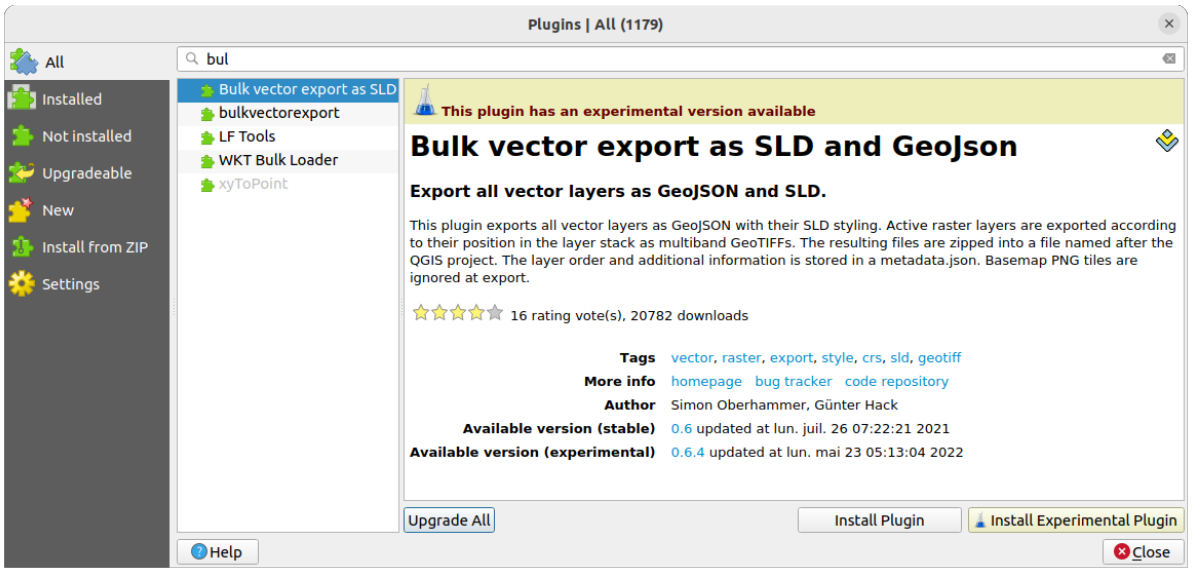


Figura 25.2: Buscar un complemento desde la pestaña *Todo*

Los Complementos

Seleccione un complemento y aparecerán algunos metadatos en el panel derecho:

- información sobre si el complemento es experimental o tiene una versión experimental disponible (si *Mostrar también Complementos experimentales* está marcada)
- resumen y descripción
- calificación por voto(s) (¡puede votar por su complemento preferido!)
- etiquetas
- algunos enlaces útiles a la página de inicio, el rastreador y el repositorio de código
- autor(es)
- versión(es) disponible(s) con enlace a la página de descarga en el repositorio, o ruta a la carpeta local para los complementos instalados

El diálogo *Gestor de complementos* permite interactuar con la última versión de los complementos. Cuando está activado, la versión experimental sólo puede mostrarse si es más reciente que la última versión estable. Dependiendo de la pestaña activa, si el plugin seleccionado está instalado, se le ofrecerán algunas de las siguientes opciones:

- *Instalar*: instala la última versión estable del complemento seleccionado
- *Instalar el complemento experimental*: instala la versión experimental del complemento seleccionado
- *Reinstalar el complemento*: instala la misma versión estable del plugin, por ejemplo, después de que haya fallado al cargarse.
- *Reinstalar complemento experimental*: instala la misma versión experimental del complemento, por ejemplo, después de que no se haya podido cargar.
- *Complemento de actualización*: actualiza el plugin seleccionado a su última versión estable.
- *Actualizar el complemento experimental*: actualiza el complemento seleccionado a su versión experimental
- *Actualizar todo*: actualiza todos los complementos instalados a su versión estable o experimental más reciente (dependiendo de si su versión previamente instalada era estable o experimental).
- *Bajar de categoría complemento*: pasa de la versión experimental del plugin a su versión estable anterior

- *Bajar de categoría complemento Experimental*: pasa de una versión experimental del complemento a su última versión experimental publicada. Esto puede ocurrir cuando se juega con una versión aún no publicada.
- *Desinstalar complemento*: elimina el complemento instalado del perfil del usuario.

Un complemento instalado muestra una casilla a su izquierda. Desactívela para desactivar temporalmente el complemento.

Haga clic con el botón derecho en un complemento de la lista y podrá ordenar la lista de complementos por varios metadatos. El nuevo orden se aplica a todas las pestañas. Las opciones de ordenación son:

- *Ordenar por nombre*
- *Ordenar por descargas*
- *Ordenar por voto*
- *Ordenar por estado*
- *Ordenar por fecha de creación*
- *Ordenar por fecha de actualización*

La pestaña instalar a partir de ZIP

La pestaña *Install from ZIP* proporciona un widget selector de archivos para importar complementos en formato comprimido, por ejemplo, plugins descargados directamente de su repositorio. Admite archivos encriptados.

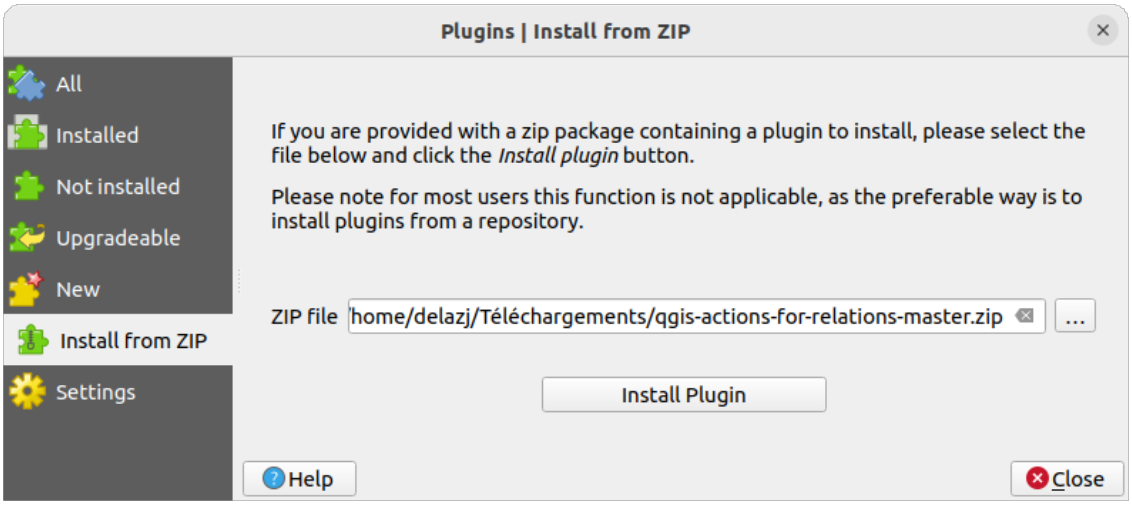


Figura 25.3: La pestaña *Instalar a partir de ZIP*

25.2 Usar complementos del núcleo de QGIS

25.2.1 Complemento de Administración de BBDD

The DB Manager Plugin is intended to be the main tool to integrate and manage spatial database formats supported by QGIS (PostgreSQL, SpatiaLite, GeoPackage, Oracle Spatial, Virtual layers) in one user interface. The DB Manager Plugin provides several features. You can drag layers from the QGIS Browser into the DB Manager, and it will import your layer into your spatial database. You can drag and drop tables between spatial databases and they will get imported.

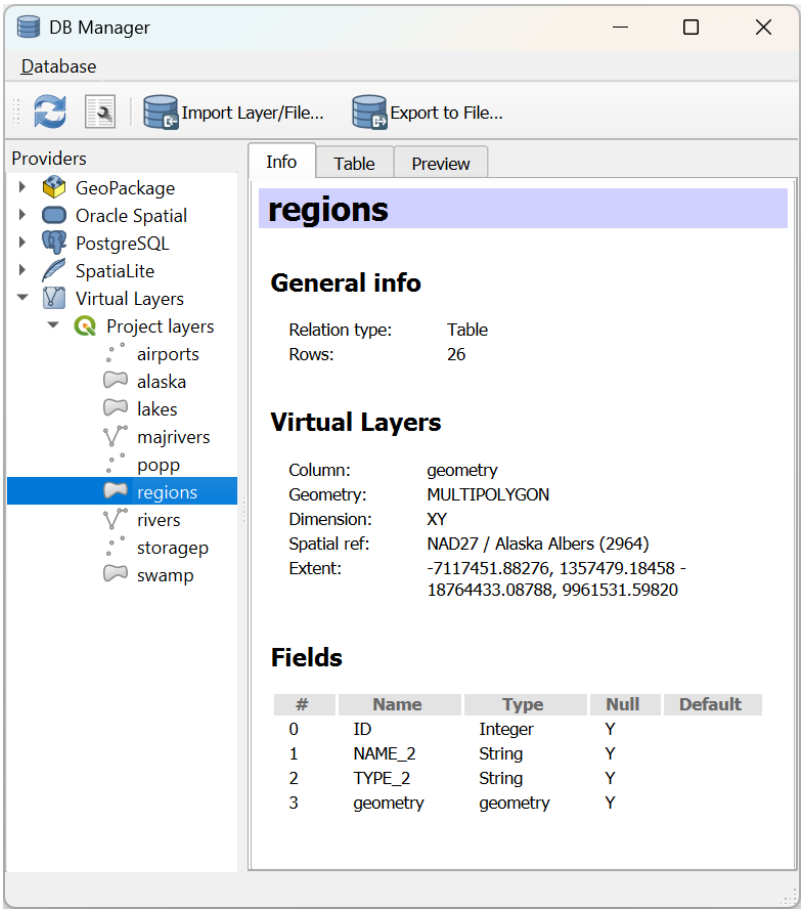


Figura 25.4: Diálogo del complemento administrador de BBDD

The *Database* menu allows you to connect to an existing database, to start the SQL window and to exit the DB Manager Plugin. Once you are connected to an existing database, the menus *Schema* (relevant for DBMSs, such as PostgreSQL) and *Table* will appear.

El menú *Schema* incluye herramientas para crear y borrar (solo si está vacía) esquemas y, si la topología está disponible (p.ej. con la topología PostGIS), para empezar una *TopoViewer*.

El menú *Table* le permite crear y editar tablas y eliminar tablas y vistas. También es posible vaciar tablas y mover tablas entre esquemas. Usted puede *Run Vacuum Analyze* para una tabla seleccionada. *Vacuum* reclama espacio y lo pone a disposición para su reutilización, y *analyze* estadísticas actualizadas que son usadas para determinar el modo mas eficiente para ejecutar una consulta. *Change Logging…* le permite agregar soporte de registro de cambios a una tabla. Finalmente puede *Import Layer/File…* y *Export to File…*.

Nota

Usando DB Manager es posible agregar comentarios para tablas y columnas de una base de datos PostgreSQL.

La ventana *Providers* lista todas las bases de datos existentes soportadas por QGIS. Con un doble-click, puede conectar

a una base de datos. Con el click derecho en el botón del ratón, puede renombrar y borrar esquemas y tablas existentes. Las tablas pueden además ser añadidas al canvas de QGIS con el menú contextual.

Si está conectado a una base de datos, la ventana **main** del administrador de BBDD le ofrece cuatro pestañas. La pestaña *Info* proporciona información sobre la tabla y su geometría, así como sobre campos existentes, restricciones e índices. Le permite crear un índice espacial en la tabla seleccionada. La pestaña *Table* muestra la tabla, y la pestaña *Preview* representa las geometrías como previsualización. Cuando abre una *SQL Window*, será ubicada en una nueva pestaña.

Trabajando con la ventana SQL

Puede usar el Administrador de BBDD para ejecutar consultas SQL en su base de datos espacial. Las consultas se pueden guardar y cargar, y allí el *SQL Query Builder* le ayudará a formular sus consultas. Incluso puede ver la salida espacial marcando *Load as new layer* y especificando *Column(s) with unique values* (IDs), *Geometry column* y *Layer name (prefix)*. Es posible resaltar una parte del SQL para ejecutar solo esa parte al presionar `Ctrl+R` o clickando el botón *Execute*.

QGIS also adds support for the `REGEXP` function in some providers. This allows users to use regular expressions in SQL filters or expressions, for example:

```
SELECT * FROM places WHERE name REGEXP '^A';
```

This returns all features where the `name` field starts with the letter A.

Después de ejecutar la consulta, puede seleccionar celdas específicas en el conjunto de resultados. Utilice el método abreviado `Ctrl+C` para copiar las celdas seleccionadas en el portapapeles. Los datos copiados estarán disponibles en forma de tabla formateada. Esto le permite pegar los datos en otras aplicaciones, como hojas de cálculo, donde aparecerán como una tabla.

El botón *Historial de consulta* almacena las últimas 20 consultas de cada base de datos y proveedor.

Al hacer doble clic en una entrada, se agregará la cadena a la ventana SQL.

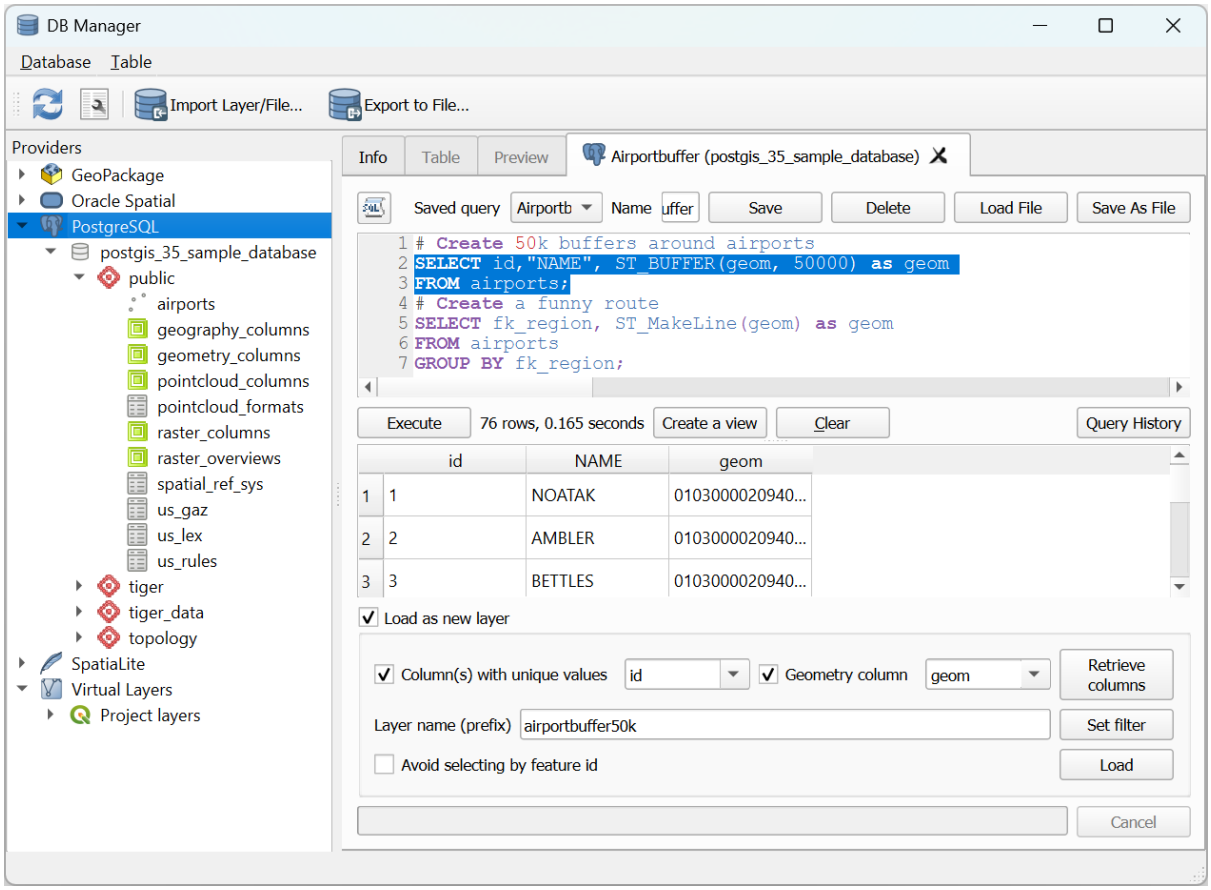


Figura 25.5: Ejecutando consultas SQL en la ventana SQL del Administrador de BBDD

Nota

La ventana SQL también se puede utilizar para crear capas virtuales. En ese caso, en lugar de seleccionar una base de datos, seleccione **QGIS Layers** debajo de **Virtual Layers** antes de abrir la ventana SQL. Ver *Creando capas virtuales* para instrucciones de la sintaxis SQL a usar.

Import Vector Layer

You can import layer or file into your database. Here are the parameters you can set for the import process:

- *Input*: Select the layer or file to import. Using the dropdown menu, select from the list of loaded layers in QGIS or click on the … button to select a file from disk. Check the ☑ *Import only selected features* to import only the selected features of the layer.
- *Output table*: Choose the *Schema* and provide a name for the new table.
- *Options*: Here are some options for the import process:
  - ☑ *Primary key*: Provide naming for the primary key field. By default, it is named `id`.
  - ☑ *Geometry column*: Provide naming for the geometry column. By default, it is named `geom`.
  - ☑ *Source SRID*: Define the SRID for the geometry column. By default, it uses the layer's CRS.
  - ☑ *Target SRID*: Define the target SRID to reproject the geometries during the import process. By default, it uses the layer's CRS.

- *Encoding*: Define the encoding of the source data. By default, it uses `Automatic`. It is QGIS's automatic detection mode that attempts to guess the file's character encoding based on available metadata or system locale.
- *Replace destination table (if exists)*: If a table with the same name already exists in the selected schema, it will be replaced.
- *Do not promote to multi-part*: Geometries will be imported as single-part geometries.
- *Convert field names to lower case*: All field names will be converted to lower case.
- *Create spatial index*: A spatial index will be created on the geometry column after import.
- *Comment*: Add comments to table. Only available for PostgreSQL databases.

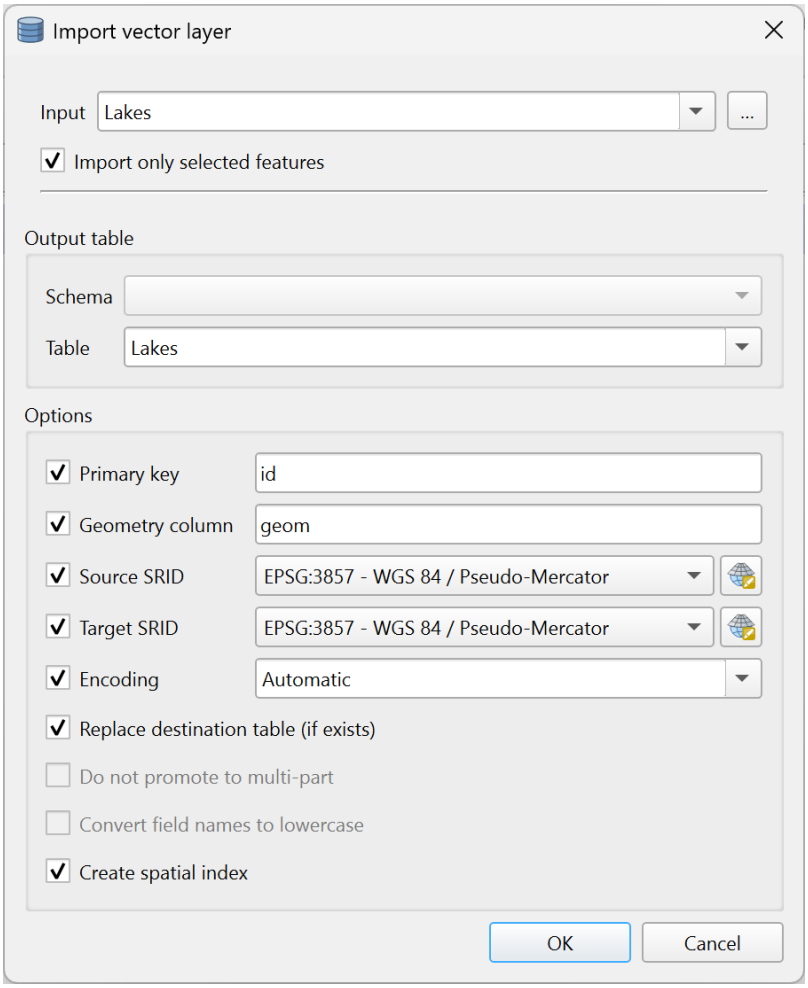


Figura 25.6: Importing a vector layer into a spatial database using DB Manager

Export to Vector File

To export a table from your database to a vector file, select the desired *Format* and *Save as* location. Under *Options*, you can set the following parameters:

- *Source SRID*: Define the source SRID of the geometry column. By default, it uses the layer's CRS.
- *Target SRID*: Define the target SRID to reproject the geometries during the export process. By default, it uses the layer's CRS.
- *Encoding*: Define the encoding of the output data.
- *Replace destination table (if exists)*: If a table with the same name already exists at the selected location, it will be replaced.

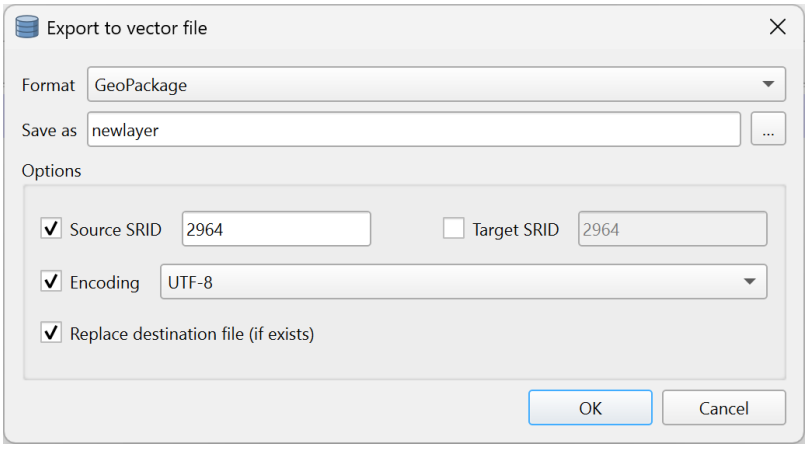


Figura 25.7: Exporting a table to a vector file using DB Manager

25.2.2 Complemento Verificador de Geometría

Verificador de Geometría es un poderoso complemento nuclear para verificar y corregir la validez de geometría de una capa. Está disponible desde el *Vector* menu (*Check Geometries…*).

Configurando las pruebas

El diálogo *Check Geometries* muestra diferentes configuraciones agrupadas en la primera pestaña (*Setup*):

- *Input vector layers*: para seleccionar las capas a chequear. Un checkbox *Only selected features* se puede usar para restringir la verificación a las geometrías de las entidades seleccionadas.
- :guilabel:`Tipos de geometría permitidos`dá la oportunidad de restringir el tipo de geometría de la(s) capa(s) de entrada:
  - De punto
  - Multipunto
  - Lineal
  - Multilínea
  - Poligonal

  - Multipolígono
- *Validez de Geometría*. Dependiendo de los tipos de geometría que elija entre:
  - *Autointersecciónes*
  - *Nodos Duplicados*
  - *Autocontactos*
  - *Polígono con menos de 3 nodos*.
- *Propiedades de Geometría*. Dependiendo del tipo de geometría, hay disponibles diferentes opciones:
  - *Polígonos y multipolígonos no deben contener huecos*
  - *Objetos Multiparte debe constar de mas de una parte*
  - *Las líneas no deben tener oscilaciones*
- *Condiciones de Geometría*. Le permite añadir alguna condición con la cuál validar las geometrías:
  - *Longitud Mínima de Segmento (unidades de mapa)* 1,00
  - *Ángulo Mínimo entre segmentos (grados)* 1,00
  - *Mínima Área Poligonal (unidades cuadradas de mapa)* 1,00
  - *No polígonos de plata* con una *Delgadez Máxima* 1,00 y un *área Max. (unidades cuadradas de mapa)* 1,00
- *Comprobaciones Topológicas*. Dependiendo del tipo de geometría, hay muchas opciones disponibles:
  - *Comprobar Duplicados*
  - *Comprobar entidades dentro de otras entidades*
  - *Comprueba si hay superposiciones menores que* 1,00
  - *Comprueba si hay espacios más pequeños que* 1,00
  - *Puntos deben ser cubiertos por líneas*
  - *Los puntos deben estar correctamente dentro de un polígono*
  - *Las líneas no deben cruzarse con ninguna otra línea.*
  - *Las líneas no deben cruzarse con las entidades de la capa.* ...
  - *Los polígonos deben seguir los límites de la capa* ...
- *Tolerancia*. Puede definir la tolerancia de la comprobación en unidades de mapa.
- *Capa Vectorial Saliente* dá la opción de:
  - *Modificar la capa de entrada*
  - *Crear nuevas capas*

Cuando esté satisfecha con la configuración, puede hacer clic en el botón *Run*

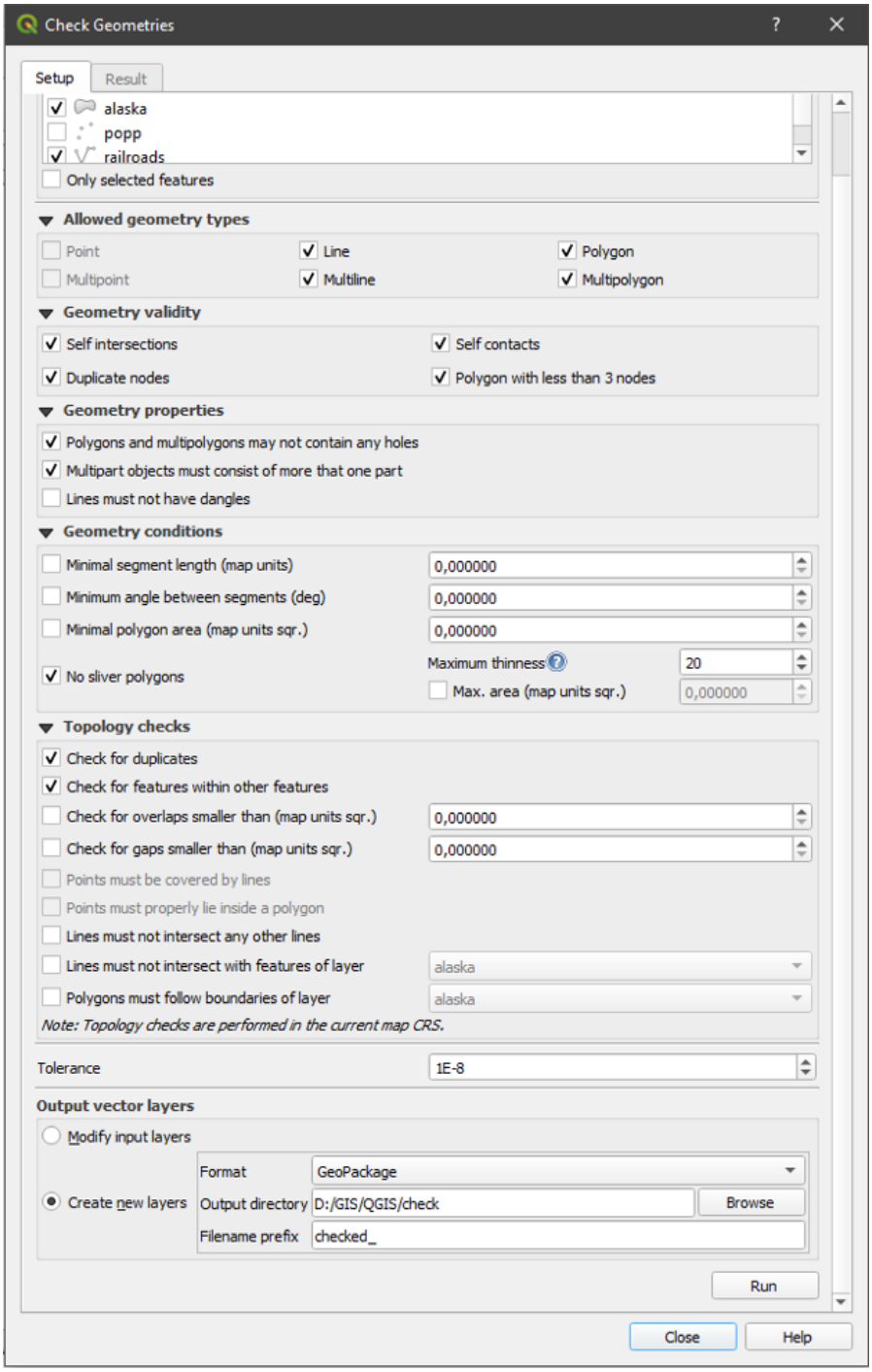


Figura 25.8: El complemento Verificador de Geometría

El *Geometry Checker Plugin* puede encontrar los siguientes errores:

- Autointersecciones: un polígono con una autointersección
- Nodos duplicados: dos nodos duplicados en un segmento
- Huecos: hueco en un polígono
- Logitud de Segmento: una longitud de segmento menor que un umbral
- Ángulo mínimo:dos segmentos con un ángulo menor que un umbral
- Área mínima: area poligonal inferior a un umbral

- Polígono de plata: este error viene de polígonos mmuy pequeños (con área pequeña) y un perímetro muy largo
- Entidades Duplicadas
- Entidad dentro de entidad
- Superposiciones: superposición poligonal
- Brechas: brechas entre polígonos

La siguiente figura muestra las diferentes comprobaciones realizadas por el complemento.

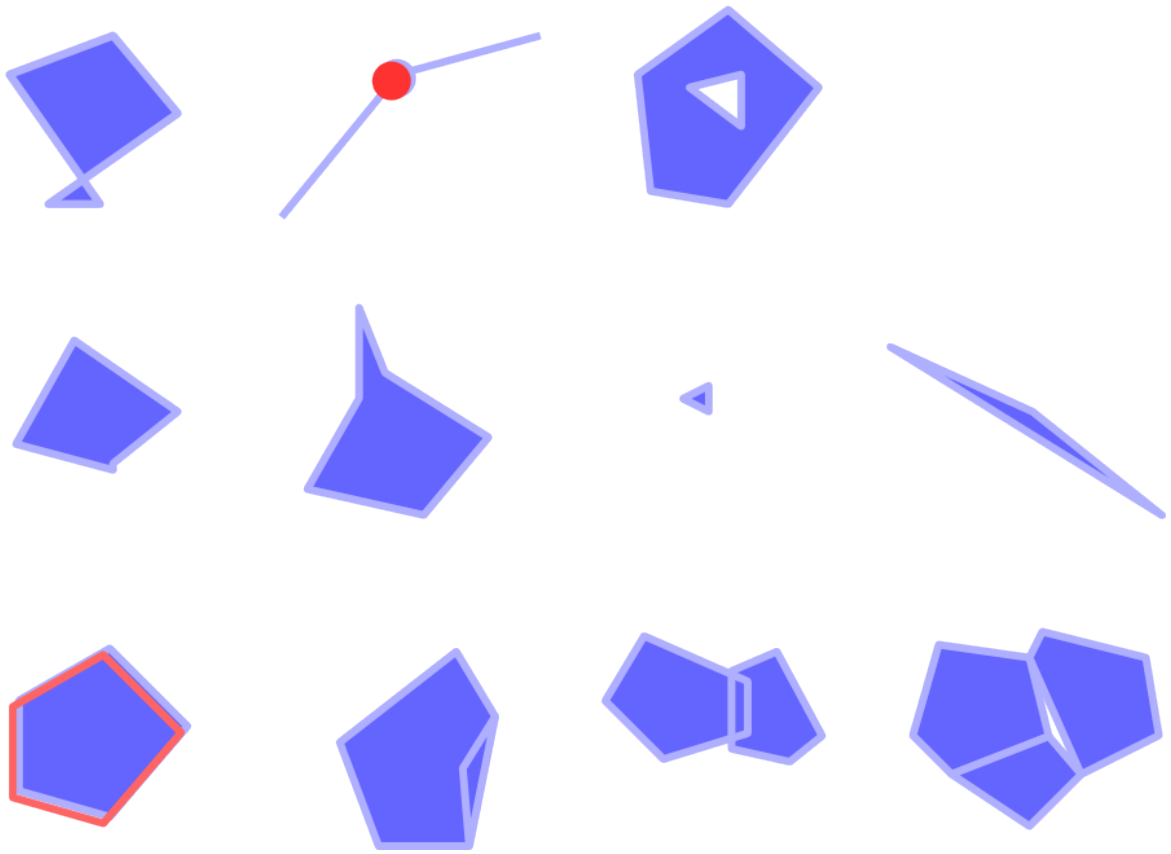


Figura 25.9: Algunas comprobaciones soportadas por el complemento

Analizando los resultados

Los resultados aparecen en la segunda pestaña (*Resultado*) y como una capa de descripción general de los errores en el lienzo (su nombre tiene el prefijo predeterminado `checked_`). Una tabla enumera *resultado de comprobación geométrica* con un error por fila y columnas que contienen: el nombre de la capa, una ID, el tipo de error, luego las coordenadas del error, un valor (dependiendo del tipo de error) y finalmente la columna de resolución que indica la resolución del error. Al final de esta tabla, puede *Exportar* el error a diferentes formatos de archivo. También tiene un contador con el número de errores totales y los corregidos.

Puede seleccionar una fila para ver la ubicación del error. Puede cambiar este comportamiento seleccionando otra acción entre *Error* (predeterminado), *Feature*, *Don't move*, y *Highlight selected features*.

Debajo de la acción de zoom al hacer click en la fila de la tabla, puede:

- *Mostrar entidades seleccionadas en la tabla de atributos*
- *Corregir errores seleccionados usando la resolución predeterminada*

- ✔ *Corregir errores seleccionados, solicitud del método de resolución* Verá una ventana para elegir el método de resolución entre los cuales:
  - Unir con el polígono vecino con el borde compartido más largo
  - Unir con polígono vecino con mayor área
  - Unir con polígono vecino con idéntico valor de atributo, si hay, o dejar como está
  - Borrar entidad
  - Acción no
- 🔧 *Ajustes de resolución de Errores* le permite cambiar el método de resolución predeterminado dependiendo del tipo de error

> **Truco**
>
> **Corregir errores múltiples**
>
> Puede corregir múltiples errores seleccionando más de una fila en la tabla con la acción *CTRL + click*.

Finálmente, puede elegir que *Atributo usar al unir entidades por valor de atributo*.

25.2.3 Cliente de Catálogo de metasearch

Introducción

MetaSearch es un complemento de QGIS para interactuar con servicios de catálogos de metadatos, compatible con los estándares OGC API - Records y OGC Catalog Service for the Web (CSW).

MetaSearch proporciona un enfoque fácil e intuitivo y una interfaz fácil de usar para buscar catálogos de metadatos dentro de QGIS.

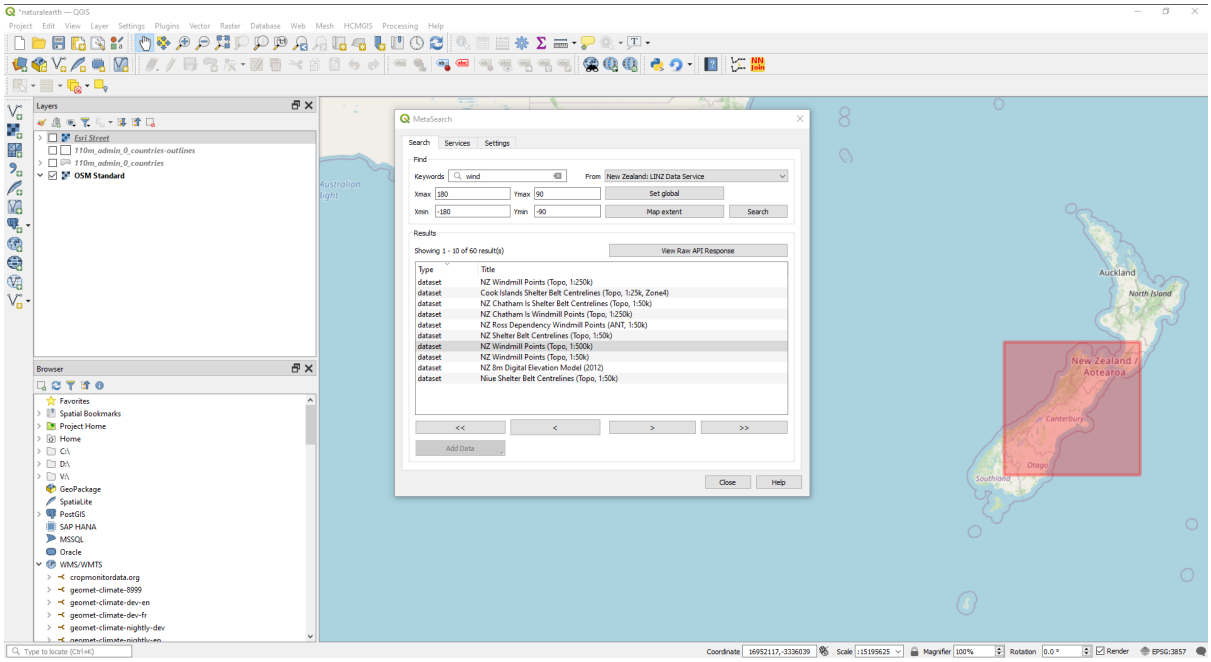


Figura 25.10: Búsqueda y resultados de servicios en MetaSearch

Trabajar con catálogos de metadatos en QGIS

MetaSearch se incluye de forma predeterminada en QGIS, con todas sus dependencias, y se puede habilitar desde el Administrador de complementos de QGIS.

API OGC - Registros

La OGC API - Registros es un estándar del OGC (Open Geospatial Consortium) para el descubrimiento de recursos geoespaciales en la Web.

CSW (Servicio de catálogo para la web)

CSW (Servicio de catálogo para la Web) _ es una especificación de OGC (Consorcio Geoespacial Abierto) que define interfaces comunes para descubrir, explorar y consultar metadatos sobre datos, servicios y otros recursos potenciales.

inicio

Para iniciar MetaSearh, click en el icono  o selecciona *Web* ► *MetaSearch* ► *MetaSearch* a través del menú principal de QGIS. Aparecerá el diálogo de MetaSearch. La GUI principal consta de tres pestañas: *Servicios*, *Buscar* y *Configuración*.

Administrando los catálogos de servicios

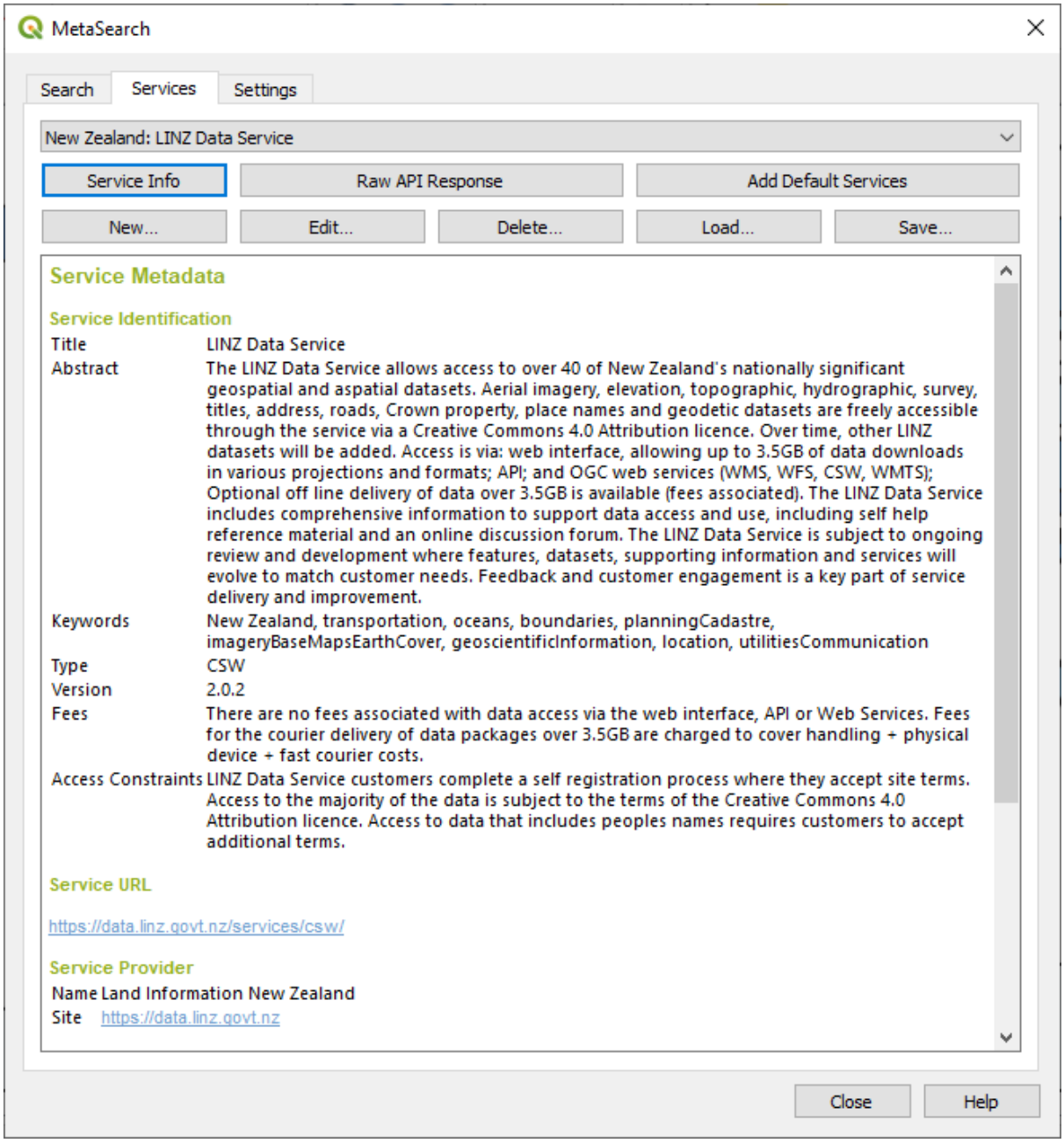


Figura 25.11: Administrando los catálogos de servicios

La pestaña *Servicios* permite al usuario administrar todos los servicios de catálogo disponibles. MetaSearch proporciona una lista predeterminada de servicios de catálogo, que se puede agregar presionando el botón *Agregar servicios predeterminados*.

Para encontrar todas las entradas del Catálogo de Servicios enumeradas, haga clic en el cuadro de selección desplegable.

Para añadir una entrada del catálogo de servicios:

1. Click en el botón *Nuevo*
2. Introduzca un *Nombre* para el servicio, así como la *URL* (punto final). Tenga en cuenta que para los catálogos OGC CSW 2.0.2, sólo se requiere la URL base (no una URL GetCapabilities completa). Para la API OGC -

Catálogos de registros, la URL debe ser la ruta al punto final de la colección

3. Si el Catálogo requiere autenticación, introduzca las credenciales *Nombre de usuario* y *Contraseña* adecuadas.
4. Click en *Aceptar* para agregar el servicio a la lista de entradas.

Para editar una entrada del catálogo de servicios existente:

1. Seleccione la entrada que le gustaría editar
2. Click en el botón *Editar*
3. Y modifica los valores *Nombre* o *URL*
4. Haz clic en *Aceptar*.

Para eliminar una entrada del catálogo de servicios, seleccione la entrada que desea eliminar y haga clic en el botón *Eliminar*. Se le pedirá que confirme la eliminación de la entrada.

MetaSearch permite cargar y guardar conexiones a un archivo XML. Esto es útil cuando necesita compartir configuraciones entre aplicaciones. A continuación se muestra un ejemplo del formato de archivo XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<qgsCSWConnections version="1.0">
    <csw type="OGC CSW 2.0.2" name="Data.gov CSW" url="https://catalog.data.gov/
↪csw-all"/>
    <csw type="OGC CSW 2.0.2" name="Geonorge - National CSW service for Norway"␣
↪url="https://www.geonorge.no/geonetwork/srv/eng/csw"/>
    <csw type="OGC CSW 2.0.2" name="Geoportale Nazionale - Servizio di ricerca␣
↪Italiano" url="http://www.pcn.minambiente.it/geoportal/csw"/>
    <csw type="OGC CSW 2.0.2" name="LINZ Data Service" url="http://data.linz.govt.
↪nz/feeds/csw"/>
    <csw type="OGC CSW 2.0.2" name="Nationaal Georegister (Nederland)" url="http://
↪www.nationaalgeoregister.nl/geonetwork/srv/eng/csw"/>
    <csw type="OGC CSW 2.0.2" name="RNDT - Repertorio Nazionale dei Dati␣
↪Territoriali - Servizio di ricerca" url="http://www.rndt.gov.it/RNDT/CSW"/>
    <csw type="OGC CSW 2.0.2" name="UK Location Catalogue Publishing Service" url=
↪"http://csw.data.gov.uk/geonetwork/srv/en/csw"/>
    <csw type="OGC CSW 2.0.2" name="UNEP/GRID-Geneva Metadata Catalog" url="http://
↪metadata.grid.unep.ch:8080/geonetwork/srv/eng/csw"/>
</qgsCSWConnections>
```

Para cargar una lista de entradas:

1. Click en el botón *Cargar*. Aparecerá una nueva ventana.
2. Click en el botón *Navegar* y navega hasta el archivo XML de las entradas que desea cargar.
3. Click *Abrir*. Se mostrará la lista de entradas.
4. Seleccione las entradas que desea agregar de la lista y haga click en *Cargar*.

Haga clic en el botón *Información de servicio* para mostrar información sobre el Servicio de Catálogo seleccionado, como la identificación del servicio, el proveedor del servicio y la información de contacto. Si desea ver la respuesta API sin procesar, haga clic en el botón *Respuesta API sin procesar*. Se abrirá una ventana independiente que mostrará la información del servidor en formato JSON o XML sin procesar.

Buscando Catálogos de Servicios

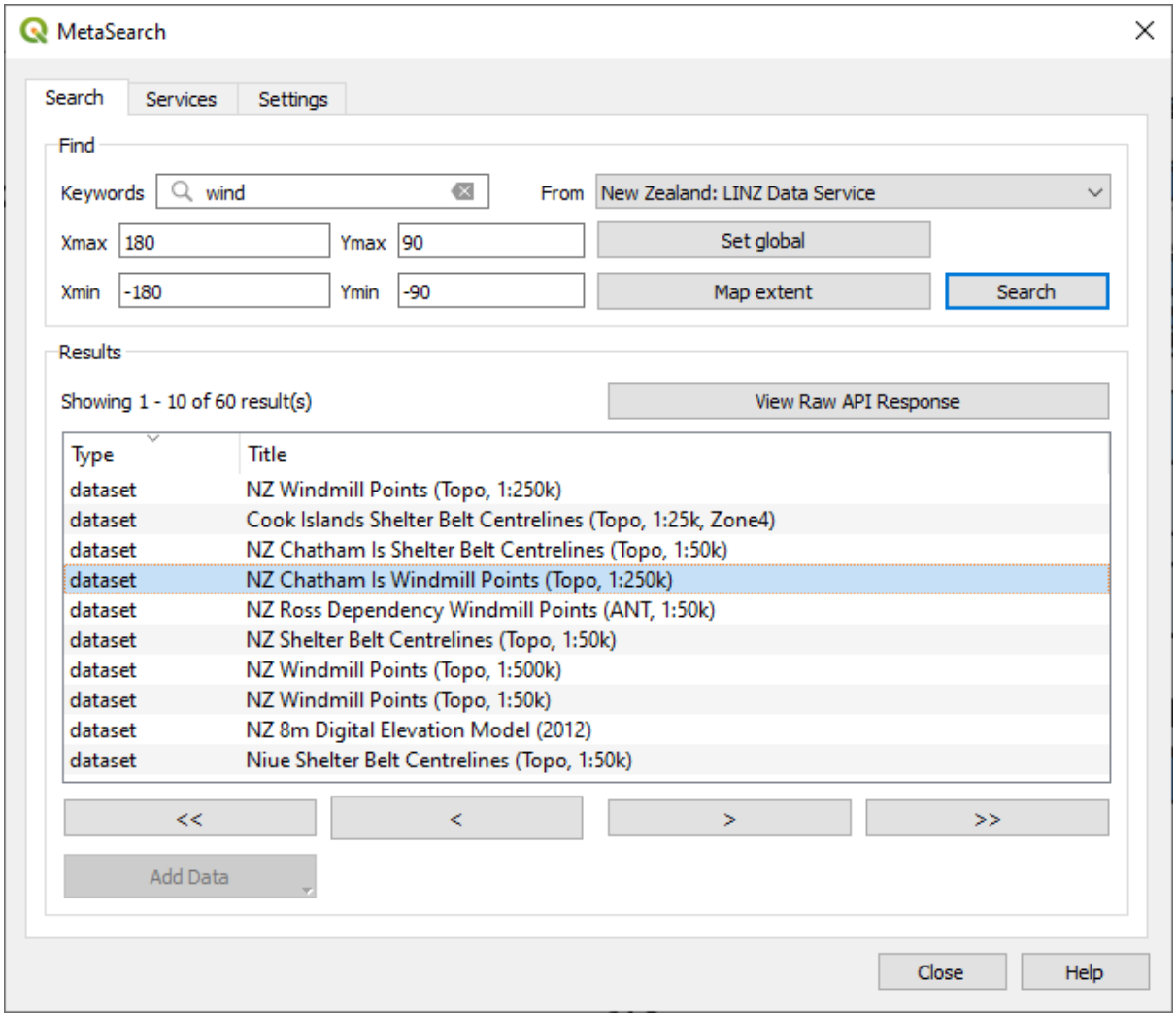


Figura 25.12: Buscando catálogo de servicios

La pestaña *Buscar* permite al usuario consultar los catálogos de servicios para obtener datos y servicios, establecer varios parámetros de búsqueda y ver los resultados.

Los siguientes parámetros de búsqueda estan disponibles:

- *Palabras clave*: palabras clave de búsqueda de texto libre;
- *De*: el servicio de catálogo sobre el que realizar la consulta;
- **Recuadro límite**: el área espacial de interés a filtrar, definido por *Xmax*, *Xmin*, *Ymax*, and *Ymin*. Click en *Establecer Global* para hacer una búsqueda global, click en *Extensión de Mapa* para hacer una búsqueda en el área visible, o introducir valores manualmente.

Al hacer clic en el botón *Buscar*, se buscará en el catálogo de metadatos seleccionado. Los resultados de la búsqueda se muestran en una lista y se pueden ordenar haciendo clic en el encabezado de la columna. Puede navegar por los resultados de la búsqueda con los botones de dirección debajo de los resultados de la búsqueda.

Seleccione un resultado y:

- Haga clic en el botón *Ver respuesta API sin procesar* para abrir una ventana con la respuesta del servicio en formato JSON o XML sin procesar.

- Si el registro de metadatos tiene un recuadro límite asociado, se mostrará una huella del cuadro delimitador en el mapa.
- Haga doble clic en el registro para mostrar los metadatos del registro con los enlaces de acceso asociados. Al hacer clic en un enlace, se abre el enlace en el navegador web del usuario.
- Si el registro es un servicio web compatible (WMS/WMTS, WFS, WCS, ArcGIS REST Service, etc.) o un archivo SIG, se activará el botón *Añadir datos*. Al hacer clic en este botón, MetaSearch verificará si se trata de un OWS válido. A continuación, el servicio se añadirá a la lista de conexiones QGIS correspondiente y aparecerá el cuadro de diálogo de conexión correspondiente. Si selecciona *Añadir archivo SIG*, el archivo SIG vinculado se añadirá al proyecto actual.

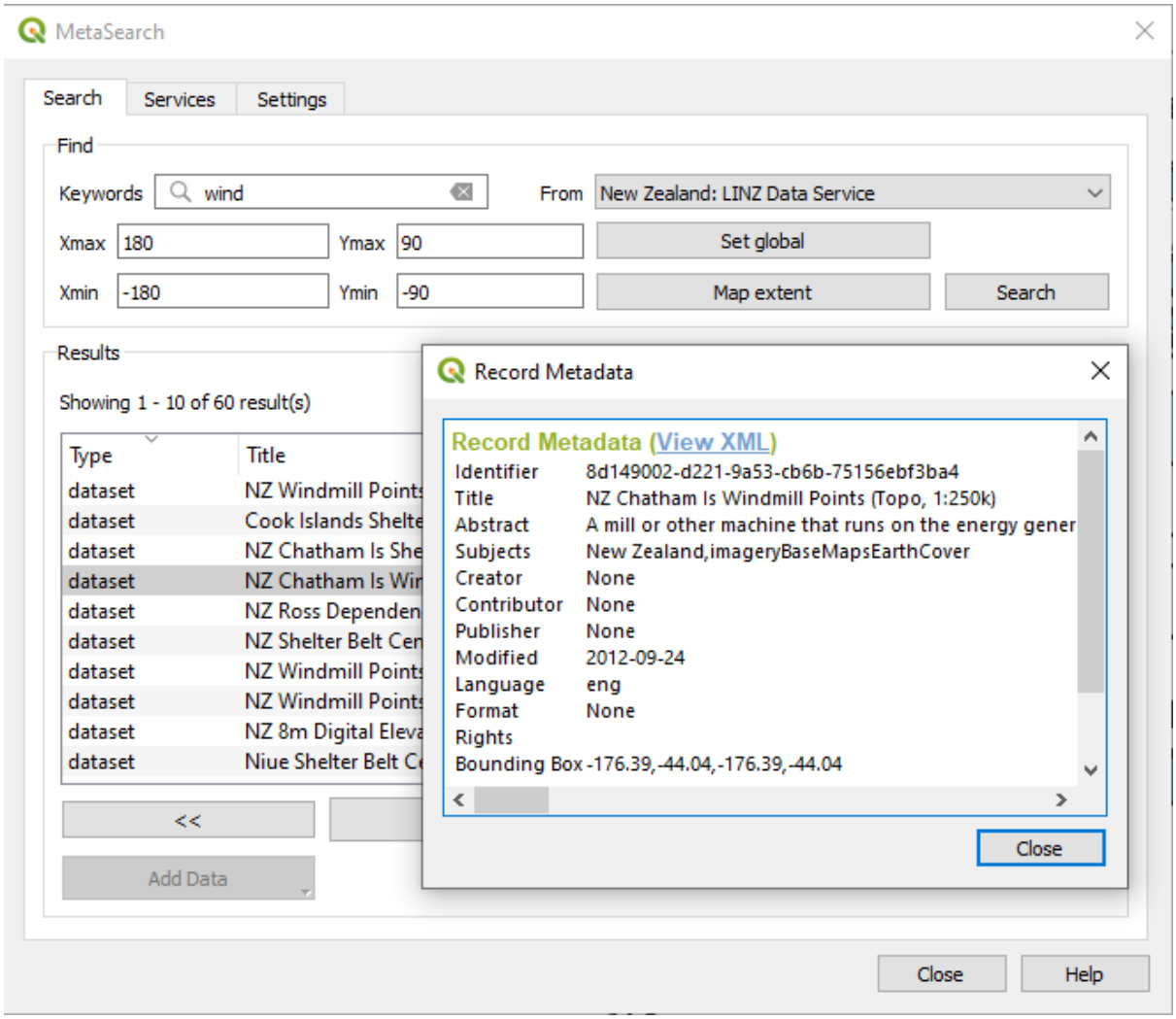


Figura 25.13: Visualización del registro de metadatos

Configuración

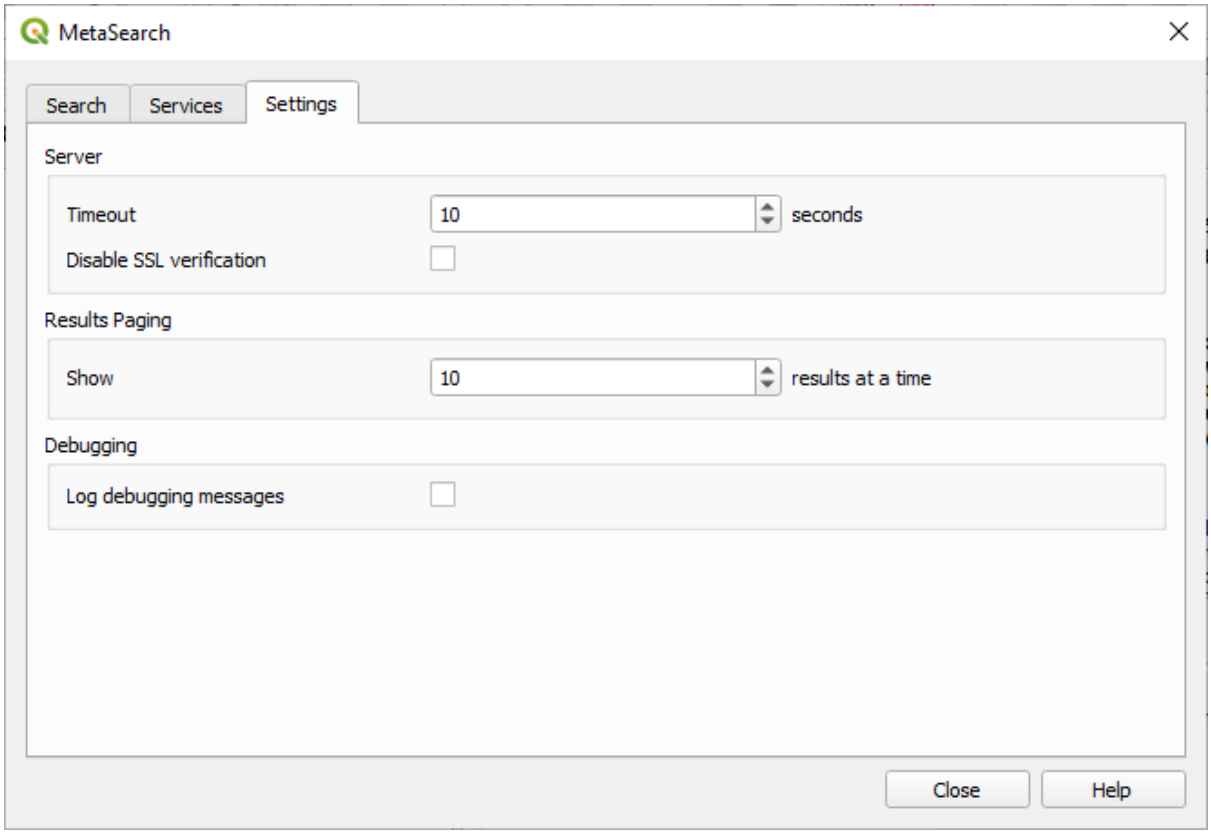


Figura 25.14: Configuración de MetaSearch

Puede ajustar MetaSearch con la siguiente *Configuración*:

- *Timeout del Servidor*: al buscar catálogos de metadatos, la cantidad de segundos para bloquear el intento de conexión. El valor predeterminado es 10.
- *Desactivar verificación SSL*: opción para desactivar la verificación SSL.
- *Paginación de resultados*: cuando se buscan catálogos de metadatos, el número de resultados que se muestran por página. El valor predeterminado es 10.
- *Mensajes de depuración de registro*: opción para activar el registro de mensajes de depuración en el *Panel de mensajes de registro*.

Errores del servidor de catálogos

En algunos casos, el Catálogo funcionará en un navegador web, pero no en MetaSearch. Esto puede deberse a la configuración del servidor de catálogos. Los proveedores de servidores de catálogos deben asegurarse de que las URL son consistentes y están actualizadas en su configuración (esto es común en escenarios de redirección HTTP -> HTTPS). Por favor, consulte el artículo pycsw FAQ item para una explicación más detallada del problema y su solución. Aunque el artículo de FAQ es específico de pycsw también puede aplicarse en general a otros servidores de Catálogo.

25.2.4 Complemento de Edición Fuera de Línea

For data collection, it is a common situation to work with a laptop or a cell phone offline in the field. Upon returning to the network, the changes need to be synchronized with the master datasource (e.g., a PostgreSQL database). If several persons are working simultaneously on the same datasets, it is difficult to merge the edits by hand, even if people don't change the same features.

La  Offline Editing automatiza la sincronización copiando el contenido de la fuente de datos en una base de datos SpatiaLite o GeoPackage y almacenando las ediciones fuera de línea en tablas dedicadas. Después de volver a conectarse a la red, es posible aplicar las ediciones fuera de línea al conjunto de datos principal.

Para usar el complemento:

1. Open a project with some vector layers (e.g., from an Esri Shapefile, PostgreSQL or WFS-T datasource).
2. Suponiendo que ya ha habilitado el complemento (ver *Complementos base y externos*) vaya a *Base de datos -> Edición sin conexión -> \`\|offlineEditingCopy\| :guilabel:`Convertir a proyecto fuera de línea*. Se abre el cuadro de diálogo del epónimo.
3. Seleccione *Tipo de almacenamiento*. Puede ser del tipo de base de datos *GeoPackage* o *SpatiaLite*.
4. Utilice el botón *Examinar* para indicar la ubicación de la base de datos en la que almacenar :guilabel:'Datos fuera de línea`. Puede ser un archivo existente o uno para crear.
5. En la sección *Seleccionar capas remotas*, marque las capas que desea guardar. El contenido de las capas se guarda en tablas de la base de datos.

   > **Nota**
   >
   > Dado que los formatos de la base de datos de destino no admiten listas nativas, el complemento de edición sin conexión transforma los campos de lista {cadena, número} en campos de cadena donde los valores están separados por comas. Esto permite leer y editar el contenido de esos campos cuando está fuera de línea.
   >
   > Si desea manejar tanto el campo de la capa original como el de la capa fuera de línea, puede confiar en el *try()* and *array* expression functions, e.g.:
   >
   > ```
   > try(array_contains("field",1),array_contains(string_to_array("field"),1))
   > ```

6. Puede marcar  :guilabel:'Solo sincroniza las entidades seleccionadas si hay una selección presente` permitiendo guardar y trabajar solo en un subconjunto. Puede ser inestimable en caso de capas grandes.

   ¡Esto es todo!
7. Guarde su proyecto y llévelo al campo.
8. Edite las capas sin conexión.
9. Después de conectarse nuevamente, cargue los cambios usando *Database -> Edición Fuera de Línea ->*  *Sincronizar*.

> **Nota**
>
> Las capas que se utilizan fuera de línea se marcan con el  icono en el panel *Capas*.

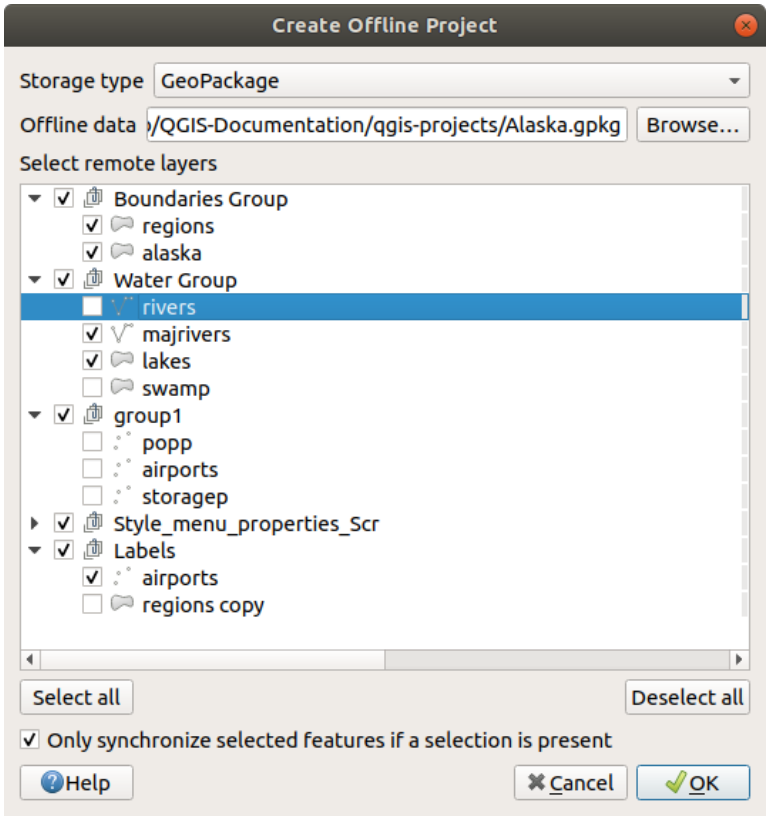


Figura 25.15: Crea un proyecto sin conexión

25.2.5 Complemento Comprobador de Topología

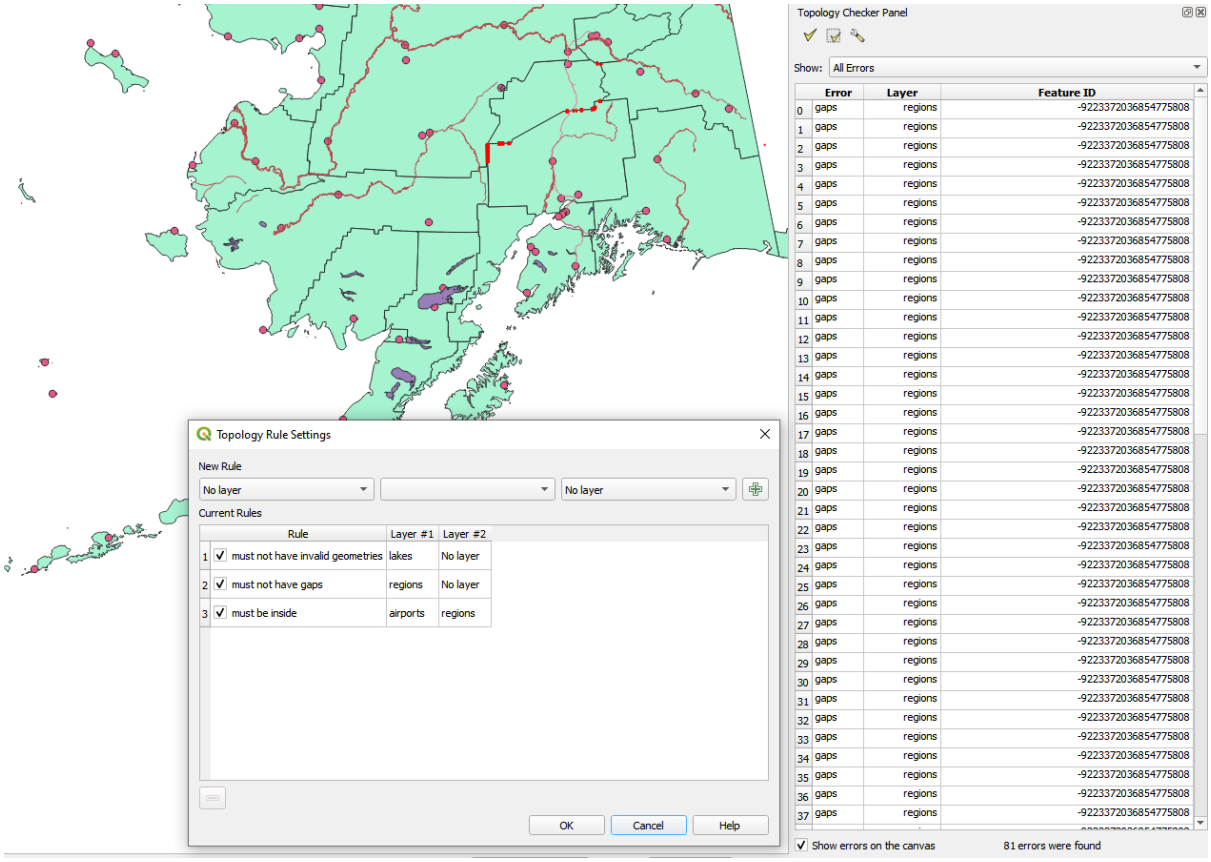


Figura 25.16: El complemento Comprobador de Topología

La topología describe las relaciones entre puntos, líneas y polígonos que representan las características de una región geográfica. Con el complementoComprobador de Topología, puede revisar sus archivos vectoriales y verificar la topología con varias reglas de topología. Estas reglas verifican con las relaciones espaciales si sus entidades “Igual”, “Contener”, “Cobertura”, “CoveredBy”, “Cross”, son “Disjoint”, “Intersect”, “Overlap”, “Touch” o son “Within “ El uno al otro. Depende de sus preguntas individuales qué reglas de topología aplica a sus datos vectoriales (por ejemplo, normalmente no aceptará sobrepasos en las capas de línea, pero si representan calles sin salida, no las eliminará de su capa vectorial).

QGIS tiene un función constructora de edición topológica incorporada, que es ideal para crear nuevas funciones sin errores. Pero los errores de datos existentes y los errores inducidos por el usuario son difíciles de encontrar. Este complemento lo ayuda a encontrar tales errores a través de una lista de reglas.

Para activar el complemento *Comprobador de topología*:

1. Ir al menú *Complementos*
2. Abra *Gestionar e instalar complementos* y elija *Instalado*
3. Activar **|checkbox||topologyChecker|** *Comprobador de topología*
4. Cierra el diálogo *Administrador de complementos*. Se añade una entrada *Comprobador de topología* al menú *Vector*.

Después de activar *Comprobador de topología* ábralo y elija *Configure* para crear sus reglas de topología.

En ** capas de puntos **, están disponibles las siguientes reglas:

- *Debe estar cubierto por*: Aquí puede elegir una capa vectorial de su proyecto. Los puntos que no están cubiertos por la capa vectorial dada aparecen en el campo *Error*.
- *Debe estar cubierto por puntos finales de*: Aquí puede elegir una capa de línea de su proyecto.
- *Debe estar dentro*: Aquí puede elegir una capa poligonal de su proyecto. Los puntos deben estar dentro de un polígono. De lo contrario, QGIS escribe un "Error" para el punto.
- *No debe haber duplicados*: Siempre que un punto se represente dos veces o más, aparecerá en el campo *Error*.
- *No debe tener geometrías inválidas*: Comprueba si las geometrías son válidas.
- *No debe tener geometrías multiparte*: Todos los puntos multiparte se escriben en el campo "Error".

En ** capas de línea **, están disponibles las siguientes reglas:

- *Los puntos finales deben estar cubiertos por*: Aquí puede seleccionar una capa de puntos de su proyecto.
- *No debe tener salientes*: Esto mostrará los rebasamientos en la capa de líneas.
- *No debe tener duplicados*: Siempre que un objeto espacial línea se represente dos veces o más, aparecerá en el campo *Error*.
- *No debe tener geometrías inválidas*: Comprueba si las geometrías son válidas.
- *No debe tener geometrías multiparte*: A veces, una geometría es en realidad una colección de geometrías simples (de una sola parte). Este tipo de geometría se denomina geometría multiparte. Si sólo contiene un tipo de geometría simple, se denomina multipunto, multilínea o multipolígono. Todas las líneas multiparte se escriben en el campo «Error».
- *No debe tener pseudos*: El punto final de una geometría lineal debe estar conectado a los puntos finales de otras dos geometrías. Si el punto final sólo está conectado al punto final de otra geometría, el punto final se denomina pseudonodo.

En ** capas de polígono **, están disponibles las siguientes reglas:

- *Debe contener*: La capa de polígonos debe contener al menos una geometría de puntos de la segunda capa.
- *No debe tener duplicados*: Los polígonos de una misma capa no deben tener geometrías idénticas. Siempre que un objeto espacial polígono se represente dos veces o más aparecerá en el campo *Error*.
- *No debe tener vacíos*: Los polígonos adyacentes no deben formar huecos entre ellos. Los límites administrativos podrían mencionarse como ejemplo (los polígonos de los estados de EE.UU. no tienen huecos entre ellos…).
- *No debe tener geometrías no válidas*: Comprueba si las geometrías son válidas. Algunas de las reglas que definen una geometría válida son:
  - Los anillos poligonales deben cerrarse.
  - Los anillos que definen los agujeros deben estar dentro de los anillos que definen los límites exteriores.
  - Los anillos no pueden cruzarse a sí mismos (no pueden tocarse ni cruzarse entre sí).
  - Los anillos no pueden tocar otros anillos, excepto en un punto.
- *No debe tener geometrías multiparte*: A veces, una geometría es en realidad una colección de geometrías simples (de una sola parte). Este tipo de geometría se denomina geometría multiparte. Si sólo contiene un tipo de geometría simple, se denomina multipunto, multilínea o multipolígono. Por ejemplo, un país formado por varias islas puede representarse como un multipolígono.
- *No deben solaparse*: Los polígonos adyacentes no deben compartir un área común.
- *No debe solaparse con*: Los polígonos adyacentes de una capa no deben compartir área común con polígonos de otra capa.

Cuando cree una *Nueva regla*, haga clic en  Añadir regla para incluirla en *Reglas actuales*. Puede activar o desactivar reglas individuales haciendo clic en la casilla de verificación. Haciendo clic con el botón derecho sobre una regla se obtienen las siguientes opciones:

- *Seleccionar todas* las reglas

- *Activar* o *Desactivar* las reglas seleccionadas
- *Conmutar la activación* de las reglas seleccionadas
- *Borrar* las reglas seleccionadas. Esto también se puede conseguir con el botón Borrar reglas seleccionadas.

Pulse *Aceptar* y elija en el panel *Comprobador de topología*:

- Validar todo: aplica las reglas activas a todos los objetos espaciales de la(s) capa(s) implicada(s).
- o Validar Extensión: aplica las reglas activas a los objetos espaciales de la(s) capa(s) implicada(s), dentro del lienzo de mapa actual. El botón se mantiene pulsado y los resultados se actualizan a medida que cambia la extensión del lienzo del mapa.

Los errores se mostrarán en la tabla de resultados con el tipo de error, la capa y el ID del objeto espacial. Utilice el menú *Filtrar errores por regla* para filtrar los errores a un tipo de error específico.

Marque *Mostrar errores en el lienzo* para mostrar la ubicación del error en el lienzo. Al hacer clic en una fila de la tabla se ampliará el lienzo del mapa hasta el objeto espacial en cuestión, donde podrá utilizar *Herramientas de digitalización de QGIS* para solucionar el error.

A continuación se muestra la lista de complementos principales que se proporcionan con QGIS. No están necesariamente habilitados de forma predeterminada.

Iconc	Complemento	Descripción	Manual de referencia
	Administrador de BBDD	Administrar tus bases de datos dentro de QGIS	*Complemento de Administración de BBDD*
	Comprobador de Geometría	Comprobar y reparar errores en geometrías vectoriales	*Complemento Verificador de Geometría*
	GRASS 7	Funcionalidad GRASS	*Integracion GRASS SIG*
	Proveedor SIG GRASS	Funcionalidad de procesamiento SIG GRASS	*Integracion GRASS SIG*
	Cliente de Catálogo de metasearch	Interacturar con metadatos del catálogo de servicios (CSW)	*Cliente de Catálogo de metasearch*
	Edición fuera de línea	Edición fuera de línea y sincronización con la base de datos	*Complemento de Edición Fuera de Línea*
	Procesado	Espacio de trabajo para el procesamiento de datos espaciales	*Entorno de trabajo de procesamiento de QGIS*
	Comprobador de Topología	Encontrar errores topológicos en capas vectoriales	*Complemento Comprobador de Topología*

Nota

Para usar los complementos núcleo GRASS 7, GRASS GIS provider, tienen que ser configurados. La información se puede encontrar *aquí*.

25.3 Consola Python de QGIS

Como verá más adelante en este capítulo, QGIS ha sido diseñado con una arquitectura de complementos. Los complementos se pueden escribir en Python, un lenguaje muy famoso en el mundo geoespacial.

QGIS trae una API de Python (ver PyQGIS Developer Cookbook para algunos ejemplos de código) para permitir que el usuario interactúe con sus objetos (capas, función o interfaz). QGIS también tiene una consola Python.

The QGIS Python Console is an interactive shell for Python command executions. It also has a Python file editor that allows you to edit and save your Python scripts. Both console and editor are based on PyQScintilla2 package. To open the console go to *Plugins* ► *Python Console* (`Ctrl+Alt+P`) or click on the Python Console icon in the *Plugins* toolbar.

25.3.1 La consola interactiva

La consola es un intérprete de Python que permite ejecutar comandos de Python. Los módulos de QGIS (analysis, core, gui, server, processing, 3d) y Qt (QtCore, QtGui, QtNetwork, QtWidgets, QtXml) así como los módulos math, os, re y sys de Python ya están importados y se pueden utilizar directamente.

La consola interactiva esta compuesta por una barra de herramientas, un área de entrada y una de salida.



Figura 25.17: Consola de Python

Barra de herramietas

La barra de herramientas propone las siguientes herramientas:

- Limpiar consola para limpiar el área de salida;
- Ejecutar Comando disponible en el área de entrada: igual que pulsar `Intro`;
- Mostrar editor: conmuta la visibilidad *El Editor de código*;
- Opciones...: abre un diálogo para configurar *propiedades de la consola*;
- Ayuda... proporciona un menú para acceder a diversa documentación:
  - *Ayuda Consola Python* (la página actual)
  - Documentación de la API PyQGIS
  - PyQGIS Cookbook

-  Acoplar Editor de Código para acoplar o desacoplar el panel en la interfaz QGIS

Área de entrada

Las principales prestaciones del área de introducción de datos de la consola son:

- Completado de código, resaltado de sintaxis y atajos para las siguientes APIs:
  - Python
  - PyQGIS
  - PyQt5
  - QScintilla2
  - osgeo-gdal-ogr
  - osgeo-geos
- `Ctrl+Alt+Espacio` para ver la lista de autocompletado habilitados en *Configuración de Python*;
- Ejecute fragmentos de código desde el área de entrada escribiendo y presionando `Enter` o *Ejecutar comando*;
- Ejecute fragmentos de código desde el área de salida usando :guilabel:‘Introducir seleccionado` del menú contextual o presionando `Ctrl+E`;
- Examine el historial de comandos desde el área de entrada usando las teclas de flecha `Arriba` y `Abajo` y ejecute el comando que desee;
- `Ctrl+Shift+Espacio` para ver el historial de comandos: haga doble clic en una fila para ejecutar el comando. También se puede acceder al diálogo *Historial de comandos* desde el menú contextual del área de entrada;
- Guarde y borre el historial de comandos. El historial se guardará en el archivo `console_history.txt` en la carpeta activa *user profile*;
- Escriba los siguientes comandos especiales:
  - `?` para mostrar una ayuda de la Consola Python
  - `_api` para abrir la documentación QGIS C++ API o `_api(object)` para la documentación de un objeto específico (en la documentación de QGIS C++ API o Qt API)
  - `_pyqgis` para abrir la documentación de API Python QGIS o `_pyqgis(object)` para la documentación de un objeto específico (en la documentación de la API Python QGIS o la API Qt).
  - `_cookbook` para abrir PyQGIS Cookbook.
  - `!` seguido de un comando para ejecutar comandos Shell desde la Consola de Python. La consola iniciará un subproceso, y reenviará su salida a la Salida de la Consola Python. Mientras se ejecuta el subproceso, la Entrada de la Consola Python cambia al modo STDIN y reenvía los caracteres introducidos al proceso hijo. Esto hace posible enviar confirmación cuando el programa hijo la pide. Cuando la Consola está en modo STDIN, pulsando `Ctrl+C` se mata el subproceso. También es posible afectar el resultado de un comando a una variable con la sintaxis `var = !cmd`.

```
>>> !echo QGIS Rocks!
QGIS Rocks

>>> !gdalinfo --version
GDAL 3.10.3, released 2025/04/01

>>> !pip install black
# Install black python formatter using pip (if available)

>>> sql_formats = !ogrinfo --formats | grep SQL
>>> sql_formats
```

(continúe en la próxima página)

(proviene de la página anterior)

```
['SQLite -vector- (rw+v): SQLite / Spatialite', '  MSSQLSpatial -vector-␣
↪(rw+): Microsoft SQL Server Spatial Database', '  PostgreSQL -vector-␣
↪(rw+): PostgreSQL/PostGIS', '  MySQL -vector- (rw+): MySQL', '  PGDUMP -
↪vector- (w+v): PostgreSQL SQL dump']
```

Truco

Reutilizar comandos ejecutados desde el panel de salida

Puede ejecutar fragmentos de código desde el panel de salida seleccionando un texto y presionando `Ctrl+E`. No importa si el texto seleccionado contiene la indicación del intérprete (`>>>`, `...`).

25.3.2 El Editor de código

Use the Show Editor button in the Interactive Console to enable the editor widget. It allows editing and saving Python scripts and offers advanced functionalities to manage your code. Depending on the *enabled settings*, it provides various capabilities for easier code writing, such as code completion, highlighting syntax and calltips for supported APIs. Automatic indentation, parenthesis insertion, code commenting and syntax checking are also available.

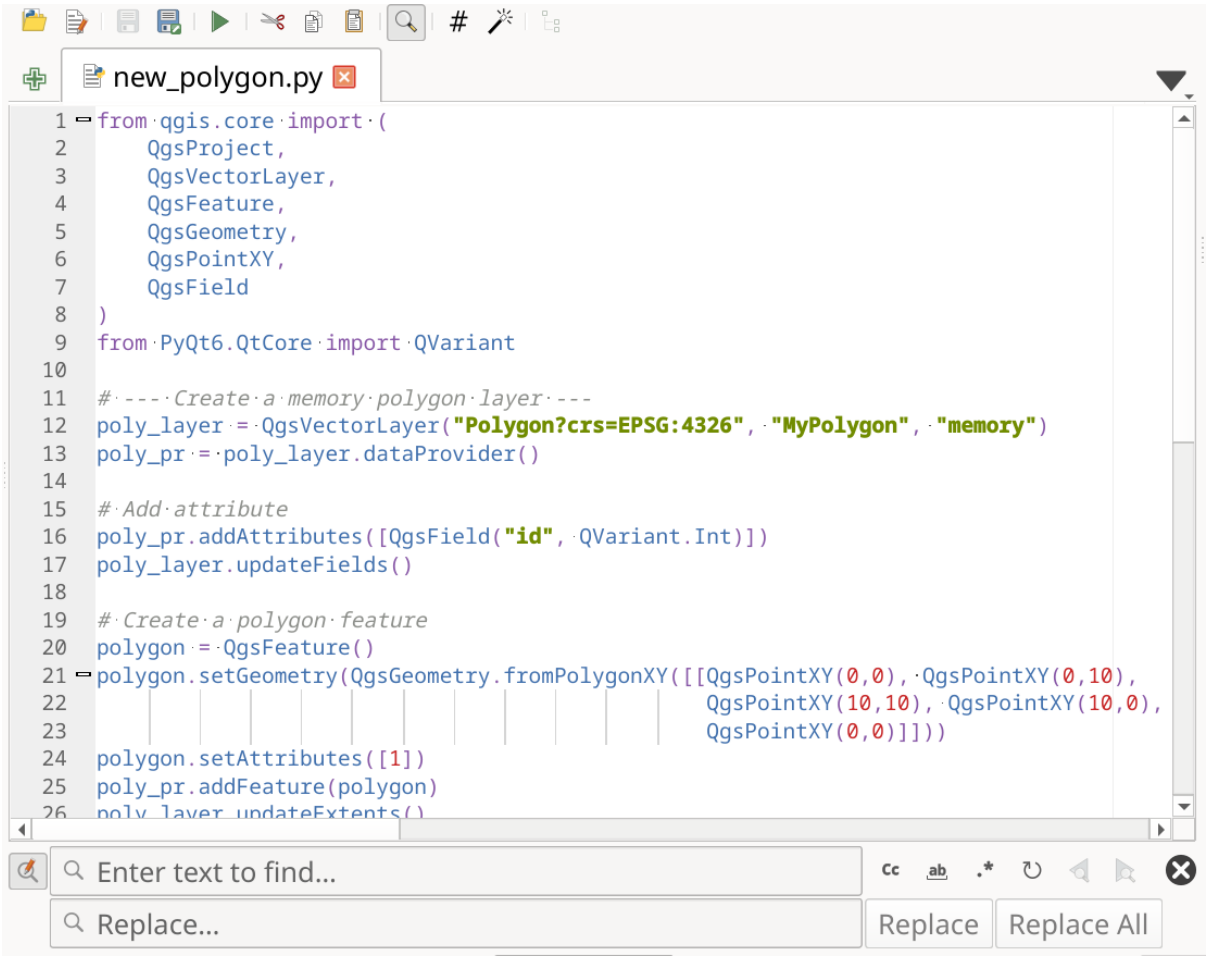


Figura 25.18: El editor de la consola de Python

The code editor area allows to simultaneously work on different scripts, each in a specific tab. Press New editor to add a new tab. You can run partially or totally a script from the *Code Editor* and output the result in the Interactive

Console output area.

Truco

Press `Ctrl+Space` to view the auto-completion list.

At the top of the dialog, a toolbar provides access to a few commands. Right-clicking the editor area provides some more options. All available tools are described in the following table.

Tabla 25.1: Python Console Editor Tools

Tool name	Function	Location
Open script…	Loads a Python file in the code editor dialog, as a new tab	Barra de herramietas
Open in external editor…	Opens a saved Python script in the default external program set for Python file editing	
Save (`Ctrl+S`)	Saves the current script	
Save as… (`Ctrl+Shift+S`)	Saves the current script as a new file	
Run script (`Ctrl+Shift+E`)	Executes the whole script in the Interactive console (this creates a byte-compiled file with the extension `.pyc`)	Toolbar & Contextual menu
Context help (`F1`)	Attempts to display help on the selected string (class, method, object,…) in its corresponding API documentation	Contextual menu
Run selected (`Ctrl+E`)	Executes in the Interactive console the lines selected in the script	Toolbar & Contextual menu
Cut (`Ctrl+X`)	Cuts selected text to the clipboard	
Copy (`Ctrl+C`)	Copies selected text to the clipboard	
Paste (`Ctrl+V`)	Pastes a cut or copied text	
Find text	Allows to search and replace a text in the script. • Utilice el acceso directo del entorno de escritorio predeterminado para buscar el siguiente/anterior (`Ctrl+G` y `Shift+Ctrl+G`); • Automáticamente encuentra la primera coincidencia al escribir en el cuadro de búsqueda; • Establecer la cadena de búsqueda inicial a la selección al abrir la búsqueda; • Habilite la opción *Usar expresiones regulares* en la barra de búsqueda para utilizar expresiones regulares en lugar de la coincidencia de texto sin formato; • Replace the matching text with the text in the replace box. When using a regular expression, you can reference captured groups (e.g. via `\1`) in the replacement text. Use *Replace All* to replace all matches at once. • Presionando `Esc` cierra la barra de búsqueda.	

continúe en la próxima página

Tabla 25.1 – proviene de la página anterior

Tool name	Function	Location
# *Toggle comment* (`Ctrl+:`)	Comments out or uncomment selected lines, by adding or removing `#` character (potentially followed by a space) at the start of the lines. If the selection contains both comments and actual commands, adds `#` at the start of each non empty line. Pressing again the button reverts to the initial state.	
Reformat code (`Ctrl+Alt+F`)	Allows to manually apply various formatting rules (sort imports, indentation, line length,…) to the code, following *user-defined settings*. This may require installation of additional Python modules.	
Object inspector…	Shows and hides a dedicated browser with a tree structure for classes and functions available in the script. Click an item fo a quick access to its definition. The tool requires an activation from the *Python settings - Run and Debug*.	
Hide editor	Hides the Python code editor block. To make it visible again, press Show editor button from the interactive console toolbar.	Contextual menu
Check syntax (`Ctrl+4`)	Browses the code and reports syntax errors, such as missing parenthesis, colons, wrong indentation,…	
Undo (`Ctrl+Z`)	Undoes the latest action	
Redo (`Ctrl+Shift+Z`)	Reverts undone actions to a more recent	
Select all (`Ctrl+A`)	Selects the whole script	
Share on GitHub	Shares the script as a *Secret Gist* or *Public Gist* on GitHub, provided a *GitHub access token*.	
Options…	Opens the *Configuración de Python* dialog.	

CAPÍTULO 26

Apéndices

26.1 Apéndice C: Formatos de Archivo de QGIS

26.1.1 QGS/QGZ - El Formato de Archivo de Proyecto de QGIS

El formato **QGS** es un formato XML para el almacenamiento de proyectos QGIS. El formato **QGZ** es un archivo comprimido (zip) conteniendo un fichero QGS y un fichero QGD. El fichero **QGD** es una base de datos sqlite asociada del proyecto qgisque contiene datos auxiliares para el proyecto. Si no hay datos auxiliares, el fichero QGD estará vacío.

Un archivo de QGIS contiene todo lo necesario para almacenar un proyecto de QGIS, incluyendo:

- título del proyecto
- SCR del proyecto
- el árbol de capas
- configuración de ajuste
- relaciones
- la extensión del mapa del lienzo
- modelos de proyecto
- legend
- muelles de vista de mapa (2Dy 3D)
- las capas con enlaces a los conjuntos de datos subyacentes (fuentes de datos) y otras propiedades de la capa, incluida la extensión, SRS, uniones, estilos, renderizador, modo de fusión, opacidad y más.
- propiedades de proyecto

Las figuras siguientes muestran las etiquetas de nivel superior en un archivo QGS y la etiqueta ampliada `ProjectLayers`.

```
−<qgis version="3.4.13-Madeira" projectname="">
    <homePath path=""/>
    <title/>
    <autotransaction active="0"/>
    <evaluateDefaultValues active="0"/>
    <trust active="0"/>
  +<projectCrs></projectCrs>
  +<layer-tree-group></layer-tree-group>
  +<snapping-settings tolerance="12" unit="1" enabled="0" type="1" mode="2" intersection-snapping="0">
    </snapping-settings>
    <relations/>
  −<mapcanvas name="theMapCanvas" annotationsVisible="1">
      <units>meters</units>
    +<extent></extent>
      <rotation>0</rotation>
    +<destinationsrs></destinationsrs>
      <rendermaptile>0</rendermaptile>
    </mapcanvas>
    <projectModels/>
  +<legend updateDrawingOrder="true"></legend>
    <mapViewDocks/>
    <mapViewDocks3D/>
  +<projectlayers></projectlayers>
  +<layerorder></layerorder>
  +<properties></properties>
    <visibility-presets/>
    <transformContext/>
  +<projectMetadata></projectMetadata>
    <Annotations/>
    <Layouts/>
  </qgis>
```

Figura 26.1: Las etiquetas de nivel superior en un archivo QGIS

```
−<projectlayers>
 −<maplayer styleCategories="AllStyleCategories" readOnly="0" autoRefreshTime="0" autoRefreshEnabled="0" refreshOnNotifyEnabled="0" maxScale="0"
  geometry="Polygon" labelsEnabled="0" type="vector" simplifyDrawingHints="1" hasScaleBasedVisibilityFlag="0" simplifyDrawingTol="1"
  simplifyMaxScale="1" minScale="1e+8" simplifyAlgorithm="0" simplifyLocal="1" refreshOnNotifyMessage="">
  +<extent></extent>
   <id>watersheds_b62efa19_8809_4406_b6ec_2951ac4c94c5</id>
  −<datasource>
     ./QGIS-Training-Data-2.0/exercise_data/processing/generalize/watersheds.shp
   </datasource>
  +<keywordList></keywordList>
   <layername>watersheds</layername>
  +<srs></srs>
  +<resourceMetadata></resourceMetadata>
   <provider encoding="UTF-8">ogr</provider>
   <vectorjoins/>
   <layerDependencies/>
   <dataDependencies/>
   <legend type="default-vector"/>
   <expressionfields/>
  +<map-layer-style-manager current="default"></map-layer-style-manager>
   <auxiliaryLayer/>
  +<flags></flags>
  +<renderer-v2 symbollevels="0" enableorderby="0" type="singleSymbol" forceraster="0"></renderer-v2>
  +<customproperties></customproperties>
   <blendMode>0</blendMode>
   <featureBlendMode>0</featureBlendMode>
   <layerOpacity>1</layerOpacity>
  +<SingleCategoryDiagramRenderer diagramType="Histogram" attributeLegend="1"></SingleCategoryDiagramRenderer>
  +<DiagramLayerSettings priority="0" linePlacementFlags="18" dist="0" showAll="1" placement="1" obstacle="0" zIndex="0"></DiagramLayerSettings>
  +<geometryOptions removeDuplicateNodes="0" geometryPrecision="0"></geometryOptions>
  +<fieldConfiguration></fieldConfiguration>
  +<aliases></aliases>
   <excludeAttributesWMS/>
   <excludeAttributesWFS/>
  +<defaults></defaults>
  +<constraints></constraints>
  +<constraintExpressions></constraintExpressions>
   <expressionfields/>
  +<attributeactions></attributeactions>
  +<attributetableconfig actionWidgetStyle="dropDown" sortExpression="" sortOrder="0"></attributetableconfig>
  +<conditionalstyles></conditionalstyles>
   <editform tolerant="1"/>
   <editforminit/>
   <editforminitcodesource>0</editforminitcodesource>
   <editforminitfilepath/>
  +<editforminitcode></editforminitcode>
   <featformsuppress>0</featformsuppress>
   <editorlayout>generatedlayout</editorlayout>
  +<editable></editable>
  +<labelOnTop></labelOnTop>
   <widgets/>
   <previewExpression>ID</previewExpression>
   <mapTip/>
  </maplayer>
 </projectlayers>
```


Figura 26.2: La etiqueta ProjectLayers de nivel superior expandido de un archivo QGS

26.1.2 QLR - El archivo de Definición de Capa de QGIS

Un archivo de definición de capa (QLR) es un archivo XML que contiene un puntero a la fuente de datos de la capa además de la información de estilo QGIS para la capa.

El caso de uso de este archivo es simple: tener un solo archivo para abrir una fuente de datos y traer toda la información de estilo relacionada. Los archivos QLR también le permiten enmascarar la fuente de datos subyacente en un archivo fácil de abrir.

Un ejemplo de uso de QLR es para abrir capas de MS SQL. En lugar de tener que ir al diálogo de conexión de MS SQL, conectar, seleccionar, cargar y finalmente diseñar, simplemente puede agregar un archivo .qlr que apunte a la capa correcta de MS SQL con todo el estilo necesario incluido.

En el futuro, un archivo .qlr puede contener una referencia a más de una capa.

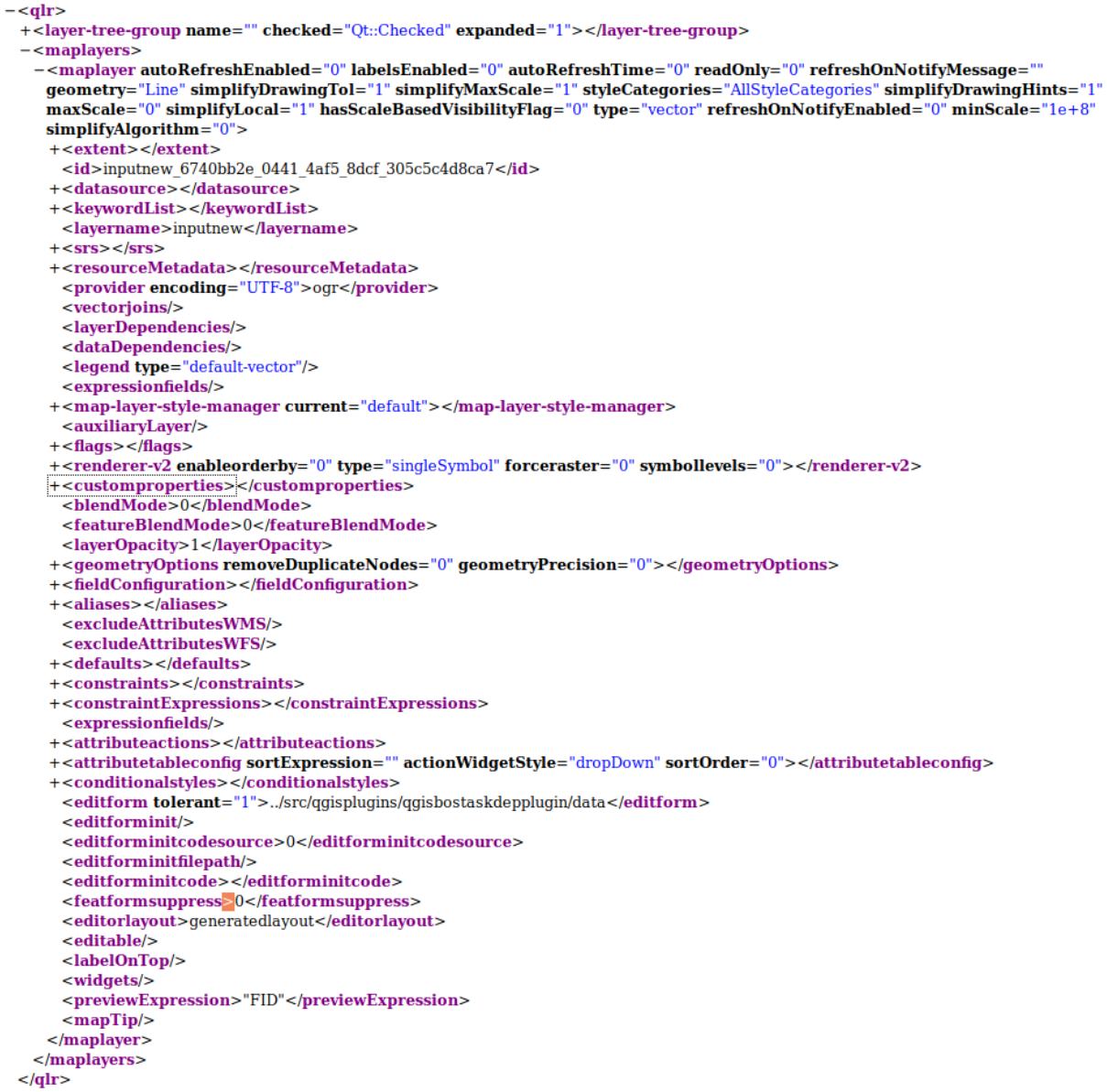


Figura 26.3: Las etiquetas de nivel superior de un archivo QLR

26.1.3 QML: el formato de archivo de estilo QGIS

QML es un formato XML para almacenar estilos de capa.

Un archivo QML contiene toda la información que QGIS puede manejar para la representación de geometrías de características, incluidas definiciones de símbolos, tamaños y rotaciones, etiquetado, opacidad y modo de fusión y más.

La siguiente figura muestra las etiquetas de nivel superior de un archivo QML (con solo `renderer_v2` y su etiqueta de `símbolo` expandida).

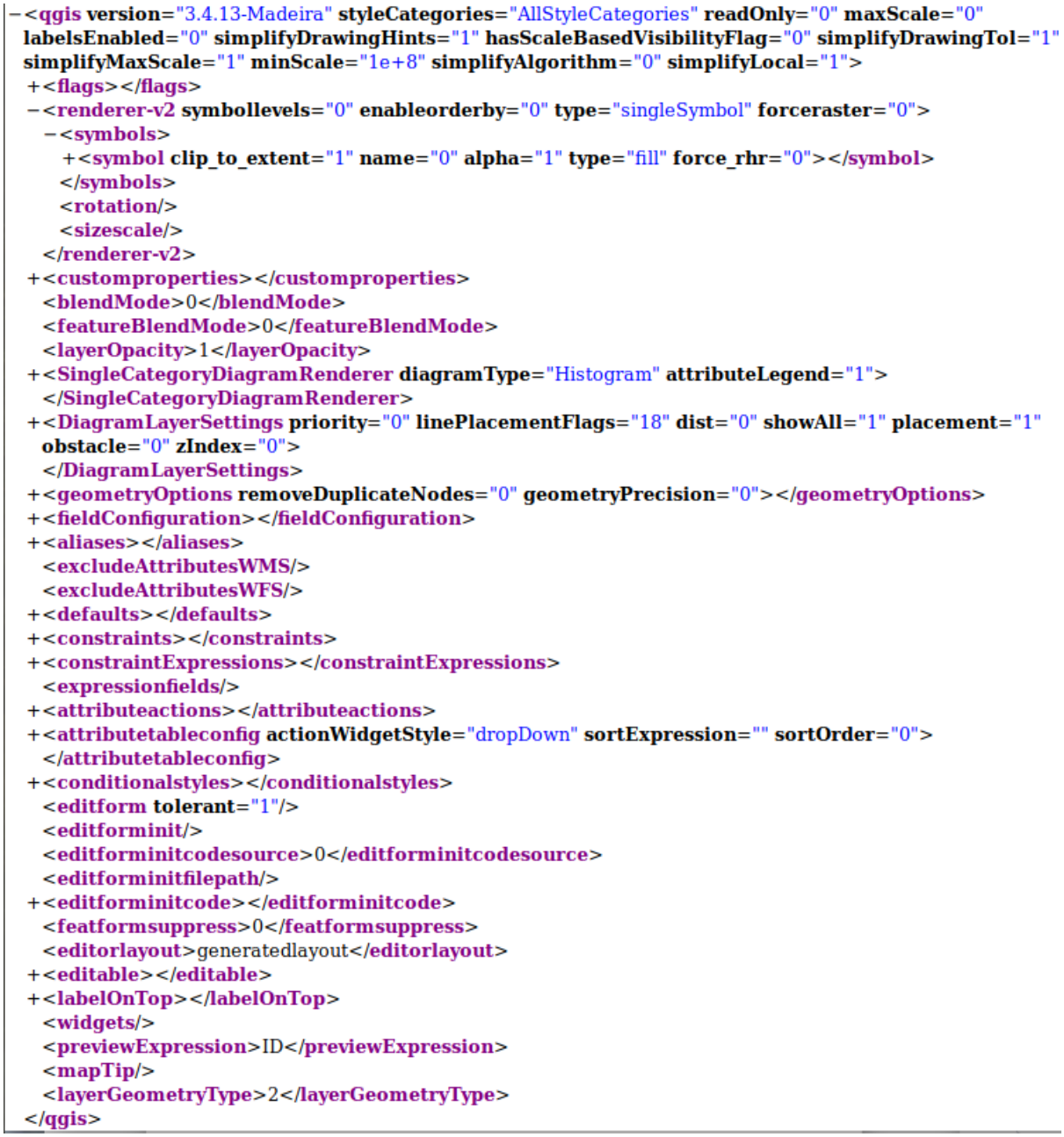


Figura 26.4: Las etiquetas de nivel superior de un archivo QML (solo se expande la etiqueta renderer_v2 con su etiqueta de símbolo)

26.2 Apéndice D: sintaxis del script QGIS R

Contributed by Matteo Ghetta - funded by Scuola Superiore Sant'Anna

Escribir scripts R en Procesos es un poco complicado debido a la sintaxis especial.

Un script Processing R comienza con la definición de sus **Entradas** y **Salidas**, cada una precedida por caracteres de doble hash (##).

Antes de las entradas, se puede especificar el grupo en el que colocar el algoritmo. Si el grupo ya existe, se le agregará el algoritmo; de lo contrario, se creará el grupo. En el siguiente ejemplo, el nombre del grupo es *Mi grupo*:

```
##My Group=group
```

26.2.1 Entradas

Deben especificarse todos los datos de entrada y los parámetros. Hay varios tipos de entradas:

- vectorial: `##Layer = vector`
- campo vectorial: `##F = Field Layer` (donde *Layer* es el nombre de una capa vectorial entrante a la que pertenece el campo)
- ráster: `##r = raster`
- tabla: `##t = table`
- número: `##Num = number`
- cadena: `##Str = string`
- booleano: `##Bol = boolean`
- elementos en un menú desplegable. Los elementos deben estar separados por punto y coma. `;`: `##type=selection point;lines;point+lines`

26.2.2 Salidas

Igual que para las entradas, cada salida se tiene que definir al principio del script:

- vectorial: `##output= output vector`
- ráster: `##output= output raster`
- tabla: `##output= output table`
- gráficos: `##output_plots_to_html` (##showplots en versiones anteriores)
- Para mostrar la salida de R en el *Visor de resultados*, coloque > delante del comando cuya salida le gustaría mostrar.

26.2.3 Resumen de sintaxis para scripts QGIS R

Se ofrecen varios tipos de parámetros de entrada y salida.

Tipos de parámetros de entrada

Parámetros	Ejemplos de sintaxis	Devolviendo objetos
Vector	Capa = vector	objeto sf (u objeto SpatialDataFrame, si se ha especificado ##load_vector_using_rgdal)
punto vector	Capa = vectorial de puntos	objeto sf (u objeto SpatialDataFrame, si se ha especificado ##load_vector_using_rgdal)
Vectorial lineal	Capa = vectorial lineal	objeto sf (u objeto SpatialDataFrame, si se ha especificado ##load_vector_using_rgdal)
polígono vector	Capa = vectorial poligonal	sf object (o SpatialPolygonsDataFrame object, si se usa ##load_vector_using_rgdal)
multiple vector	Capa = vectorial múltiple	sf object (u objetos SpatialDataFrame si se especifica ##load_vector_using_rgdal)
tabla	Capa = tabla	conversión de dataframe desde csv, objeto predeterminado de la función `read.csv`
campo	Campo = Capa de campo	nombre del campo seleccionado, p.ej. `"Area"`
raster	Capa= raster	Objeto RasterBrick, objeto predeterminado de paquete `ráster`
ráster múltiple	Capa = ráster multiple	Objetos RasterBrick, objeto predeterminado de paquete `ráster`
número	N = número	número entero o flotante elegido
cadena	S = Cadena	cadena agregada en el cuadro
longstring	LS = longstring	cadena agregada en el cuadro, podría ser más larga que la cadena normal
selección	S = selección primera;segunda;tercera	cadena del elemento seleccionado elegido en el menú desplegable
src	C = src	cadena del CRS resultante elegido, en el formato: `"EPSG:4326"`
extensión	E = extensión	Extensión de objeto de paquete ráster, puede extraer valores con `E@xmin`
punto	P = punto	Al hacer click en el mapa, tiene las coordenadas del punto
archivo	F = archivo	ruta del archivo elegido, p. ej. «/home/matteo/file.txt»
folder	F = carpeta	ruta del directorio elegido, p. ej. «/home/matteo/Downloads»

Un parámetro puede ser **OPCIONAL**, lo que significa que puede ser ignorado.

Con el fin de establecer una entrada como opcional, puede añadir la cadena `optional` **antes** de la entrada, p.ej.:

```
##Layer = vector
##Field1 = Field Layer
##Field2 = optional Field Layer
```

Tipos de parámetros de salida

Parámetros	Ejemplos de sintaxis
Vector	Salida = vector de salida
raster	Salida = salida de ráster
tabla	Salida = tabla de salida
archivo	Salida = archivo de salida

Nota

Puedes grabar gráficos como `png` desde el **Visor de Resultados de Procesos**, o puedes escoger guardar el gráfico directamente desde la interfaz del algoritmo.

Cuerpo de lineas de codigo

El cuerpo del script sigue la sintaxis de R y el panel de **Registro** puede ayudarte si hay algo mal en tu script.

Recuerda que tienes que cargar todas las librerías adicionales en el script:

```
library(sp)
```

26.2.4 Ejemplos

Ejemplo con salida vectorial

Tomemos un algoritmo de la colección en línea que crea puntos aleatorios a partir de la extensión de una capa de entrada:

```
##Point pattern analysis=group
##Layer=vector polygon
##Size=number 10
##Output=output vector
library(sp)
spatpoly = as(Layer, "Spatial")
pts=spsample(spatpoly,Size,type="random")
spdf=SpatialPointsDataFrame(pts, as.data.frame(pts))
Output=st_as_sf(spdf)
```

Explicación (por línea en el script):

1. **Análisis de patrón de puntos** es el grupo del algoritmo
2. `Layer` es la capa **vectorial** entrante
3. `Size` es un parámetro **numérico** con un valor por defecto de 10
4. `Output` es la capa **vectorial** que creará el algoritmo
5. `library(sp)` carga la libreria **sp**
6. `spatpoly = as(Layer, "Spatial")` traduce a un objeto sp
7. Llama a la función `spsample` de la librería `sp` y la ejecuta usando la entrada definida antes (`Layer` y `Size`)
8. Crea un objeto *SpatialPointsDataFrame* usando la función `SpatialPointsDataFrame`
9. Crear la capa vectorial saliente usando la función `st_as_sf`

¡Eso es! Simplemente ejecute el algoritmo con una capa vectorial que tenga en la leyenda de QGIS, elija el número de puntos aleatorios. La capa resultante se agregará a su mapa.

Ejemplo con salida ráster

El siguiente script ejecutará un kriging ordinario básico para crear un mapa ráster de valores interpolados a partir de un campo específico de la capa vectorial de puntos de entrada usando la función `autoKrige` del paquete R `automap`. Primero calculará el modelo kriging y luego creará un ráster. El ráster se crea con la función `ráster` del paquete Ráster R:

```
##Basic statistics=group
##Layer=vector point
##Field=Field Layer
##Output=output raster
##load_vector_using_rgdal
require("automap")
require("sp")
```

(continúe en la próxima página)

(proviene de la página anterior)

```
require("raster")
table=as.data.frame(Layer)
coordinates(table)= ~coords.x1+coords.x2
c = Layer[[Field]]
kriging_result = autoKrige(c~1, table)
prediction = raster(kriging_result$krige_output)
Output<-prediction
```

Usando `##load_vector_using_rgdal`, la capa vectorial entrante estará disponible como objetos `SpatialDataFrame`, por lo que evitamos tener que traducirlo desde un objeto `sf`.

Ejemplo con salida de tabla

Editemos el algoritmo de `Estadísticas de resumen` para que la salida sea un archivo de tabla (csv).

El cuerpo del script es el siguiente:

```
##Basic statistics=group
##Layer=vector
##Field=Field Layer
##Stat=Output table
Summary_statistics<-data.frame(rbind(
    sum(Layer[[Field]]),
    length(Layer[[Field]]),
    length(unique(Layer[[Field]])),
    min(Layer[[Field]]),
    max(Layer[[Field]]),
    max(Layer[[Field]])-min(Layer[[Field]]),
    mean(Layer[[Field]]),
    median(Layer[[Field]]),
    sd(Layer[[Field]]))),
  row.names=c("Sum:","Count:","Unique values:","Minimum value:","Maximum value:",
↪"Range:","Mean value:","Median value:","Standard deviation:"))
colnames(Summary_statistics)<-c(Field)
Stat<-Summary_statistics
```

La tercera línea especifica la **Campo Vectorial** en la entrada y la cuarta línea le dice al algoritmo que la salida debe ser una tabla.

La última línea tomará el objeto `Stat` creado en el script y lo convierte en una tabla `csv`.

Ejemplo con la consola de salida

Podemos usar el ejemplo anterior y en lugar de crear una tabla, imprimir el resultado en el **Visor de resultados**:

```
##Basic statistics=group
##Layer=vector
##Field=Field Layer
Summary_statistics<-data.frame(rbind(
sum(Layer[[Field]]),
length(Layer[[Field]]),
length(unique(Layer[[Field]])),
min(Layer[[Field]]),
max(Layer[[Field]]),
max(Layer[[Field]])-min(Layer[[Field]]),
mean(Layer[[Field]]),
median(Layer[[Field]]),
sd(Layer[[Field]])),row.names=c("Sum:","Count:","Unique values:","Minimum value:",
↪"Maximum value:","Range:","Mean value:","Median value:","Standard deviation:"))
```

(continúe en la próxima página)

(proviene de la página anterior)

```
colnames(Summary_statistics)<-c(Field)
>Summary_statistics
```

El script es exactamente el mismo que el de arriba excepto por dos ediciones:

1. sin salida especificada (la cuarta línea ha sido eliminada)
2. la última línea comienza con >, indicando a Processing que haga que el objeto esté disponible a través del visor de resultados

Ejemplo con gráfico

Para crear gráficos, debe usar el parámetro `##output_plots_to_html` como en el siguiente script:

```
##Basic statistics=group
##Layer=vector
##Field=Field Layer
##output_plots_to_html
####output_plots_to_html
qqnorm(Layer[[Field]])
qqline(Layer[[Field]])
```

El script usa un campo (`Field`) de una capa vectorial (`Layer`) como entrada y crea un *Gráfico QQ* (para probar la normalidad de la distribución).

El gráfico se agrega automáticamente al *Visor de resultados* de procesos.

26.3 Apéndice E: Conexiones de red de la aplicación QGIS

Esta es una lista de las conexiones de red preconfiguradas/automatizadas que hace QGIS. Algunas de las conexiones son iniciadas por el usuario porque requieren una acción del usuario antes de que se lleven a cabo, otras suceden automáticamente.

Nombre	Propósito	Modo UX	Servidor	Información enviada al servidor	Información guardada en el servidor
qgis.org					
Ayuda sobre la API (Interfaz de programación de aplicaciones) de Python	Navegador de la documentación de PyQGIS	Usuario iniciado	https://qgis.org/pyqgis/3.44/search	IP, versión de QGIS version, OS	IP en registro del servidor
version.qgis.org					
Verificar nueva versión	Notificar al usuario sobre la disponibilidad de nuevas versiones de QGIS	Automátic	https://version.qgis.org	IP, versión de QGIS version, OS	IP en registro del servidor
feed.qgis.org					
Fuente QGIS	Recuperar noticias desde la fuente QGIS	Automátic	https://feed.qgis.org	IP, versión de QGIS, código de idioma, fecha de la última descarga, sistema operativo	IP en el registro del servidor; la versión de QGIS, el sistema operativo y la IP se agregan y se utilizan para recopilar algunas estadísticas
plugins.qgis.org					
Verificar actualizaciones de complementos	Notificar al usuario de actualizaciones de complementos	Usuario iniciado/a (configura	https://plugins.qgis.org	IP, versión de QGIS version, OS	IP en registro del servidor
Lista de complementos	Recuperar la lista de complementos	Usuario iniciado	https://plugins.qgis.org	IP, versión de QGIS version, OS	IP en registro del servidor
Instalación del complemento	Descargar e instalar el paquete del complemento	Usuario iniciado	https://plugins.qgis.org	IP, versión de QGIS version, OS	Incrementar el contador de descarga del complemento por uno
Estilos	Lista de estilos aportados por el usuario	Usuario iniciado	https://hub.qgis.org/styles	IP, versión de QGIS version, OS	Incrementar el contador de descarga por uno
De terceros					
Datos del terreno	Generar un MDE a partir de vistas 3D	Usuario iniciado	https://s3.amazonaws.com/elevation-tiles-p terrarium/{z}/{x	IP, versión de QGIS version, OS	Ver Amazon TOS
Geocodificador Google Map	Servicios de geocodificación	Usuario iniciado	https://maps.googleapis.com/maps/api/geocode/json	IP, versión de QGIS version, OS	Ver API TOS de google maps
Geocodificador Nominatim	Servicios de geocodificación	Usuario iniciado	https://nominatim.qgis.org/ui/search.html	IP, versión de QGIS version, OS	
Rejilla geodésica	Agregar una nueva rejilla PROJ	Usuario injciado	https://cdn.proj.org	IP, versión PROJ	Los registros de acceso son permanentemente eliminados transcurrido un día

CAPÍTULO 27

Referencias bibliográficas y web

GDAL-SOFTWARE-SUITE. Geospatial data abstraction library. https://gdal.org/en/latest, 2013.

GRASS-PROJECT. Geographic resource analysis support system. https://grass.osgeo.org, 2013.

NETELER, M., AND MITASOVA, H. Open source gis: A grass gis approach, 2008.

OGR-SOFTWARE-SUITE. Geospatial data abstraction library. https://gdal.org/en/latest, 2013.

OPEN-GEOSPATIAL-CONSORTIUM. Web map service (1.1.1) implementation specification. https://portal.ogc.org/files/?artifact_id=1081&version=1&format=pdf, 2002.

OPEN-GEOSPATIAL-CONSORTIUM. Web map service (1.3.0) implementation specification. https://portal.ogc.org/files/?artifact_id=14416&format=pdf, 2004.

POSTGIS-PROJECT. Spatial support for postgresql. http://www.refractions.net/products/postgis/, 2013.