QGIS Desktop 3.44 User Guide

QGIS Project

14.03.2026

Inhaltsverzeichnis

1 Erste Schritte **3**
- 1.1 QGIS installieren . . . 3
  - 1.1.1 Installation mit Binärdateien . . . 3
  - 1.1.2 Kompilieren des Quellcodes . . . 3
  - 1.1.3 Auf externen Medien installieren . . . 4
  - 1.1.4 Beispieldaten herunterladen . . . 4
- 1.2 QGIS starten und beenden . . . 4
- 1.3 Beispielsitzung: Laden von Raster- und Vektorlayern . . . 5

2 Arbeiten mit Projektdateien **13**
- 2.1 Vorstellung von QGIS-Projekten . . . 13
- 2.2 Behandlung von fehlerhaften Dateipfaden . . . 15
- 2.3 Ausgaben erzeugen . . . 16

3 Die QGIS Oberfläche **17**
- 3.1 Menüleiste . . . 18
  - 3.1.1 Projekt . . . 18
  - 3.1.2 Bearbeiten . . . 19
  - 3.1.3 Sicht (View) . . . 22
  - 3.1.4 Layer . . . 24
  - 3.1.5 Einstellungen . . . 27
  - 3.1.6 Erweiterungen . . . 27
  - 3.1.7 Vektor . . . 28
  - 3.1.8 Raster . . . 29
  - 3.1.9 Datenbank . . . 31
  - 3.1.10 Web . . . 31
  - 3.1.11 Netz . . . 32
  - 3.1.12 Processing . . . 32
  - 3.1.13 Hilfe . . . 32
  - 3.1.14 QGIS . . . 33
- 3.2 Bedienfelder und Werkzeugkästen . . . 33
  - 3.2.1 Werkzeugleisten . . . 33
  - 3.2.2 Bedienfelder . . . 35
- 3.3 Statusleiste . . . 35
  - 3.3.1 Suchleiste . . . 35
  - 3.3.2 Anzeige von Aktionen . . . 36
  - 3.3.3 Steuerung der Kartenansicht . . . 36
  - 3.3.4 Meldungen . . . 37

4 Das Browser Bedienfeld **39**
- 4.1 Ressourcen, die über den Browser geöffnet/ausgeführt werden können . . . 41

4.2 Die obersten Einträge des Brower-Bedienfelds . . . 42
4.2.1 Favoriten . . . 42
4.2.2 Räumliche Lesezeichen . . . 42
4.2.3 Projektverzeichnis . . . 42
4.2.4 Laufwerke und Dateisystem . . . 42
4.2.5 Datentyp . . . 43
4.2.6 Datenbankeinträge . . . 44
4.2.7 Kacheln und Internetdienste . . . 47
4.2.8 Projekt und unterstützte Dateitypen . . . 49

5 QGIS anpassen **51**
5.1 Optionen . . . 51
5.1.1 Allgemein . . . 52
5.1.2 System . . . 54
5.1.3 Benutzerprofile . . . 56
5.1.4 KBS und Transformation . . . 56
5.1.5 Datenquellen . . . 59
5.1.6 Darstellung . . . 63
5.1.7 Karte & Legende . . . 66
5.1.8 Kartenwerkzeuge . . . 68
5.1.9 Höhe . . . 72
5.1.10 3D-Einstellungen . . . 72
5.1.11 Farben . . . 73
5.1.12 Schriftarten . . . 74
5.1.13 Layouts . . . 75
5.1.14 Variablen . . . 76
5.1.15 Authentifizierung . . . 76
5.1.16 Netzwerk . . . 77
5.1.17 GPS . . . 79
5.1.18 Suche . . . 81
5.1.19 Beschleunigung . . . 83
5.1.20 IDE . . . 83
5.1.21 Verarbeitung . . . 87
5.1.22 Erweitert . . . 88
5.2 Arbeiten mit Benutzerprofilen . . . 88
5.2.1 Das Konzept . . . 88
5.2.2 Benutzerprofil einstellen . . . 90
5.3 Projekteigenschaften . . . 90
5.3.1 Allgemein . . . 91
5.3.2 Metadaten . . . 94
5.3.3 Ansichtseinstellungen . . . 95
5.3.4 KBS . . . 95
5.3.5 Transformationen . . . 96
5.3.6 Stile . . . 96
5.3.7 Farben . . . 97
5.3.8 Datenquellen . . . 99
5.3.9 Beziehungen . . . 100
5.3.10 Variablen . . . 101
5.3.11 Makros . . . 101
5.3.12 QGIS Server . . . 102
5.3.13 Zeitlich . . . 104
5.3.14 Geländehöhe . . . 104
5.3.15 Sensoren . . . 106
5.4 Anpassung . . . 108
5.5 Tastenkürzel . . . 110
5.6 QGIS mit erweiterten Einstellungen ausführen . . . 111
5.6.1 Eingabeaufforderung und Umgebungsvariablen . . . 111
5.6.2 Bereitstellen von QGIS innerhalb von Organisationen . . . 117

6 Arbeiten mit Projektionen **119**
6.1 Überblick zur Projektionsunterstützung . . . 119
6.2 Koordinatenbezugssysteme für Layer . . . 119
6.3 Koordinatenbezugssysteme für Projekte . . . 121
6.4 Auswahl des Koordinatenbezugssystems . . . 123
6.5 Ein eigenes Koordinatenbezugssystem definieren . . . 124
6.5.1 Einbinden einer NTv2-Transformation in QGIS . . . 125
6.6 Datumstransformationen . . . 126

7 Karten visualisieren **129**
7.1 2D Kartenansichten . . . 129
7.1.1 Funktionen des Kartenfensters . . . 130
7.1.2 Steuerung der Kartendarstellung . . . 131
7.1.3 Zeitabhängige Darstellung in der Kartenansicht . . . 132
7.1.4 Das Bedienelement für die Höhensteuerung . . . 134
7.1.5 Räumliche Lesezeichen . . . 135
7.1.6 Dekorationen . . . 137
7.1.7 Anmerkungen hinzufügen . . . 144
7.1.8 Messen . . . 150
7.1.9 Weitere Kartenansichten hinzufügen . . . 153
7.1.10 Exportieren der Kartenansicht . . . 154
7.2 3D-Kartenansicht . . . 156
7.2.1 3D-Konfigurationen . . . 158
7.2.2 Navigationsmöglichkeiten . . . 166
7.2.3 Eine Animation erstellen . . . 167
7.2.4 3D-Vektor-Layer . . . 168
7.3 Geländehöhenprofil . . . 168
7.3.1 Die Benutzeroberfläche . . . 168
7.3.2 Erstellen eines Geländehöhenprofils . . . 170
7.3.3 Interaktion mit der Profilansicht . . . 172
7.3.4 Exportieren aus dem Höhenprofil . . . 173

8 Allgemeine Werkzeuge **175**
8.1 Hilfe . . . 175
8.2 Bedienfelder . . . 175
8.2.1 Layer-Bedienfeld . . . 175
8.2.2 Das Layergestaltung-Bedienfeld . . . 184
8.2.3 Das Layerreihenfolge-Bedienfeld . . . 186
8.2.4 Das Übersicht-Bedienfeld . . . 186
8.2.5 Das Protokoll-Bedienfeld . . . 187
8.2.6 Das Rücknahme/Wiederholung-Bedienfeld . . . 187
8.2.7 Das Statistik-Bedienfeld . . . 187
8.2.8 Das Diagnose/Entwicklungswerkzeuge-Bedienfeld . . . 188
8.3 Layer aus externen Projekten einbinden . . . 192
8.4 Interaktion mit Objekten . . . 193
8.4.1 Objekte auswählen . . . 193
8.4.2 Objekte abfragen . . . 196
8.5 Layer-Eigenschaften speichern und teilen . . . 200
8.5.1 Verwalten von benutzerdefinierten Stilen . . . 200
8.5.2 Speichern von Stilen in einer Datei oder einer Datenbank . . . 202
8.5.3 Layerdefinitionsdatei . . . 204
8.6 Dokumentation Ihrer Daten . . . 204
8.6.1 Metadaten . . . 204
8.6.2 Layernotizen . . . 205
8.7 Speichern von Werten in Variablen . . . 206
8.8 Häufige wiederkehrende Werkzeuge . . . 208
8.8.1 Farbauswahl . . . 208
8.8.2 Symbol-Widget . . . 212

8.8.3 Entfernte oder eingebettete Dateiauswahl . . . 213
8.8.4 Einstellen des Sichtbarkeitsmaßstabs . . . 213
8.8.5 Auswahl der räumlichen Ausdehnung . . . 214
8.8.6 Schrift Auswahl . . . 214
8.8.7 Einheiten Auswahl . . . 215
8.8.8 Zahlen Formatierung . . . 217
8.8.9 Mischmodi . . . 218
8.8.10 Einrichten von datendefinierter Übersteuerung . . . 220
8.8.11 Authentifizierung . . . 223

9 Level up mit Ausdrücken **227**
9.1 Ausdrücke . . . 227
9.1.1 Der Ausdruckseditor . . . 227
9.1.2 Funktions Editor . . . 232
9.2 Liste der Funktionen . . . 235
9.2.1 Aggregatsfunktionen . . . 235
9.2.2 Array-Funktionen . . . 245
9.2.3 Farbfunktionen . . . 257
9.2.4 Funktionen für Bedingungen . . . 266
9.2.5 Umwandlungs-Funktionen . . . 269
9.2.6 CRS Functions . . . 275
9.2.7 Benutzerdefinierte Funktionen . . . 275
9.2.8 Datums- und Zeit-Funktionen . . . 275
9.2.9 Felder und Werte . . . 286
9.2.10 Funktionen für Dateien und Pfade . . . 286
9.2.11 Formular-Funktionen . . . 289
9.2.12 Unscharfer Vergleich Funktion . . . 290
9.2.13 Allgemeine Funktionen . . . 291
9.2.14 Geometriefunktionen . . . 293
9.2.15 Layout Funktionen . . . 369
9.2.16 Kartenlayer . . . 370
9.2.17 Zuordnungs-Funktionen . . . 373
9.2.18 Mathematische Funktionen . . . 377
9.2.19 Netz-Layer-Funktionen . . . 385
9.2.20 Operatoren . . . 387
9.2.21 Verarbeitungs-Funktionen . . . 396
9.2.22 Raster-Funktionen . . . 396
9.2.23 Funktionen für Datensätze und Attribute . . . 398
9.2.24 Beziehungen . . . 407
9.2.25 Sensoren-Funktionen . . . 408
9.2.26 Zeichenketten-Funktionen . . . 408
9.2.27 Benutzerdefinierte Ausdrücke . . . 419
9.2.28 Variablen . . . 419
9.2.29 Kürzlich verwendete Funktionen . . . 423

10 Die Stilbibliothek **425**
10.1 Die Stilverwaltung . . . 425
10.1.1 Der Stilverwaltung-Dialog . . . 425
10.1.2 Einstellen einer Farbrampe . . . 431
10.1.3 Ein Legendensymbol erstellen . . . 433
10.2 Die Symbolauswahl . . . 435
10.2.1 Der Symbollayer-Baum . . . 436
10.2.2 Ein Symbol konfigurieren . . . 437
10.3 Einrichten einer Beschriftung . . . 457
10.3.1 Formatierung der Text-Elemente . . . 458
10.3.2 Konfigurieren der Interaktion mit Beschriftungen . . . 468
10.4 3D-Symbole erstellen . . . 480
10.4.1 3D Punkt-Symbole . . . 480

10.4.2 3D Linien-Symbole . . . 481
10.4.3 3D Polygon-Symbole . . . 482
10.4.4 Schattierung der Textur . . . 484
10.4.5 Anwendungsbeispiel . . . 484

11 Verwaltung von Datenquellen **485**
11.1 Öffnen von Daten . . . 485
11.1.1 Das Browser-Bedienfeld . . . 487
11.1.2 Die DB-Verwaltung . . . 490
11.1.3 Anbieterbasierte Ladetools . . . 491
11.1.4 Das Bedienfeld für die Suche nach Layer-Metadaten . . . 513
11.1.5 Eigene QGIS Formate . . . 514
11.1.6 QLR - QGIS Layer Definitionsdatei . . . 514
11.1.7 Verbindung zu Webdiensten . . . 515
11.2 Erstellung von Layern . . . 523
11.2.1 Neue Vektorlayer erstellen . . . 524
11.2.2 Erstellen neuer Layer aus einem vorhandenen Layer . . . 533
11.2.3 Erstellen neuer DXF-Dateien . . . 538
11.2.4 Erstellen neuer Layer über die Zwischenablage . . . 539
11.2.5 Creating SQL Query Layers . . . 540
11.2.6 Erstellen von virtuellen Layern . . . 543
11.3 Georeferenzierung . . . 546
11.3.1 Georeferenzierung von Rasterlayern . . . 546
11.3.2 Georeferenzierung des Vektor-Layers . . . 552
11.3.3 Layer-Eigenschaften anzeigen und anpassen . . . 553
11.3.4 Konfiguration des Georeferenzierers . . . 553
11.4 Datenformate und Felder erkunden . . . 554
11.4.1 Rasterdaten . . . 554
11.4.2 Vektordaten . . . 554
11.4.3 Layer, die den 180° Längengrad kreuzen . . . 563

12 Arbeiten mit Vektordaten **565**
12.1 Der Eigenschaften-Dialog von Vektorlayern . . . 565
12.1.1 Information . . . 566
12.1.2 Quelle . . . 567
12.1.3 Symbolisierung . . . 570
12.1.4 Beschriftungen Eigenschaften . . . 593
12.1.5 Diagrammeigenschaften . . . 606
12.1.6 Eigenschaften Masken . . . 612
12.1.7 3D-Ansicht Eigenschaften . . . 613
12.1.8 Eigenschaften Felder . . . 615
12.1.9 Attributformular . . . 616
12.1.10 Verknüpfungen . . . 628
12.1.11 Hilfsspeicher . . . 628
12.1.12 Aktionen . . . 637
12.1.13 Anzeige . . . 643
12.1.14 Darstellung . . . 646
12.1.15 Zeitlich . . . 648
12.1.16 Variablen . . . 649
12.1.17 Höhe . . . 649
12.1.18 Metadaten . . . 652
12.1.19 Abhängigkeiten . . . 652
12.1.20 Legende . . . 652
12.1.21 QGIS Server . . . 653
12.1.22 Digitalisieren . . . 655
12.2 Arbeiten mit der Attributtabelle . . . 657
12.2.1 Vorbemerkungen: Räumliche und nicht räumliche Tabellen . . . 657
12.2.2 Einführung in die Benutzeroberfläche der Attributtabelle . . . 658

12.2.3 Mit den Objekten in einer Attributtabelle interagieren . . . 663
12.2.4 Bearbeiten von Attributwerten . . . 667
12.2.5 Erkundung von Attribute mit dem Objekte-abfragen-Werkzeug . . . 671
12.3 Editierfunktionen . . . 673
12.3.1 Einrasttoleranz und Suchradius einstellen . . . 674
12.3.2 Optionen für das Einrasten und Digitalisieren . . . 674
12.3.3 Topologisches Editieren . . . 677
12.3.4 Einen vorhandenen Layer editieren . . . 679
12.3.5 Erweiterte Digitalisierung . . . 692
12.3.6 Formen digitalisieren . . . 702
12.3.7 Das Bedienfeld Erweiterte Digitalisierung . . . 705
12.3.8 Mit Verarbeitung Objekte innerhalb eines Layers verändern . . . 714
12.4 Verbinden und Bearbeiten von Daten auf verschiedenen Layern . . . 716
12.4.1 Verknüpfung von Objekten zwischen zwei Layern . . . 716
12.4.2 Festlegung von Beziehungen zwischen mehreren Layern . . . 718
12.4.3 Storing and fetching an external resource . . . 733

13 Arbeiten mit Rasterdaten 737
13.1 Der Eigenschaften-Dialog von Rasterlayern . . . 737
13.1.1 Information . . . 738
13.1.2 Quelle . . . 738
13.1.3 Symbolisierung . . . 739
13.1.4 Transparenz . . . 750
13.1.5 Labels Properties . . . 752
13.1.6 Histogramm . . . 753
13.1.7 Darstellung . . . 754
13.1.8 Eigenschaften Zeitlich . . . 755
13.1.9 Pyramiden . . . 756
13.1.10 Höhe . . . 757
13.1.11 Metadaten . . . 758
13.1.12 Eigenschaften Legende . . . 759
13.1.13 Anzeige . . . 760
13.1.14 QGIS Server . . . 761
13.1.15 Attribute Tables . . . 762
13.1.16 Identifizieren von Rasterzellen . . . 767
13.2 Rasteranalyse . . . 767
13.2.1 Rasterrechner . . . 767

14 Working with Mesh Data 771
14.1 What's a mesh? . . . 771
14.2 Supported formats . . . 773
14.3 Mesh Dataset Properties . . . 774
14.3.1 Information . . . 775
14.3.2 Quelle . . . 776
14.3.3 Symbolisierung . . . 777
14.3.4 Beschriftungen Eigenschaften . . . 782
14.3.5 3D-Ansicht Eigenschaften . . . 784
14.3.6 Darstellung . . . 786
14.3.7 Eigenschaften Zeitlich . . . 787
14.3.8 Höhe . . . 787
14.3.9 Metadaten . . . 789
14.4 Exploring mesh datasets . . . 789
14.5 Editing a mesh layer . . . 790
14.5.1 Overview of the mesh digitizing tools . . . 791
14.5.2 Exploring the Z value assignment logic . . . 791
14.5.3 Selecting mesh elements . . . 793
14.5.4 Modifying mesh elements . . . 794
14.5.5 Reindexing meshes . . . 798

14.6 Mesh Calculator . . . 798

15 Arbeiten mit Vektorkacheln (Vector Tiles) 801
15.1 Was sind Vektorkacheln? . . . 801
15.2 Unterstützte Formate . . . 802
15.3 Eigenschaften von Vektorkacheldatensätzen . . . 803
15.3.1 Information . . . 803
15.3.2 Quelle . . . 803
15.3.3 Eigenschaften der Reiter Symbolisierung und Beschriftung . . . 804
15.3.4 Darstellung . . . 806
15.3.5 Metadaten . . . 806
15.4 QGIS Server Properties . . . 806

16 Arbeiten mit Punktwolken 807
16.1 Einführung in Punktwolken . . . 807
16.2 Punktwolkenlayer-Eigenschaften . . . 808
16.2.1 Information . . . 808
16.2.2 Quelle . . . 809
16.2.3 Symbolisierung . . . 812
16.2.4 3D-Ansicht Eigenschaften . . . 820
16.2.5 Darstellung . . . 822
16.2.6 Höhe . . . 822
16.2.7 Metadaten . . . 824
16.2.8 Statistik . . . 824
16.3 Virtuelle Punktwolke . . . 825

17 Arbeiten mit 3D Tiles 829
17.1 Was sind 3D Tiles? . . . 829
17.1.1 State of 3D Tiles Support . . . 830
17.2 3D Tiles Eigenschaften . . . 831
17.2.1 Information . . . 831
17.2.2 Quelle . . . 831
17.2.3 Symbolisierung . . . 832
17.2.4 3D-Ansicht Eigenschaften . . . 833
17.2.5 Darstellung . . . 833
17.2.6 Höhe . . . 834
17.2.7 Metadaten . . . 834

18 Kartenlayout 835
18.1 Überblick über das Druck-Layout . . . 835
18.1.1 Beispielsitzung für Einsteiger . . . 835
18.1.2 Layoutverwaltung . . . 836
18.1.3 Menüs, Werkzeuge und Bedienfelder des Drucklayouts . . . 838
18.2 Layout-Elemente . . . 852
18.2.1 Gemeinsame Optionen der Layout-Elemente . . . 852
18.2.2 Das Karten-Element . . . 858
18.2.3 Das 3D Kartenelement . . . 869
18.2.4 Das Beschriftungs-Element . . . 870
18.2.5 Das Legenden-Element . . . 875
18.2.6 Das Maßstabs-Element . . . 884
18.2.7 Das Tabellen-Element . . . 888
18.2.8 Die Elemente Markierung, Bild und Nordpfeil . . . 897
18.2.9 Das Geländehöhenprofil-Element . . . 902
18.2.10 Das HTML-Rahmen Element . . . 903
18.2.11 Das Form-Element . . . 906
18.3 Eine Ausgabe erzeugen . . . 910
18.3.1 Exporteinstellungen . . . 910
18.3.2 Exportieren als Rasterbild . . . 911
18.3.3 Als SVG exportieren . . . 912

18.3.4 Als PDF exportieren . . . 914
18.3.5 Einen Atlas erzeugen . . . 915
18.4 Einen Bericht erstellen . . . 921
18.4.1 Was ist ein Bericht? . . . 921
18.4.2 Erste Schritte . . . 922
18.4.3 Arbeitsbereich für Layout-Berichte . . . 923
18.4.4 Exporteinstellungen . . . 938

19 Arbeiten mit OGC / ISO-Protokollen 939
19.1 WMS/WMTS Client . . . 940
19.1.1 Übersicht über die WMS-Unterstützung . . . 940
19.1.2 Überblick über die WMTS Unterstützung . . . 940
19.1.3 WMS/WMTS Server auswählen . . . 941
19.1.4 WMS/WMTS Layer laden . . . 943
19.1.5 Tilesets . . . 945
19.1.6 Das Objekte abfragen Werkzeug . . . 946
19.1.7 Anzeigen von Eigenschaften . . . 946
19.1.8 WMS-Legendengrafik im Inhaltsverzeichnis und Layout anzeigen . . . 950
19.2 WCS Client . . . 951
19.3 WFS und WFS-T Klient . . . 952
19.4 SensorThings . . . 955
19.4.1 Verbindung herstellen . . . 955
19.4.2 Laden von SensorThings-Daten . . . 957
19.4.3 Arbeiten mit einem Vektor-Layer aus SensorThings . . . 958
19.5 STAC (SpatioTemporal Asset Catalogs) . . . 959
19.5.1 Verbindung herstellen . . . 959
19.5.2 Browsing STAC Catalogs . . . 960
19.5.3 Filtering and Searching STAC Items . . . 961
19.6 ArcGIS REST Server . . . 963

20 Arbeiten mit GPS Daten 965
20.1 Einführung in GNSS/GPS-Daten . . . 965
20.1.1 Was ist GPS? . . . 965
20.1.2 Übertragen oder Laden von GPS-Daten . . . 966
20.2 Live-GPS-Verfolgung . . . 968
20.2.1 GPS Werkzeugleiste . . . 968
20.2.2 Das Bedienfeld GPS-Information . . . 970
20.2.3 Ein Bluetooth GPS fürs Live Tracking anbinden . . . 972
20.2.4 Beispiele für die Anbindung von GPS-Geräten in QGIS . . . 972

21 Authentifizierungssystem 975
21.1 Authentifizierungssystem Übersicht . . . 975
21.1.1 Authentifikationsdatenbank . . . 976
21.1.2 Custom authentication databases . . . 976
21.1.3 Hauptpasswort . . . 977
21.1.4 Authentifikationskonfiguration . . . 978
21.1.5 Authentifizierungsmethoden . . . 981
21.1.6 Hauptpasswort und Auth Konfig Hilfsmittel . . . 988
21.1.7 Gebrauche Authentifikationskonfiguration . . . 989
21.1.8 Python-Bindung . . . 989
21.2 Benutzerauthentifizierung Workflows . . . 990
21.2.1 HTTP(S) Authentifikation . . . 990
21.2.2 Datenbank Authentifikation . . . 991
21.2.3 PKI Authentifikation . . . 992
21.2.4 Defekte Layer behandeln . . . 999
21.2.5 Kennung der Authentifikationskonfiguration ändern . . . 1000
21.2.6 QGIS-Server-Unterstützung: . . . 1000
21.2.7 SSL-Serverausnahmen . . . 1001

22 GRASS GIS Integration **1005**
22.1 Beispieldatensatz . . . 1005
22.2 GRASS Layer visualisieren . . . 1005
22.3 Daten in eine GRASS LOCATION importieren via drag and drop . . . 1006
22.4 GRASS Daten im QGIS Browser verwalten . . . 1006
22.5 GRASS Optionen . . . 1006
22.6 GRASS Plugin starten . . . 1007
22.7 GRASS Mapset öffnen . . . 1007
22.8 Information zur GRASS-Datenbank . . . 1007
22.9 Daten in eine GRASS LOCATION importieren . . . 1008
22.9.1 Eine neue GRASS LOCATION erstellen . . . 1009
22.9.2 Eine neue GRASS MAPSET erstellen . . . 1010
22.10 Das GRASS Vektormodell . . . 1011
22.11 Einen neuen GRASS Vektorlayer erstellen . . . 1011
22.12 Digitalisieren und Editieren eines GRASS Vektorlayers . . . 1012
22.13 Einstellung der GRASS Region . . . 1014
22.14 Die GRASS Werkzeugkiste . . . 1015
22.14.1 Arbeiten mit GRASS Modulen . . . 1015
22.14.2 GRASS Beispielanwendung . . . 1018
22.14.3 Anpassen der Module . . . 1020

23 Die QGIS Verarbeitungsumgebung **1021**
23.1 Einführung . . . 1021
23.2 Konfigurieren der Verarbeitungsumgebung . . . 1024
23.2.1 Allgemein . . . 1024
23.2.2 Menüs . . . 1026
23.2.3 Modelle und Skripte . . . 1026
23.2.4 Anbieter . . . 1026
23.3 Die Werkzeugkiste . . . 1027
23.3.1 Der Algorithmus Dialog . . . 1029
23.3.2 Von Algorithmen erstellte Datenobjekte . . . 1037
23.4 Das Protokoll . . . 1039
23.4.1 Das Verarbeitung Protokoll . . . 1039
23.4.2 Das Verarbeitung Log . . . 1040
23.5 Ein Modell erstellen . . . 1040
23.5.1 Die Benutzeroberfläche des Modellentwurfs . . . 1041
23.5.2 Ein Modell erstellen . . . 1044
23.5.3 Speichern und laden von Modellen . . . 1053
23.5.4 Ein Modell bearbeiten . . . 1054
23.6 Die Schnittstelle zur Stapelverarbeitung . . . 1055
23.6.1 Einführung . . . 1055
23.6.2 Die Parameter-Tabelle . . . 1056
23.6.3 Füllen der Parameter-Tabelle . . . 1057
23.6.4 Ausführen eines Batch-Prozesses . . . 1058
23.7 Verarbeitungs-Algorithmen von der Konsole aus verwenden . . . 1059
23.7.1 Algorithmen von der Python-Konsole aus aufrufen . . . 1059
23.7.2 Scripte erstellen und diese aus der Werkzeugkiste starten. . . . 1064
23.7.3 Skript-Hooks vor und nach der Ausführung . . . 1067
23.8 Verarbeitung über die Befehlszeile . . . 1067
23.9 Schreiben neuer Verarbeitungs-Algorithmen als Python-Skripte . . . 1069
23.9.1 Erweitern von QgsProcessingAlgorithm . . . 1069
23.9.2 Der @alg decorator . . . 1073
23.9.3 Handhabung der Algorithmus-Ausgabe . . . 1075
23.9.4 Kommunikation mit dem Nutzer. . . . 1075
23.9.5 Ihre Scripte dokumentieren . . . 1076
23.9.6 Flags . . . 1076
23.9.7 Bewährte Verfahren für das Schreiben von Skript-Algorithmen . . . 1076
23.10 Eingabe- und Ausgabetypen für Verarbeitungsalgorithmen . . . 1076

23.10.1 Eingabe-Typen . . . 1076
23.10.2 Ausgabetypen . . . 1079
23.11 Konfiguration externer Anwendungen . . . 1079
23.11.1 Ein Hinweis für Windows Anwender . . . 1080
23.11.2 Eine Bemerkung zu den Datenformaten . . . 1080
23.11.3 Eine Bemerkung zu Vektorlayern mit ausgewählten Objekten . . . 1080
23.11.4 Datenanbieter von Dritten benutzen . . . 1080

24 Verarbeitungsalgorithmen **1091**
24.1 QGIS Verarbeitungsalgorithmen . . . 1091
24.1.1 3D Tiles . . . 1091
24.1.2 Kartografie . . . 1093
24.1.3 Check Geometry . . . 1113
24.1.4 Datenbank . . . 1159
24.1.5 File tools . . . 1166
24.1.6 Fix Geometry . . . 1169
24.1.7 GPS . . . 1195
24.1.8 Interpolation . . . 1198
24.1.9 Layer tools . . . 1211
24.1.10 Netz . . . 1215
24.1.11 Metadata tools . . . 1228
24.1.12 Modellierungswerkzeuge . . . 1233
24.1.13 Network analysis . . . 1243
24.1.14 Plots . . . 1257
24.1.15 Punktwolkenkonvertierung . . . 1271
24.1.16 Punktwolkendatenverwaltung . . . 1278
24.1.17 Point Cloud Extraction . . . 1292
24.1.18 Raster analysis . . . 1298
24.1.19 Raster Creation . . . 1363
24.1.20 Raster terrain analysis . . . 1381
24.1.21 Raster tools . . . 1394
24.1.22 Vektoranalyse . . . 1406
24.1.23 Vector coverage . . . 1433
24.1.24 Vector creation . . . 1436
24.1.25 Vector general . . . 1466
24.1.26 Vektor-Geometrie . . . 1506
24.1.27 Vector overlay . . . 1629
24.1.28 Vector selection . . . 1649
24.1.29 Vector table . . . 1667
24.1.30 Vector Tiles . . . 1684
24.2 GDAL Verarbeitungsalgorithmen . . . 1688
24.2.1 Raster analysis . . . 1688
24.2.2 Raster conversion . . . 1719
24.2.3 Raster extraction . . . 1728
24.2.4 Raster miscellaneous . . . 1736
24.2.5 Rasterprojektionen . . . 1756
24.2.6 Vector conversion . . . 1761
24.2.7 Vector geoprocessing . . . 1767
24.2.8 Vector miscellaneous . . . 1776

25 Erweiterungen **1789**
25.1 QGIS Plugins . . . 1789
25.1.1 Kern-Erweiterungen und externe Erweiterungen . . . 1789
25.1.2 Der Erweiterungen Dialog . . . 1790
25.2 QGIS Kern-Erweiterungen verwenden . . . 1794
25.2.1 DB Verwaltung . . . 1794
25.2.2 Die Erweiterung Geometrien prüfen . . . 1798
25.2.3 MetaSearch Catalog Client . . . 1802

25.2.4 Offline-Bearbeitung Erweiterung . . . 1809
25.2.5 Die Topologie-prüfen-Erweiterung . . . 1811
25.3 QGIS Python Konsole . . . 1814
25.3.1 Die interaktive Konsole . . . 1814
25.3.2 Der Code-Editor . . . 1816

26 Anhänge **1819**
26.1 Anhang C: QGIS-Dateiformate . . . 1819
26.1.1 QGS/QGZ - Das QGIS-Projekt-Dateiformat . . . 1819
26.1.2 QLR – die QGIS-Layer-Definitionsdatei . . . 1821
26.1.3 QML - Das QGIS-Stil-Dateiformat . . . 1822
26.2 Anhang D: QGIS R Skriptsyntax . . . 1823
26.2.1 Eingaben . . . 1824
26.2.2 Ausgaben . . . 1824
26.2.3 Syntaxzusammenfassung für QGIS R-Skripte . . . 1824
26.2.4 Beispiele . . . 1826
26.3 Anhang E: Netzwerkverbindungen der QGIS-Anwendung . . . 1828

27 Literatur und Internetreferenzen **1831**

KAPITEL 1

Erste Schritte

Dieses Kapitel gibt einen kurzen Überblick über die Installation von QGIS, den Download von QGIS-Beispieldaten und die Erstellung eines ersten kleinen Projekts zur Visualisierung von Raster- und Vektordaten.

1.1 QGIS installieren

QGIS steht als Installation für verschiedene Plattformen und mit verschiedenen Optionen zur Verfügung.

1.1.1 Installation mit Binärdateien

Für MS Windows und macOS sind Standardinstallationen verfügbar. Binärpakete (rpm und deb) oder Software-Repositorys werden für viele Varianten von GNU/Linux bereitgestellt.

Weitere Informationen und Anleitungen für Ihr Betriebssystem finden Sie unter https://qgis.org/download/.

1.1.2 Kompilieren des Quellcodes

Wenn Sie QGIS mit Hilfe des Quellcodes erstellen wollen, lesen Sie bitte die Installationsanweisungen. Sie erhalten diese zusammen mit dem QGIS-Quellcode in einer Datei namens `INSTALL`. Sie können die Datei auch online unter https://github.com/qgis/QGIS/blob/release-3_44/INSTALL.md finden.

Wenn Sie eine bestimmte Version und nicht die in der Entwicklung befindliche Version erstellen möchten, sollten Sie `Master` durch den Release-Zweig (üblicherweise im Formular `release-X_Y`) im oben genannten Link ersetzen (Installationsanweisungen können abweichen).

1.1.3 Auf externen Medien installieren

Es ist möglich, QGIS (mit allen Plugins und Einstellungen) auf einem Flash-Laufwerk zu installieren. Dies wird durch die Definition der *–profiles-path* -Option erreicht, die den standardmäßigen Pfad *user profile* überschreibt und **QSettings** zwingt, auch dieses Verzeichnis zu verwenden. Siehe Abschnitt *System* für weitere Informationen.

1.1.4 Beispieldaten herunterladen

Dieses Benutzerhandbuch enthält Beispiele, in denen Daten aus dem QGIS-Beispieldatensatz verwendet werden (manchmal wird dieser auch `Alaska-Datensatz` genannt). Diese Daten können Sie von https://github.com/qgis/QGIS-Sample-Data/archive/master.zip herunterladen.

Der QGIS-Beispieldatensatz enthält Geodaten von Alaska und deckt sämtliche Übungen und Screenshots dieser Dokumentation ab, inklusive einer GRASS GIS Datenbank. Das Koordinatenbezugssystem ist Albers Equal Area (EPSG-Code 2964) mit der Maßeinheit 'feet' .

```
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]]
```

Wenn Sie beabsichtigen, QGIS als grafisches Frontend für GRASS zu verwenden, finden Sie eine Auswahl von Beispieldaten (z.B. Spearfish oder South Dakota) auf der offiziellen GRASS GIS Website https://grass.osgeo.org/download/data/.

1.2 QGIS starten und beenden

QGIS kann wie jede andere Anwendung auf Ihrem Computer gestartet werden. Das bedeutet, dass Sie QGIS wie folgt starten können:

- über das Anwendungsmenü, das Startmenü oder das Dock
- Doppelklicken Sie das QGIS Icon oder die Desktop Verknüpfung
- Doppelklicken Sie auf eine vorhandene QGIS-Projektdatei (mit der Erweiterung `.qgz` oder `.qgs`). Beachten Sie, dass dadurch auch das Projekt geöffnet wird.
- `qgis` in die Kommandozeile eintippen, vorausgesetzt, QGIS wurde ihrem PFAD hinzugefügt oder Sie befinden sich im Installationsordner

Um QGIS zu stoppen, benutzen Sie:

- die Menüoptionen *Projekt* ► *QGIS beenden* oder benutzen Sie das Tastenkürzel `Strg+Q`

- *QGIS* ► *QGIS beenden*, oder benutzen Sie das Tastenkürzel `Ctrl+Q`
- oder benutzen Sie das rote Kreuz in der rechten, oberen Ecke des Hauptfensters der Anwendung.

1.3 Beispielsitzung: Laden von Raster- und Vektorlayern

Nachdem Sie nun *QGIS installiert* und einen *Beispiel-Datensatz* zur Verfügung haben, werden wir eine erste Beispielsitzung demonstrieren. In diesem Beispiel werden wir einen Raster- und einen Vektorlayer laden. Wir verwenden:

- den `landcover` Rasterlayer (`qgis_sample_data/raster/landcover.img`)
- und den `lakes` Vektorlayer (`qgis_sample_data/gml/lakes.gml`)

Wobei `qgis_sample_data` der Pfad zum entpackten Datensatz ist.

1. Starten Sie QGIS wie in *QGIS starten und beenden* gezeigt
2. Die Daten, mit denen wir arbeiten werden, liegen im `Albers Equal Area` Koordiantensystem vor, also stellen wir das KBS des Projekts entsprechend ein:
    1. Klicken Sie auf die Schaltfläche Projektion wählen unten rechts in der QGIS-Oberfläche. Der Projekteigenschaften-Dialog öffnet sich mit aktivem *KBS* Reiter.
    2. Schreiben Sie``2964` in das *Filter* Textfeld.
    3. Wählen Sie `NAD27 / Alaska Albers` aus.

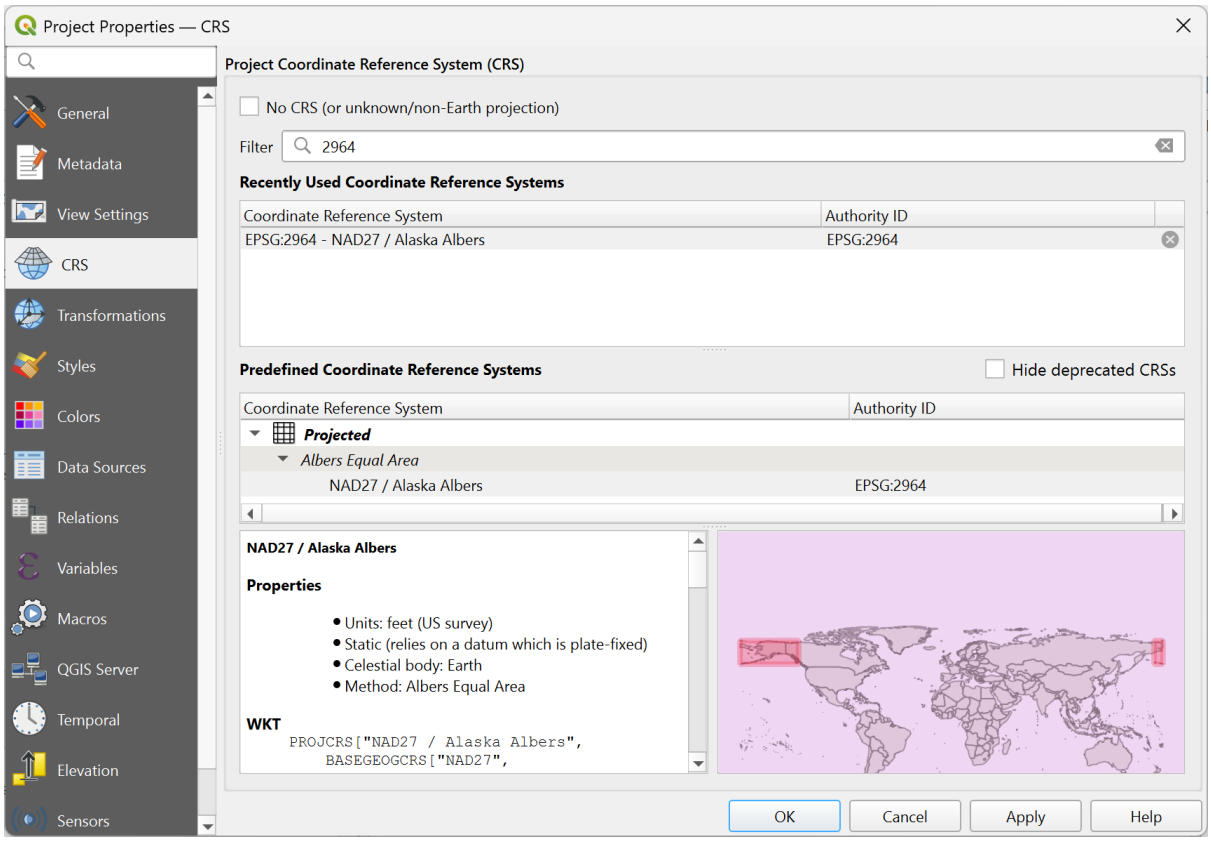


Abb. 1.1: Auswahl des Koordinatenbezugssystems für die Datei

4. Klicken Sie auf *OK*

> **Bemerkung**
>
> Sollten Meldungen zu möglichen Transformationsfehlern erscheinen, können Sie diese vorerst ignorieren.

3. Laden Sie jetzt Dateien in QGIS dazu:

   1. Klicken Sie auf die Schaltfläche Datenquellenverwaltung öffnen. Die Datenquellenverwaltung sollte sich im Browsermodus öffnen.
   2. Navigieren Sie zum Verzeichnis `qgis_sample_data/raster/`
   3. Wählen Sie die ERDAS IMG-Datei *landcover.img* mit einem Doppelklick aus. Der Layer wird im Hintergrund geladen, während das Fenster der Datenquellenverwaltung geöffnet bleibt.

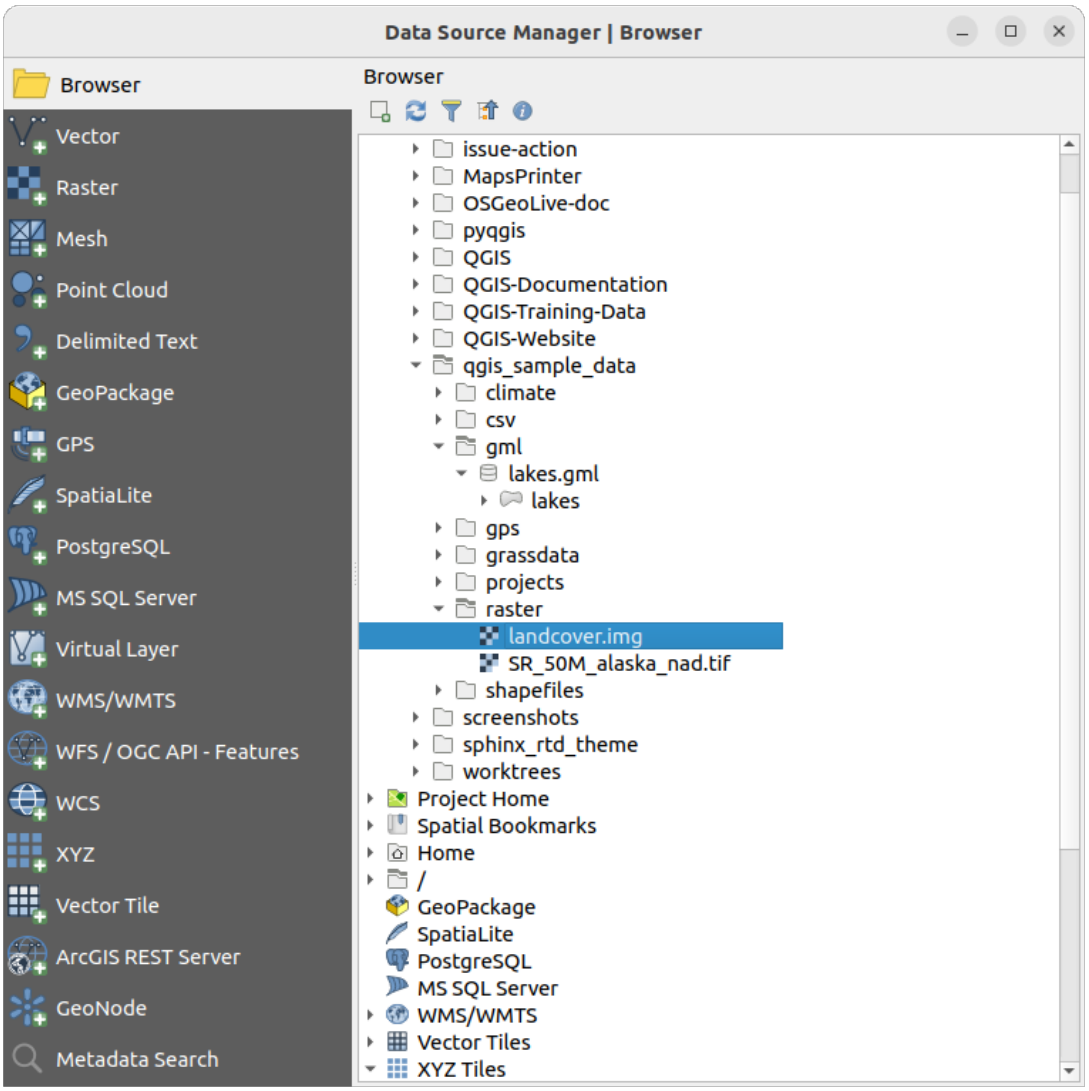


Abb. 1.2: Daten zu einem neuen Projekt in QGIS hinzufügen

4. Um die Seen-Daten zu laden, wechseln Sie in den Ordner `qgis_sample_data/gml/` und doppelklicken auf die Datei *lakes.gml*. Sie können die Daten auch dazu laden, indem Sie sie vom Browser in das QGIS Hauptfenster ziehen.
5. Da das `.gml`-Dateiformat mehr als einen Layer enthalten kann öffnet sich das Dialogfeld *Elemente zum*

Hinzufügen auswählen.

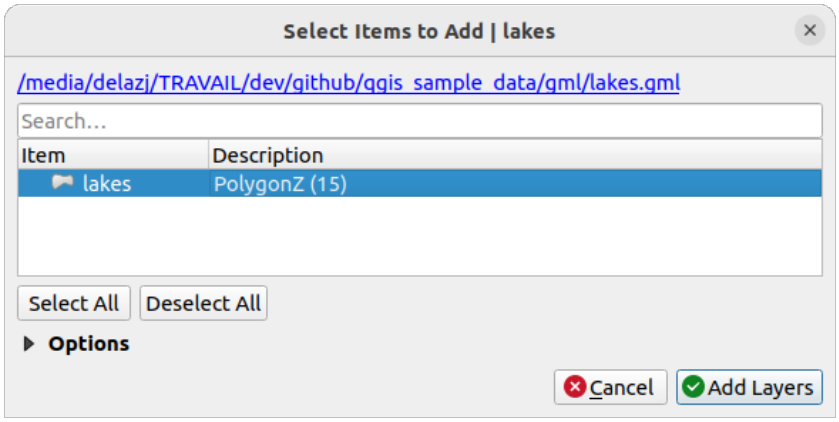


Abb. 1.3: Layer aus einer Datei auswählen

6. In unserem Fall gibt es nur einen Layer: *lakes*. Wählen Sie diesen aus und klicken Sie auf *Layer hinzufügen*.
7. Der Layer wird im *Layer*-Bedienfeld hinzugefügt.

4. Schließen Sie das Fenster „Datenquellenverwaltung“

Im *Layer*-Bedienfeld können Sie sehen, dass neben dem Layer *lakes* Für den Layer ist kein Koordinatenbezugssystem gesetzt anzeigt wird. Lassen Sie uns das ändern.

1. Klicken Sie auf das Symbol . Das Dialogfenster *Koordinatenbezugssystem Auswahl* wird geöffnet.
2. Wie zuvor suchen Sie den KBS-Eintrag *NAD27 / Alaska Albers* und wählen ihn aus.
3. Klicken Sie auf *OK*

Die beiden Layer in Ihrem Projekt werden jetzt noch in zufälligen Farben dargestellt. Lassen Sie uns nun die Darstellung des Layers `lakes` etwas anpassen.

1. Wählen Sie das Werkzeug Hineinzoomen in der Werkzeugleiste *Kartennavigation* aus.
2. Zoomen Sie in ein Gebiet, in dem sich einige Seen befinden.
3. Doppelklicken Sie auf den Layer `lakes` im *Layer*-Bedienfeld. Das *Layereigenschaften* Fenster öffnet sich.
4. So können Sie die Farbe der Seen ändern:
    1. Klicken Sie auf den Reiter *Symbolisierung*.
    2. Wählen Sie blau als Füllfarbe aus.

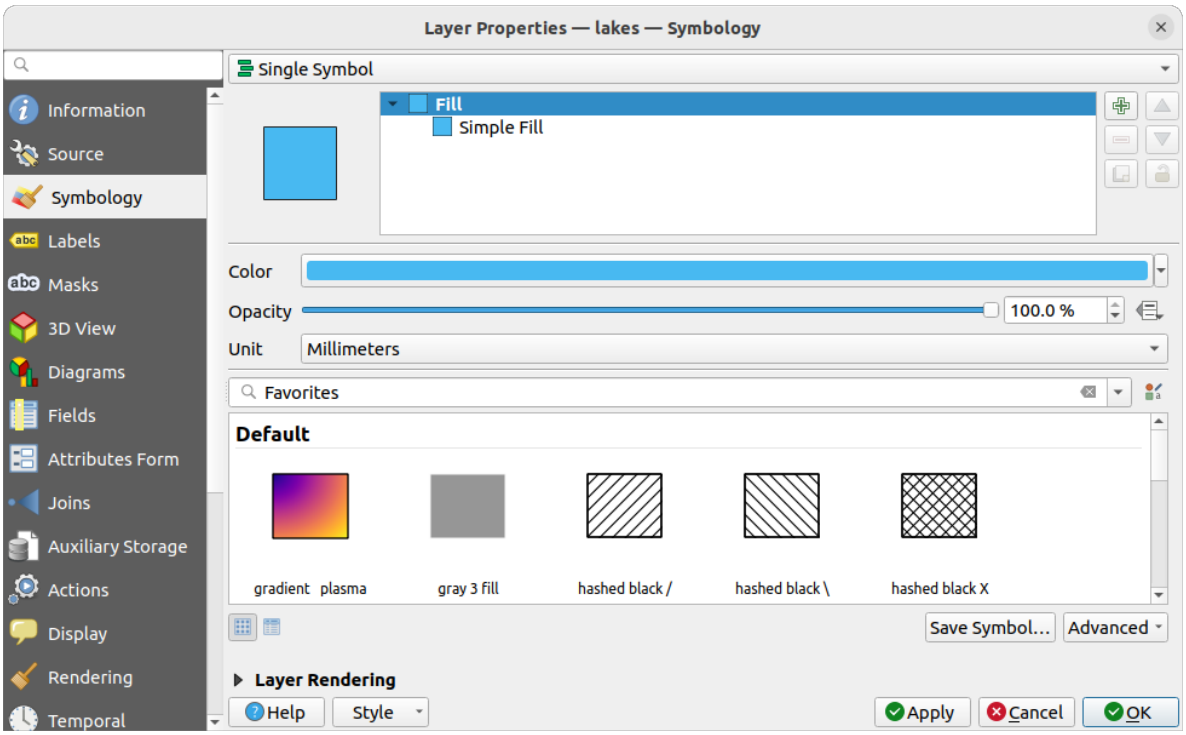


Abb. 1.4: Farbe für die Seen auswählen

3. Drücken Sie *OK*. Die Seen werden in der Kartenansicht jetzt in blau dargestellt.

5. So können Sie die Namen der Seen darstellen lassen:
   1. Öffnen Sie, wie oben beschrieben, das *Layereigenschaften* Fenster des Layers `lakes`.
   2. Klicken Sie auf den Reiter *Beschriftungen*
   3. Wählen Sie *Einzelne Beschriftungen* aus dem Aufklappmenü aus, um die Beschriftung anzuschalten.
   4. Wählen Sie aus der Liste *Beschriften mit* das Feld `NAMES`.

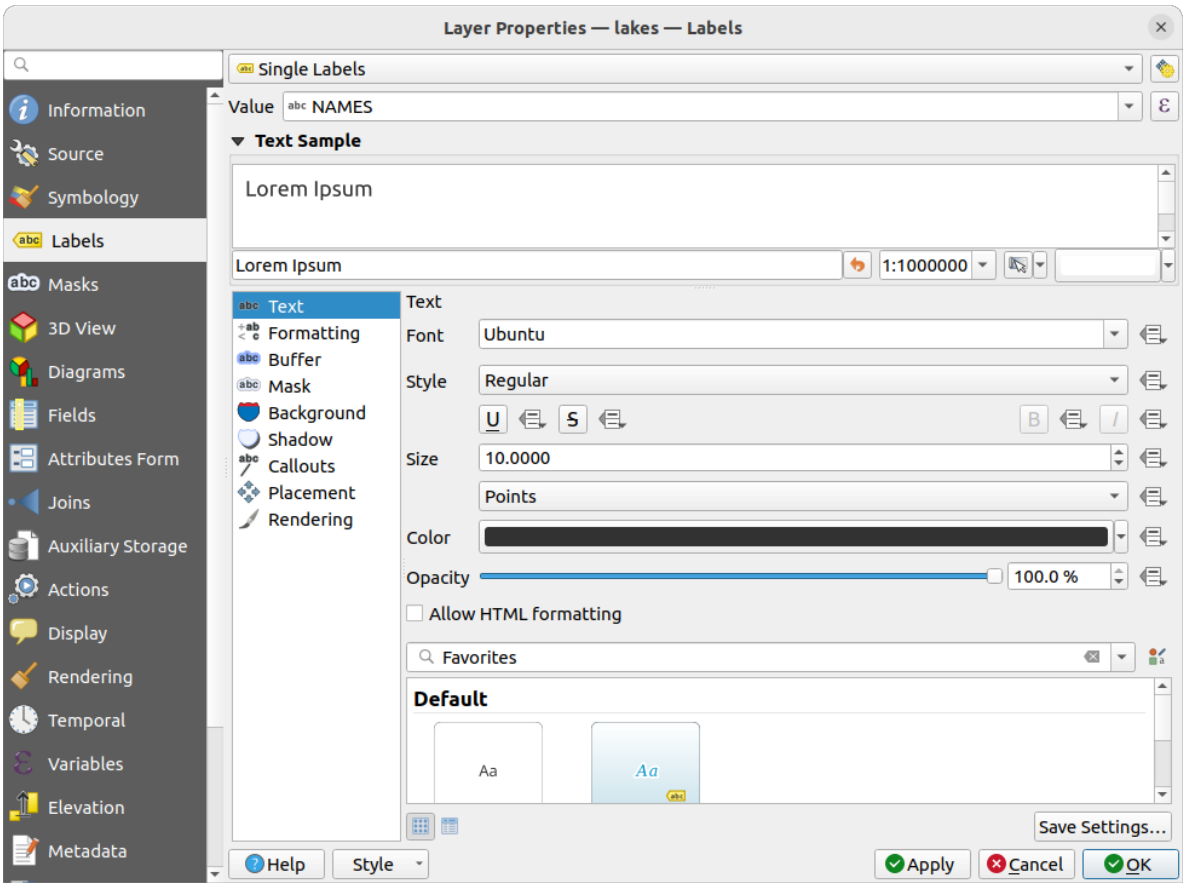


Abb. 1.5: Die Namen der Seen anzeigen

5. Drücken Sie auf *Anwenden*. Die Namen werden nun über den Grenzen dargestellt.

6. Sie können die Lesbarkeit der Beschriftungen verbessern, indem Sie einen weißen Puffer drumherum legen:

   1. Klicken Sie auf den Reiter *Puffer* in der Liste links
   2. Haken Sie *Textpuffer zeichnen* an
   3. Wählen Sie``*3*` als Größe aus
   4. Klicken Sie auf *Anwenden*
   5. Prüfen Sie nach, ob das Ergebnis gut aussieht und verändern Sie den Wert, falls nötig.
   6. Klicken Sie abschließend auf *OK*, um das *Layereigenschaften* Fenster zu schließen und die Änderungen zu übernehmen.

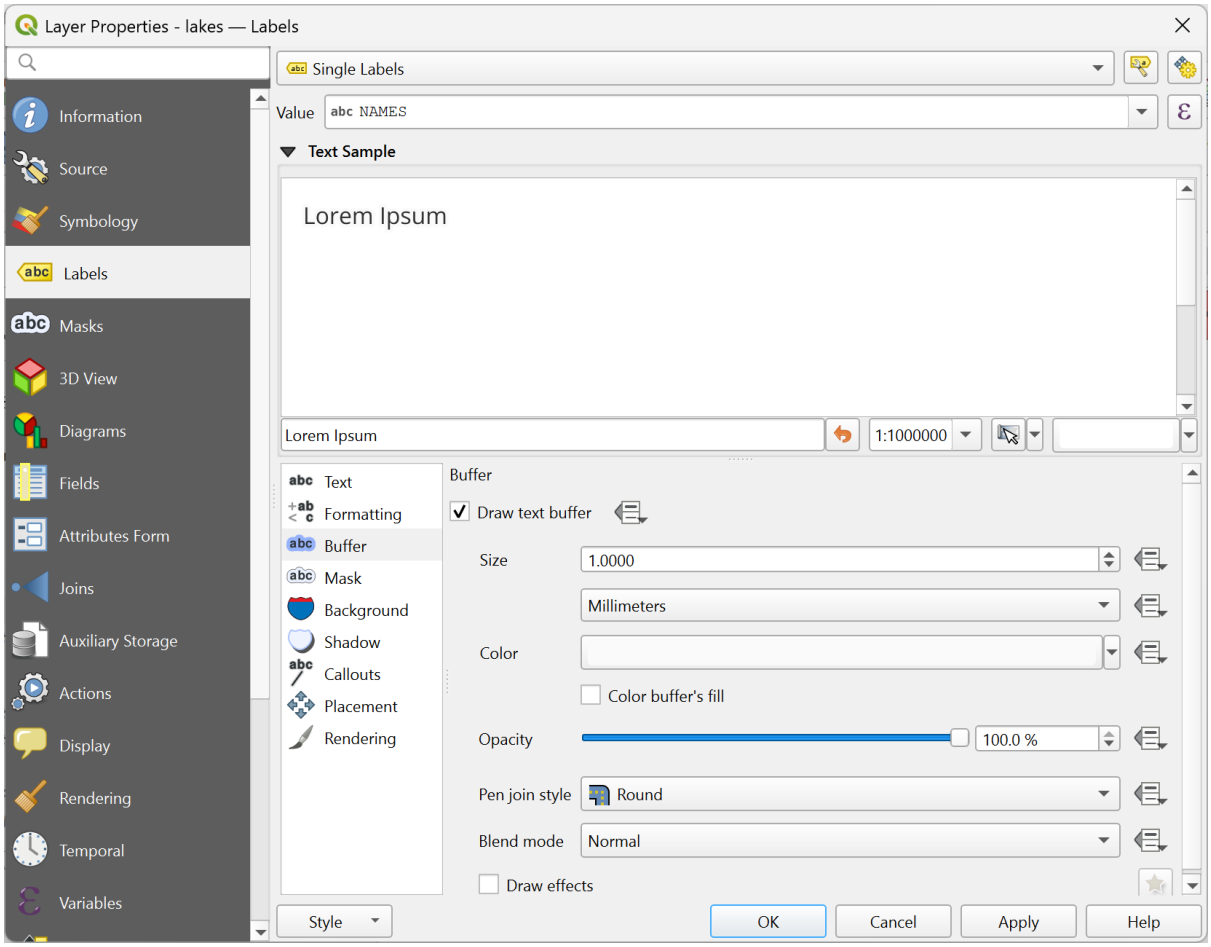


Abb. 1.6: Puffer um die Beschriftungen zeichnen

Sie können nun einige Dekorationen hinzufügen, um die Karte zu optimieren, und sie dann aus QGIS zu exportieren:

1. Wählen Sie im Menü *Ansicht* ► *Dekorationen* ► *Maßstab* aus.
2. Im Dialog, der sich öffnet, haken Sie *Aktiviere Maßstab* an.
3. Passen Sie die Einstellungen des Dialogs nach Ihren Wünschen an.
4. Drücken Sie auf *Anwenden*.
5. Auf die gleiche Art können Sie aus dem Menü Dekorationen weitere Objekte (Nordpfeil, Urheberrechtshinweis…) mit den gewünschten Eigenschaften zur Kartenansicht hinzufügen.
6. Klicken Sie auf den Menüpunkt *Projekt* ► *Import/Export* ► *Karte als Bild speichern…*.
7. Drücken Sie in dem Fenster, das sich geöffnet, hat auf *Speichern*.
8. Wählen Sie einen Speicherpfad und ein Format aus und klicken dann nochmals auf *Speichern*.
9. Wählen Sie *Projekt* ► *Speichern unter…*, um Ihr Projekt als `.qgz`-Datei zu speichern.

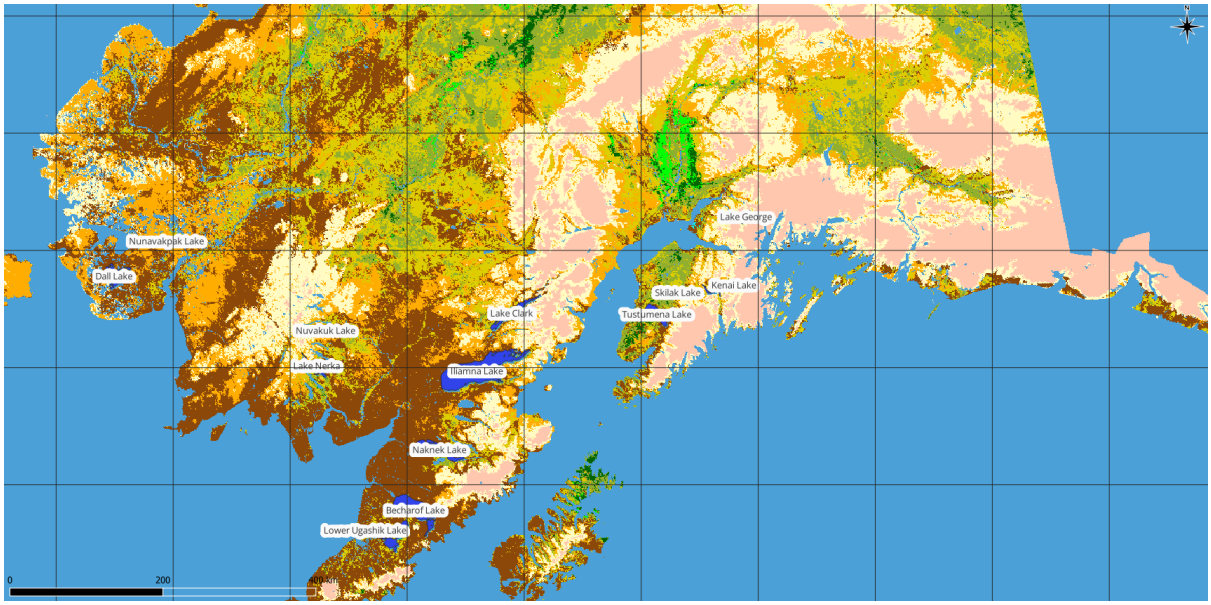


Abb. 1.7: So sieht die exportierte Karte mit Dekorationen aus

Das war's. Sie sehen, wie einfach es ist, in QGIS Raster- und Vektorlayer darzustellen, ihre Eigenschaften zu ändern und Karten in einem Bildformat zu speichern (die Sie in anderer Software benutzen können). Sie können jetzt im Handbuch weitergehen und mehr über die verfügbaren Funktionen und Einstellungen lernen.

Bemerkung

Lesen Sie das QGIS Übungshandbuch, um mit Übungen QGIS schrittweise weiter kennenzulernen.

KAPITEL 2

Arbeiten mit Projektdateien

2.1 Vorstellung von QGIS-Projekten

Eine Kartenzusammenstellung mit allen Einstellungen und Werkzeugen wird in QGIS „Projekt“ genannt. In einem QGIS-Fenster kann immer nur ein Projekt gleichzeitig bearbeitet werden, es können aber mehrere Fenster geöffnet sein. Einstellungen können projektspezifisch oder anwendungsweit für alle neuen Projekte sein (siehe Abschnitt *Optionen*). QGIS kann den aktuellen Zustand in eine *QGIS-Projektdatei* speichern; benutzen Sie dafür das Menü *Projekt* ► *Speichern* oder *Projekt* ► *Speichern als…*.

Bemerkung

Sobald das Projekt verändert wurde, erscheint das Zeichen * in der Titelleiste und QGIS wird Sie standardmäßig fragen, ob Sie die Änderungen speichern wollen. Dieses Verhalten steuern Sie über die Einstellung *Bei Bedarf nachfragen, ob geänderte Projekte und Datenquellen gespeichert werden sollen*, die Sie unter *Einstellungen* ► *Optionen* ► *Allgemein* finden.

Sie können bestehende Projekte über *Projekt* ► *Öffnen…*, *Projekt* ► *Neu aus Vorlage* oder *Projekt* ► *Zuletzt geöffnet* ► laden.

Im *Browser-Bedienfeld* haben Sie über das Kontextmenü einer Projektdatei verschiedene Optionen: Sie können das *Projekt öffnen*, die *Symbole extrahieren…* (öffnet den Stil-Manager, um Symbole zu exportieren (XML/PNG/SVG) oder sie zu den Vorgabestilen hinzuzufügen). Sie können auch auf das Dialogfeld *Dateieigenschaften…* zugreifen. Über das Dreieckssymbol vor dem Namen der Projektdatei kann diese ausgeklappt werden, um die enthaltenen Layer anzuzeigen (das Kontextmenü eines Layers bietet dann dieselben Möglichkeiten wie für alle Layer, wen sie im Browser angezeigt werden).

Beim Start wird eine Liste mit *Projektvorlagen* und den zuletzt verwendeten Projekten *Aktuelle Projekte* angezeigt, einschließlich Screenshots, Namen und Dateipfaden (für bis zu zehn Projekte). Doppelklicken Sie auf einen Eintrag, um das Projekt oder die Projektvorlage zu öffnen. Mit einem Rechtsklick auf einen Eintrag haben Sie die Optionen: *An Liste hängen* (dieser Eintrag wird dann dauerhaft angezeigt), *Verzeichnis öffnen…* oder *Aus Liste entfernen*. Wenn Sie Ihr Projekt in der Liste *Aktuelle Projekte* sehen, es aber in Ihrem Dateimanager nicht finden können, verwenden Sie die Option *Verzeichnis öffnen…*, um Projekte zu finden, die möglicherweise fehlen, verschoben oder umbenannt wurden. Sie können auch auf *Liste löschen* klicken, wenn Sie alle Projekte aus der Liste *Aktuelle Projekte* entfernen möchten. Wenn Sie Projekte angeheftet haben, wird nach dem Klicken auf *Liste löschen* eine Meldung angezeigt, in der Sie gefragt werden, ob die angehefteten Projekte ebenfalls entfernt werden sollen. Sie können auch einen Layer

hinzufügen, um automatisch ein neues Projekt zu erstellen. Die Listen verschwinden dann automatisch und machen Platz für die Darstellung des hinzugefügten Layers.

Wenn Sie Ihre Sitzung löschen und neu beginnen möchten, gehen Sie zu *Projekt* ► *Neu*. Dabei werden Sie aufgefordert, das bestehende Projekt zu speichern, wenn seit dem Öffnen oder letzten Speichern Änderungen vorgenommen wurden.

Sobald Sie ein neues Projekt erzeugen, zeigt die Titelleiste bis zum Speichern den Titel `Unbenanntes Projekt` an.

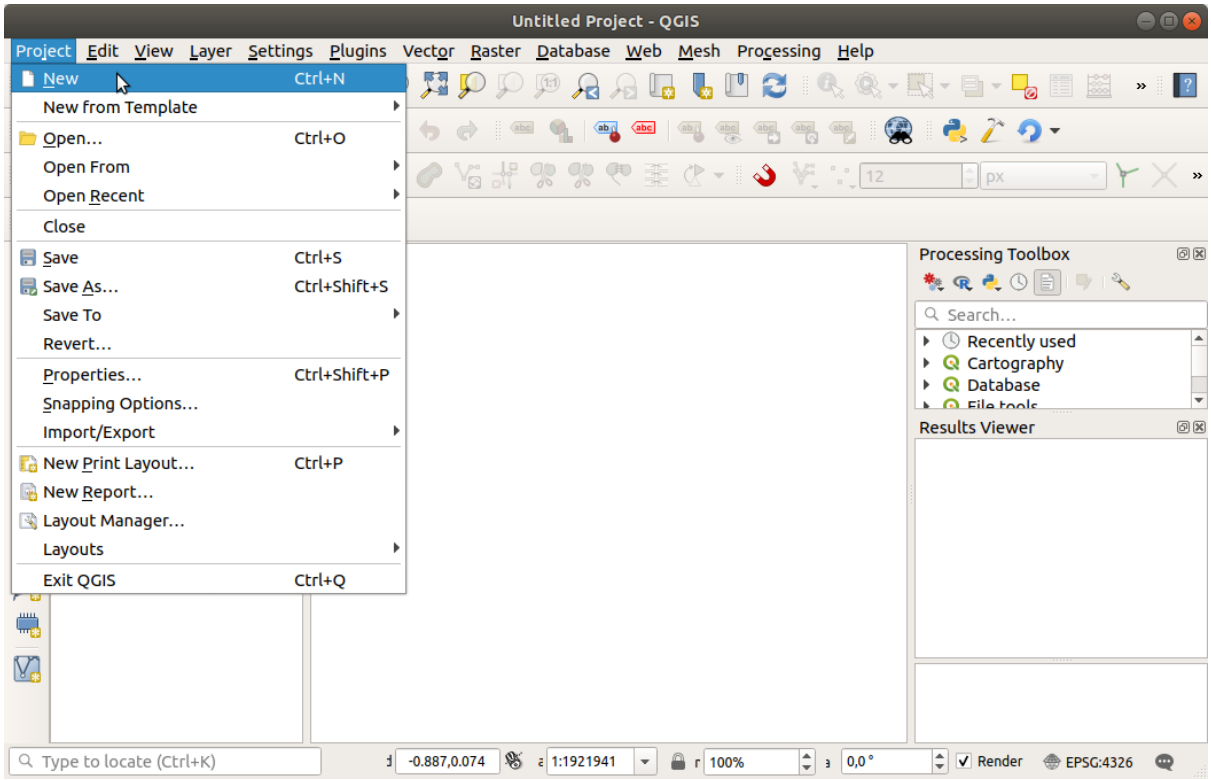


Abb. 2.1: Start eines neuen Projekts in QGIS

In einer Projektdatei sind folgende Informationen gespeichert:

- Hinzugefügte Layer
- Welche Layer können abgefragt werden
- Layereigenschaften, inklusive Symbologie und Styles
- Layernotizen
- 2D und 3D Kartenansichten
- Projektion für jede Kartenansicht
- Der zuletzt betrachtete Ausschnitt für jede Kartenansicht
- Drucklayouts
- Drucklayout Elemente mit Einstellungen
- Drucklayouts Atlas Einstellungen
- Digitalisieren
- Tabellen Beziehungen
- Projekt Makros

- Projekt Vorgabestile
- Plugin-Einstellungen
- QGIS-Servereinstellungen aus der Registerkarte QGIS-Server in den Projekteigenschaften.
- Im DB-Manager gespeicherte Abfragen

Die Projektdatei wird im XML-Format gespeichert (siehe *QGS/QGZ - Das QGIS-Projekt-Dateiformat*). Damit ist es möglich, die Datei außerhalb von QGIS zu bearbeiten, wobei Sie aber wissen sollten, was Sie tun. Das Dateiformat der Projektdatei wurde bereits mehrere Male erneuert. Deshalb kann es sein, dass Projektdateien älterer QGIS-Versionen nicht mehr richtig funktionieren.

Bemerkung

Standardmäßig warnt Sie QGIS, wenn es Versionskonflikte gibt. Dieses Verhalten steuern Sie im Reiter *Allgemein* im Menü *Einstellungen* ► *Optionen* (*Warnung ausgeben, wenn QGIS-Projekt einer früheren Version geöffnet wird*).

Immer wenn Sie eine `.qgs`-Projektdatei speichern, wird eine Sicherungsdatei im selben Verzeichnis angelegt, die die Dateierweiterung `.qgs~` hat.

Die Erweiterung für QGIS-Projekte ist `.qgs`, standardmäßig wird jedoch beim Speichern ein komprimiertes Format mit der Erweiterung `.qgz` verwendet. Die `.qgs`-Datei ist in diese `.qgz`-Datei (Zip-Archiv) eingebettet, zusammen mit der zugehörigen SQLite-Datenbank (`.qgd`) für den *Hilfsspeicher*. Sie können auf diese Dateien zugreifen, indem Sie die `.qgz`-Datei entpacken.

Bemerkung

Das gepackte Projekt-Format `.qgz` ist daher besonders nützlich, wenn der *Hilsspeicher* Daten enthält, da dieser direkt eingebunden ist.

Projekte können auch in/aus einer PostgreSQL-, GeoPackage- oder Oracle-Datenbank gespeichert/geladen werden, indem Sie die folgenden Optionen aus dem Projekt-Menü verwenden:

- *Projekt* ► *Öffnen aus*
- *Projekt* ► *Speichern als*

Beide Menüpunkte haben ein Untermenü mit einer Liste von zusätzlichen Projektspeicherimplementierungen (PostgreSQL, GeoPackage und Oracle). Wenn Sie darauf klicken, öffnet sich ein Dialog zur Auswahl einer GeoPackage-Verbindung und eines Projekts, einer PostgreSQL-Verbindung, eines Schemas und eines Projekts oder einer Oracle-Verbindung, eines Eigentümers und eines Projekts.

Projekte, die in GeoPackage, PostgreSQL oder Oracle gespeichert sind, können auch über das QGIS-Browser-Panel geladen werden, entweder durch Doppelklick oder durch Ziehen auf die Kartenfläche.

2.2 Behandlung von fehlerhaften Dateipfaden

Beim Öffnen eines Projekts kann es vorkommen, dass QGIS Datenquellen nicht laden kann, weil ein Dienst/eine Datenbank nicht verfügbar ist oder eine Datei umbenannt, verschoben oder gelöscht wurde. QGIS öffnet dann den Dialog *Nicht verfügbare Layer behandeln* und führt die nicht gefundenen Layer auf. Sie können:

- Auf das Feld *Datenquelle* doppelklicken, den Pfad des Layers anpassen und auf *Änderungen anwenden* klicken;
- Eine Zeile auswählen, auf *Durchsuchen* klicken, den korrekten Ort auswählen und danach auf *Änderungen anwenden* klicken;

- Auf *Auto-Finden* klicken, die Verzeichnisse durchsuchen und versuchen alle oder die ausgewählten fehlerhaften Dateipfade automatisch zu reparieren. Achtung: das Durchsuchen kann eine Weile dauern. Klicken Sie danach auf *Änderungen anwenden*.
- Die Nachricht ignorieren und Ihr Projekt mit dem oder den fehlerhaften Pfaden öffnen, indem Sie auf *Nicht verfügbare Layer behalten* klicken. Der Layer wird dann im Bedienfeld *Layer* angezeigt, aber er enthält keine Daten, bis Sie den Pfad durch einen Klick auf Nicht verfügbarer Layer! neben dem jeweiligen Layer im Bedienfeld *Layer* oder auf *Datenquelle reparieren…* im Kontextmenü des Layers reparieren.

  Sobald Sie mit dem Werkzeug *Datenquelle reparieren…* einen Dateipfad korrigiert haben, sucht QGIS durch alle anderen fehlerhaften Pfade und versucht selbständig die zu reparieren, die denselben Dateipfad haben.
- *Nicht verfügbare Layer entfernen* entfernt alle Layer mit fehlerhaftem Pfad aus dem Projekt.

Wenn Sie QGIS von der Kommandozeile aus mit der Option *--skipbadlayers* starten, können Sie den Dialog *Nicht verfügbare Layer behandeln* beim Start überspringen.

2.3 Ausgaben erzeugen

Es gibt mehrere Möglichkeiten, um Ausgaben aus Ihrer QGIS-Sitzung zu erzeugen. Das Speichern als Projektdatei wurde bereits in *Vorstellung von QGIS-Projekten* besprochen. Andere Möglichkeiten, Ausgabedateien zu erzeugen, sind:

- Erstellen von Bildern: *Projekt* ► *Import/Export* ► *Karte als Bild speichern…* gibt die Karte in einem Bildformat (PNG, JPG, TIFF, …) in beliebigem Maßstab, Auflösung und Größe aus. Die Einbindung von Georeferenzierungsinformationen in das exportierte Bild ist möglich, aktivieren Sie einfach *Georeferenzierungsinformationen anhängen (eingebettet oder als World-Datei)*. Siehe *Exportieren der Kartenansicht* für weitere Details.
- PDF-Dateien erzeugen: *Projekt* ► *Import/Export* ► *Karte als PDF exportieren…* gibt die Kartenansicht als PDF im gewünschten Maßstab, in der gewünschten Auflösung und mit einigen erweiterten Einstellungen (Vereinfachung, Georeferenzierung, …) aus. Für eine genauere Beschreibung siehe *Exportieren der Kartenansicht*.
- DXF-Dateien erzeugen: *Project* ► *Import/Export* ► *Projekt als DXF speichern…* startet einen Dialog, in dem Sie den ‚Symbolisierungsmodus', den ‚Symbolisierungsmaßstab' und die Vektorlayer, die Sie nach DXF exportieren möchten, festlegen. Mithilfe des ‚Symbolisierungsmodus' können Sie Symbole des QGIS-Symbolsatzes mit hoher Ähnlichkeit exportieren (siehe Abschnitt *Erstellen neuer DXF-Dateien*).
- Karten erstellen: *Projekt* ► *Neues Drucklayout…* öffnet einen Dialog, in dem Sie die aktuelle Kartenansicht layouten und drucken können (siehe Abschnitt *Kartenlayout*).

KAPITEL 3

Die QGIS Oberfläche

In der unten stehenden Abbildung ist die grafische Benutzeroberfläche (GUI) von QGIS dargestellt. Die Zahlen 1 bis 5 in den gelben Kreisen kennzeichnen wichtige Elemente der Oberfläche und werden nachfolgend erläutert.

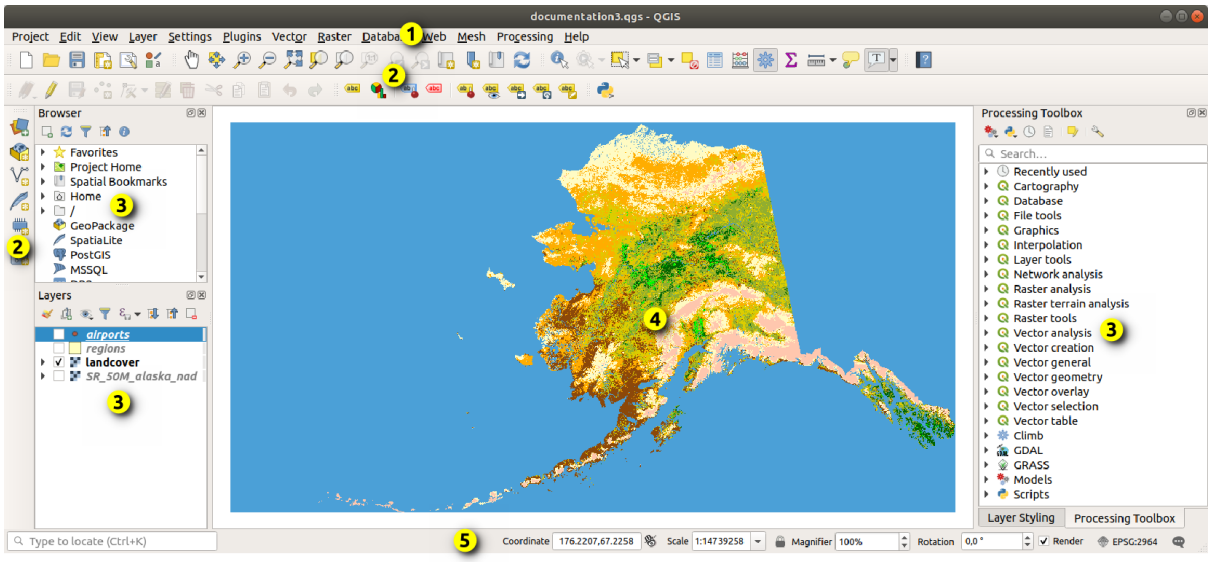


Abb. 3.1: QGIS GUI mit Alaska-Beispieldatensatz

Bemerkung

Das Aussehen einzelner Bereiche (Titelleiste, etc.) kann in Abhängigkeit vom Betriebssystem und den Darstellungs-Einstellungen abweichen.

Das QGIS-Hauptfenster (Abb. 3.1) setzt sich aus den folgenden Bereichen bzw. Elementen zusammen:

1. *Menüleiste*
2. *Werkzeugleisten*
3. *Bedienfelder*

4. *Kartenansicht*
5. *Statusleiste*

Im Folgenden finden Sie nähere Erklärungen zu den einzelnen Punkten.

3.1 Menüleiste

Die Menüleiste ermöglicht den Zugriff auf QGIS-Funktionen über übliche, hierarchisch gegliederte Menüs. Die Menüs, ihre Optionen, die zugehörigen Symbole und Tastaturkürzel werden im Folgenden beschrieben. Die Tastaturkürzel können neu konfiguriert werden (*Einstellungen* ► *Tastaturkürzel*).

Die meisten Menü-Optionen haben ein entsprechendes Werkzeug und umgekehrt. Die Menüs sind jedoch nicht genau wie die Werkzeugleisten organisiert. Die Positionen der Menü-Optionen in den Werkzeugleisten sind in der Tabelle unten angegeben. Plugins können den Menüs neue Optionen hinzufügen. Für weitere Informationen über Werkzeuge und Werkzeugleisten siehe *Werkzeugleisten*.

Bemerkung

QGIS ist für verschiedene Plattformen/Betriebssysteme verfügbar. Die Werkzeuge sind im Allgemeinen auf allen Plattformen verfügbar, können aber je nach Betriebssystem in unterschiedlichen Menüs untergebracht sein. Die nachstehenden Listen zeigen die gängigsten Positionen, einschließlich bekannter Variationen.

3.1.1 Projekt

Das Menü *Projekt* bietet verschiedene Optionen für das Arbeiten mit *Projektdateien* :

- Erstellen Sie mit *Neu* eine Projektdatei von Grund auf oder verwenden Sie eine andere Projektdatei als Vorlage (siehe *Projektdateien* für die Konfiguration von Vorlagen)
- *Öffnen…* eines Projektes aus einer Datei, einem GeoPackage, einer PostgreSQL oder Oracle Datenbank
- *Schließen* eines Projekts oder Zurücksetzen auf den zuletzt gespeicherten Zustand
- *Speichern* eines Projektes im `.qgs` oder `.qgz` Dateiformat, entweder als Datei oder innerhalb einer GeoPackage, PostgreSQL oder Oracle Datenbank
- Exportieren Sie das Kartenfenster in verschiedenen Bild-Formaten oder verwenden Sie ein *Drucklayout* für eine komplexere Ausgabe
- Legen Sie die Projekteigenschaften und die Fangoptionen für die Geometriebearbeitung fest.

Tab. 3.1: Die Einträge im Projekt-Menü

Menüeintrag	Tastenkürzel	Werkzeu	Referenz
Neu	`Ctrl+N`	*Projekt*	*Vorstellung von QGIS-Projekten*
Neu aus Vorlage ►			*Vorstellung von QGIS-Projekten*
Öffnen…	`Strg+O`	*Projekt*	*Vorstellung von QGIS-Projekten*
Öffnen aus ►			*Vorstellung von QGIS-Projekten*
► *GeoPackage…*			*Vorstellung von QGIS-Projekten*
► *PostgreSQL…*			*Vorstellung von QGIS-Projekten*
► *Oracle…*			*Vorstellung von QGIS-Projekten*
Zuletzt verwendetes ►	`Alt+J + R`		*Vorstellung von QGIS-Projekten*
►*Liste leeren*			*Vorstellung von QGIS-Projekten*
Schließen			*Vorstellung von QGIS-Projekten*
Speichern	`Strg+S`	*Projekt*	*Vorstellung von QGIS-Projekten*

Fortsetzung auf der nächsten Seite

Tab. 3.1 – Fortsetzung der vorherigen Seite

Menüeintrag	Tastenkürzel	Werkzeu	Referenz
Speichern als…	Strg+Shift+S	*Projekt*	*Vorstellung von QGIS-Projekten*
Speichern als ►			*Vorstellung von QGIS-Projekten*
► Vorlagen…			*Vorstellung von QGIS-Projekten*
► GeoPackage…			*Vorstellung von QGIS-Projekten*
► PostgreSQL…			*Vorstellung von QGIS-Projekten*
Zurücknehmen…			
Eigenschaften…	Strg+Shift+P		*Projekteigenschaften*
Fangoptionen…			*Optionen für das Einrasten und Digitalisieren*
Import/Export ►			
► Karte als Bild speichern…			*Exportieren der Kartenansicht*
► Karte als PDF exportieren…			*Exportieren der Kartenansicht*
► Projekt als DXF speichern…			*Erstellen neuer DXF-Dateien*
► Layer aus DWG/DXF importieren…			*Importieren einer DXF- oder DWG-Datei*
Neues Drucklayout…	Strg+P	*Projekt*	*Kartenlayout*
Neuer Bericht…			*Einen Bericht erstellen*
Layout-Verwaltung…		*Projekt*	*Kartenlayout*
Layouts ►			*Kartenlayout*
Modelle ►			*Ein Modell erstellen*
QGIS beenden	Strg+Q		

Unter X macOS entspricht der Befehl *QGIS beenden* dem Befehl *QGIS ► QGIS beenden* (Cmd+Q).

3.1.2 Bearbeiten

Das Menü *Bearbeiten* bietet die meisten Standard-Werkzeuge, die zur Bearbeitung von Layer-Attributen oder -Geometrien benötigt werden. Um die Optionen des Menüs *Bearbeiten* zu aktivieren, müssen Sie in den Bearbeitungsmodus wechseln, indem Sie auf Bearbeitungsstatus umschalten klicken (siehe *Editierfunktionen* für Details).

Tab. 3.2: Die Einträge im Bearbeiten-Menü

Menüeintrag	Tastenkürzel	Werkzeugle	Referenz
Rückgängig	Strg+Z	*Digitalisierun*	*Rückgängig und Wiederholen*
Wiederholen	Strg+Shift+Z	*Digitalisierun*	*Rückgängig und Wiederholen*
Ausgewählte Objekte ausschneiden	Strg+X	*Digitalisierun*	*Objekte ausschneiden, kopieren und einfügen*
Objekte kopieren	Strg+C	*Digitalisierun*	*Objekte ausschneiden, kopieren und einfügen*
Objekte einfügen	Strg+V	*Digitalisierun*	*Objekte ausschneiden, kopieren und einfügen*
Objekte einfügen als ►			*Arbeiten mit der Attributtabelle*

Fortsetzung auf der nächsten Seite

Tab. 3.2 – Fortsetzung der vorherigen Seite

Menüeintrag	Tastenkürzel	Werkzeugle	Referenz
► *Neuer Vektorlayer…*			*Arbeiten mit der Attributtabelle*
► *Temporärlayer…*	`Strg+Alt+V`		*Arbeiten mit der Attributtabelle*
Ausgewähltes löschen		*Digitalisierun*	*Ausgewählte Objekte löschen*
Auswahl ►			*Objekte auswählen*
► *Objekte über Rechteck und Einzelklick wählen*		*Auswahl*	*Objekte auswählen*
► *Objekte über Polygon wählen*		*Auswahl*	*Objekte auswählen*
► *Objekte freihändig oder über Einzelklick wählen*		*Auswahl*	*Objekte auswählen*
► *Objekte über Radius oder Einzelklick wählen*		*Auswahl*	*Objekte auswählen*
► *Objekte nach Wert wählen…*	`F3`	*Auswahl*	*Objekte auswählen*
► *Objekte nach Ausdruck wählen…*	`Strg+F3`	*Auswahl*	*Objekte auswählen*
► *Objektauswahl aller Layer aufheben*	`Strg+Alt+A`	*Auswahl*	*Objekte auswählen*
► *Objektauswahl des aktuell aktiven Layers aufheben*	`Strg+Umschalt`	*Auswahl*	*Objekte auswählen*
► *Objekte wiederwählen*			*Objekte auswählen*
► *Alle Objekte wählen*	`Strg+A`	*Auswahl*	*Objekte auswählen*
► *Objektauswahl umkehren*		*Auswahl*	*Objekte auswählen*
Datensatz hinzufügen	`Strg+.`	*Digitalisierun*	
Punktobjekt hinzufügen	`Strg+.`	*Digitalisierun*	*Objekte digitalisieren*
Linienobjekt hinzufügen	`Strg+.`	*Digitalisierun*	*Objekte digitalisieren*
Polygonobjekt hinzufügen	`Strg+.`	*Digitalisierun*	*Objekte digitalisieren*
Beschriftung hinzufügen ►			*Anmerkungen hinzufügen*
► *Beschriftungsformular*		*Anmerkung*	*Anmerkungen hinzufügen*
► *HTML-Beschriftung*		*Anmerkung*	*Anmerkungen hinzufügen*
Attribute bearbeiten ►			
► *Attribute der gewählten Objekte ändern*		*Digitalisierun*	*Bearbeiten von Attributwerten*
► *Attribute gewählter Objekte vereinen*		*Erweiterte Digitalisierun*	*Attribute gewählter Objekte vereinen*
Geometrie bearbeiten ►			
► *Objekt(e) verschieben*		*Erweiterte Digitalisierun*	*Objekt(e) verschieben*
► *Objekt(e) kopieren und verschieben*		*Erweiterte Digitalisierun*	*Objekt(e) verschieben*

Fortsetzung auf der nächsten Seite

Tab. 3.2 – Fortsetzung der vorherigen Seite

Menüeintrag	Tastenkürzel	Werkzeugle	Referenz
► *Objekt(e) drehen*		*Erweiterte Digitalisierun*	*Objekt(e) drehen*
► *Objekt(e) skalieren*		*Erweiterte Digitalisierun*	*Objekt skalieren*
► *Objekt vereinfachen*		*Erweiterte Digitalisierun*	*Objekt vereinfachen*
► *Ring hinzufügen*		*Erweiterte Digitalisierun*	*Ring hinzufügen*
► *Teil hinzufügen*		*Erweiterte Digitalisierun*	*Teil hinzufügen*
► *Ring füllen*		*Erweiterte Digitalisierun*	*Ring füllen*
► *Ring löschen*		*Erweiterte Digitalisierun*	*Ring löschen*
► *Teil löschen*		*Erweiterte Digitalisierun*	*Teil löschen*
► *Objekt überarbeiten*		*Erweiterte Digitalisierun*	*Objekte überarbeiten*
► *Linie versetzen*		*Erweiterte Digitalisierun*	*Linie versetzen*
► *Objekt teilen*		*Erweiterte Digitalisierun*	*Objekte trennen*
► *Teile zerlegen*		*Erweiterte Digitalisierun*	*Teile zerlegen*
► *Gewählte Objekte verschmelzen*		*Erweiterte Digitalisierun*	*Gewählte Objekte verschmelzen*
► *Knotenwerkzeug (alle Layer)*		*Digitalisierun*	*Das Knotenwerkzeug*
► *Knotenwerkzeug (aktiver Layer)*		*Digitalisierun*	*Das Knotenwerkzeug*
► *Linie umkehren*		*Erweiterte Digitalisierun*	*Linie umkehren*
► *Objekt abschneiden/ausdehnen*		*Erweiterte Digitalisierun*	*Objekt abschneiden/ausdehnen*
Punktsymbole drehen		*Erweiterte Digitalisierun*	*Punktsymbole drehen*
Punktsymbolversatz		*Erweiterte Digitalisierun*	*Punktsymbolversatz*

Werkzeuge, die vom ausgewählten Geometrietyp des Layers abhängen (Punkt, Linie oder Polygon) werden entsprechend jeweils nur angezeigt:

Tab. 3.3: Werkzeuge für verschiedene Geometrietypen

Menüeintrag	Punkt	Linie	Polygon
Objekt(e) verschieben			
Objekt(e) kopieren und verschieben			

3.1.3 Sicht (View)

Die meisten Werkzeuge aus dem Menü *Ansicht* dienen dazu, um mit der Kartenansicht (die Darstellung der Karte im zentralen Bereich) zu interagieren. Sie können zum Beispiel:

- Ein zusätzliches 2D- oder 3D-Kartenfensters erstellen (neben der Haupt-Ansicht)
- Auf einen beliebigen Ort *Zoomen oder Verschieben*
- Die Attribute oder Geometrien der angezeigten Objekte abfragen
- Die Kartenansicht durch Kommentare, Dekorationen oder Voransichtsmodi ergänzen
- Alle Bedienfelder und Werkzeugleisten aktivieren

Sie können über das Menü *Ansicht* auch die QGIS-Oberfläche selbst umgestalten:

- *Vollbildmodus umschalten*: erweitert das QGIS-Fenster auf den gesamten Bildschirm und blendet die Titelleiste aus
- *Bedienfeldsichtbarkeit umschalten*: zeigt oder verbirgt aktivierte *Bedienfelder* - nützlich um schnell eine größere Kartenansicht zu erhalten
- *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.

Tab. 3.4: Die Einträge im Ansicht-Menü

Menüeintrag	Tastenkürzel	Werkzeugl	Referenz
Neue Kartenansicht	`Strg+M`	*Kartennavig*	*2D Kartenansichten*
3D-Kartenansichten ►			*3D-Kartenansicht*
► *Neue 3D-Kartenansicht*	`Strg+Alt+M`	*Kartennavig*	*3D-Kartenansicht*
► *3D-Kartenansichten verwalten*			*3D-Kartenansicht*
Karte verschieben		*Kartennavig*	*Funktionen des Kartenfensters*
Karte zur Auswahl verschieben		*Kartennavig*	*Funktionen des Kartenfensters*
Hineinzoomen	`Strg+Alt++`	*Kartennavig*	*Funktionen des Kartenfensters*
Hinaus zoomen	`Strg+Alt+-`	*Kartennavig*	*Funktionen des Kartenfensters*
Datenfilterung ►			*2D Kartenansichten*
► *Zeitsteuerung*		*Kartennavig*	*Das Zeitsteuerung-Bedienfeld*
► *Höhensteuerung*			*Das Bedienelement für die Höhensteuerung*
Objekte abfragen	`Strg+Shift+I`	*Attribute*	*Objekte abfragen*
Messen ►		*Attribute*	*Messen*

Fortsetzung auf der nächsten Seite

Tab. 3.4 – Fortsetzung der vorherigen Seite

Menüeintrag	Tastenkürzel	Werkzeugl	Referenz
► *Linie messen*	Strg+Umschalt+M	*Attribute*	*Messen*
► *Fläche messen*	Strg+Umschalt+J	*Attribute*	*Messen*
► *Kurs messen*		*Attribute*	*Messen*
► *Winkel messen*		*Attribute*	*Messen*
Statistische Zusammenfassung		*Attribute*	*Das Statistik-Bedienfeld*
Geländehöhenprofil			*Geländehöhenprofil*
Volle Ausdehnung	Strg+Shift+F	*Kartennavig*	*Funktionen des Kartenfensters*
Zur Auswahl zoomen	Strg+J	*Kartennavig*	*Funktionen des Kartenfensters*
Auf Layer zoomen		*Kartennavig*	*Funktionen des Kartenfensters*
Auf eigene Auflösung zoomen (100%)		*Kartennavig*	*Funktionen des Kartenfensters*
Zoom zurück		*Kartennavig*	*Funktionen des Kartenfensters*
Zoom vor		*Kartennavig*	*Funktionen des Kartenfensters*
Dekorationen ►	Alt+V + D		*Dekorationen*
► *Gitter…*			*Gitter*
► *Maßstab…*			*Maßstab*
► *Bild…*			*Bild Dekoration*
► *Nordpfeil…*			*Nordpfeil*
► *Titelbeschriftung…*			*Titelbeschriftung*
► *Urheberrechtshinweis…*			*Urheberrechtshinweis*
► *Layoutausdehnung…*			*Layoutausdehnung*
Vorschaumodus ►			
► *Normal*			
► *Monochrom simulieren*			
► *Farbenblindheit simulieren (Graustufen)*			
► *Farbenblindheit simulieren (kein Rot)*			
► *Farbenblindheit simulieren (kein Grün)*			
► *Farbenblindheit simulieren (kein Blau)*			
Kartenhinweise anzeigen		*Attribute*	*Anzeige*
Neues räumliche Lesezeichen…	Ctrl+B	*Kartennavig*	*Räumliche Lesezeichen*
Räumliche Lesezeichen anzeigen	Strg+Umschalt+B	*Kartennavig*	*Räumliche Lesezeichen*
Räumliche Lesezeichenverwaltung anzeigen			*Räumliche Lesezeichen*
Aktualisieren	F5	*Kartennavig*	
Layersichtbarkeit►			*Layer-Bedienfeld*
► *Alle Layer anzeigen*	Strg+Shift+U		*Layer-Bedienfeld*

Fortsetzung auf der nächsten Seite

Tab. 3.4 – Fortsetzung der vorherigen Seite

Menüeintrag	Tastenkürzel	Werkzeugl	Referenz
► *Alle Layer ausblenden*	Strg+Shift+H		*Layer-Bedienfeld*
► *Zeige gewählte Layer*			*Layer-Bedienfeld*
► *Ausblenden von gewählten Layern*			*Layer-Bedienfeld*
► *Gewählte Layer umschalten*			*Layer-Bedienfeld*
► *Gewählte Layer einzeln umschalten*			*Layer-Bedienfeld*
► *Abgewählte Layer verbergen*			*Layer-Bedienfeld*
Bedienfelder ►			*Bedienfelder und Werkzeugkästen*
Werkzeugleiste ►			*Bedienfelder und Werkzeugkästen*
Volle Ausdehnung	F11		
Bedienfeldsichtbarkeit umschalten	Strg+Tab		
Nur Karte umschalten	Strg+Umschalt+Ta		

Unter Linux KDE befinden sich *Bedienfelder* ►, *Werkzeugleisten*► und *Vollbildmodus umschalten* im Menü *Einstellungen*.

3.1.4 Layer

Das *Layer* Menü bietet eine große Auswahl an Werkzeugen, um neue Datenquellen zu *erstellen*, diese zu einem Projekt *hinzuzufügen* oder zu *bearbeiten* . Darüber hinaus finden Sie in diesem Menü Werkzeuge, um:

- Einen Layer zu *duplizieren* und so eine Kopie zu erhalten, bei der sie Namen, Stil (Symbologie, Beschriftungen, …), Verknüpfungen, … verändern können. Das Duplikat benutzt dabei die gleiche Datenquelle wie der ursprüngliche Layer.
- Layer oder Layer-Gruppen *Kopieren* und in ein anderes Projekt *Einfügen*. Wie bei *Duplizieren* basieren die Layer weiterhin auf der gleichen Datenquelle, die Layer-Eigenschaften in dem neuen Projekt können aber unabhängig davon geändert werden.
- Oder Sie können *Eingebettete Layer und Gruppen…* aus einem anderen Projekt als Nur-Lesen-Kopien einfügen, deren Eigenschaften nur im Ursprungs-Projekt geändert werden können (siehe *Layer aus externen Projekten einbinden*).

Das *Layer* Menü enthält auch Werkzeuge, um Layer-Eigenschaften zu kopieren und einzufügen (Symbologie, Maßstab, KBS, …).

Tab. 3.5: Die Einträge im Layer-Menü

Menüeintrag	Tastenkürzel	Werkzeugleiste	Referenz
Datenquellenverwaltung	Strg+L	*Datenquellenverwa*	*Datenquellenverwaltung*
Layer erzeugen ►			*Neue Vektorlayer erstellen*
► *Neuer GeoPackage-Layer…*	Strg+Umschalt	*Datenquellenverwa*	*Einen neuen GeoPackage-Layer erstellen*
► *Neuer Shapefile-Layer…*		*Datenquellenverwa*	*Einen neuen Shapefile-Layer erstellen*
► *Neuer SpatiaLite-Layer…*		*Datenquellenverwa*	*Erstellen eines neuen SpatiaLite Layers*

Fortsetzung auf der nächsten Seite

Tab. 3.5 – Fortsetzung der vorherigen Seite

Menüeintrag	Tastenkürzel	Werkzeugleiste	Referenz
► *Neuer Temporärlayer…*		*Datenquellenverwa*	*Erstellen eines neuen temporären Scratch Layers*
► *Neuer Netzlayer…*		*Datenquellenverwa*	*Erstellen eines neuen Netzlayers*
► *Neuer GPX-Layer…*		*Datenquellenverwa*	*Erstellen eines neuen GPX-Layers*
► *Neuer virtueller Layer…*		*Datenquellenverwa*	*Erstellen von virtuellen Layern*
Layer hinzufügen ►			*Öffnen von Daten*
► *Vectorlayer hinzufügen…*	`Strg+Shift+V`	*Layerverwaltungsw*	*Laden eines Layers aus einer Datei*
► *Rasterlayer hinzufügen…*	`Ctrl+Shift+R`	*Layerverwaltungsw*	*Laden eines Layers aus einer Datei*
► *Netzlayer hinzufügen…*		*Layerverwaltungsw*	*Einen Netz-Layer laden*
► *Getrennte Textdatei als Layer hinzufügen…*	`Strg+Umschalt`	*Layerverwaltungsw*	*Importieren einer Textdatei (mit Trennzeichen)*
► *Add PostgreSQL Layer…*	`Strg+Umschalt`	*Layerverwaltungsw*	*Datenbankbezogene Werkzeuge*
► *SpatiaLite-Layer hinzufügen…*	`Strg+Umschalt`	*Layerverwaltungsw*	*SpatiaLite-Layer*
► *MS-SQL-Server-Layer hinzufügen…*		*Layerverwaltungsw*	*Datenbankbezogene Werkzeuge*
► *Oracle Spatial-Layer hinzufügen…*		*Layerverwaltungsw*	*Datenbankbezogene Werkzeuge*
► *Räumlichen SAP-HANA-Layer hinzufügen…*		*Layerverwaltungsw*	*Datenbankbezogene Werkzeuge*
► *Virtuellen Layer hinzufügen/bearbeiten…*		*Layerverwaltungsw*	*Erstellen von virtuellen Layern*
► *WMS/WMTS-Layer hinzufügen…*	`Strg+Umschalt`	*Layerverwaltungsw*	*WMS/WMTS Layer laden*
► *XYZ-Layer hinzufügen…*			*Verwendung von XYZ Tile-Diensten*
► *WCS-Layer hinzufügen…*		*Layerverwaltungsw*	*WCS Client*
► *WFS/OGC API-Objektlayer hinzufügen…*		*Layerverwaltungsw*	*WFS und WFS-T Klient*
► *ArcGIS-REST Server-Layer hinzufügen…*		*Layerverwaltungsw*	*Verwendung von ArcGIS REST-Servern*
► *Vektorkachellayer hinzufügen…*			*Verwendung von Vector Tiles-Diensten (Vektorkacheln)*
► *Punktwolkenlayer hinzufügen…*			*Arbeiten mit Punktwolken*
► *Neuen GPX-Layer hinzufügen…*			*Einführung in GNSS/GPS-Daten*

Fortsetzung auf der nächsten Seite

Tab. 3.5 – Fortsetzung der vorherigen Seite

Menüeintrag	Tastenkürzel	Werkzeugleiste	Referenz
Eingebettete Layer und Gruppen…			*Layer aus externen Projekten einbinden*
Aus Layerdefinitionsdatei hinzufügen…			*Layerdefinitionsdatei*
Georeferenzierung…			*Georeferenzierung*
Stil kopieren			*Layer-Eigenschaften speichern und teilen*
Stil einfügen			*Layer-Eigenschaften speichern und teilen*
Layer kopieren			
Layer/Gruppe einfügen			
Attributtabelle öffnen	`F6`	*Attribute*	*Arbeiten mit der Attributtabelle*
Attributtabelle filtern ►			*Arbeiten mit der Attributtabelle*
► *Attributtabelle öffnen (gewählte Objekte)*	`Umschalt+F6`	*Attribute*	*Arbeiten mit der Attributtabelle*
► *Attributtabelle öffnen (sichtbare Objekte)*	`Strg+F6`	*Attribute*	*Arbeiten mit der Attributtabelle*
► *Attributtabelle öffnen (geänderte und neue Objekte)*		*Attribute*	*Arbeiten mit der Attributtabelle*
Bearbeitungsstatus umschalten		*Digitalisierung*	*Einen vorhandenen Layer editieren*
Layeränderungen speichern		*Digitalisierung*	*Änderungen speichern*
Aktuelle Änderungen ►		*Digitalisierung*	*Änderungen speichern*
► *Speichern gewählter Layer*		*Digitalisierung*	*Änderungen speichern*
► *Verwerfen für gewählter Layer*		*Digitalisierung*	*Änderungen speichern*
► *Abbruch für gewählter Layer*		*Digitalisierung*	*Änderungen speichern*
► *Alle Layer speichern*		*Digitalisierung*	*Änderungen speichern*
► *Alle Layeränderungen verwerfen*		*Digitalisierung*	*Änderungen speichern*
► *Abbruch für alle Layer*		*Digitalisierung*	*Änderungen speichern*
Speichern als…			*Erstellen neuer Layer aus einem vorhandenen Layer*
Als Layerdefinitionsdatei speichern…			*Layerdefinitionsdatei*
Layer/Gruppe löschen	`Strg+D`		
Layer duplizieren			
Maßstabsabhängige Sichtbarkeit des/der Layer setzen			*Einstellen des Sichtbarkeitsmaßstabs*
KBS von Layer(n) setzen	`Strg+Shift+C`		*Koordinatenbezugssysteme für Layer*
Layer-KBS dem Projekt zuweisen			*Koordinatenbezugssysteme für Projekte*

Fortsetzung auf der nächsten Seite

Tab. 3.5 – Fortsetzung der vorherigen Seite

Menüeintrag	Tastenkürzel	Werkzeugleiste	Referenz
Layereigenschaften…			• *vector* • *raster* • *Netzlayer* • *Punktwolkenlayer* • *Vektorkachellayer*
Filter…	`Strg+F`		*Abfrageerstellung*
Beschriftung			*Beschriftungen Eigenschaften*
In der Übersicht anzeigen			*Das Übersicht-Bedienfeld*
Alle in der Übersicht anzeigen			*Das Übersicht-Bedienfeld*
Alle aus Übersicht entfernen			*Das Übersicht-Bedienfeld*

3.1.5 Einstellungen

Tab. 3.6: Die Einträge im Einstellungen-Menü

Menüeintrag	Referenz
Benutzerprofile ►	*Arbeiten mit Benutzerprofilen*
► *Standard*	*Arbeiten mit Benutzerprofilen*
► *Aktuelles Profilverzeichnis öffnen*	*Arbeiten mit Benutzerprofilen*
► *Neues Profil…*	*Arbeiten mit Benutzerprofilen*
Stilverwaltung…	*Die Stilverwaltung*
Benutzerprojektionen…	*Ein eigenes Koordinatenbezugssystem definieren*
Tastenkürzel…	*Tastenkürzel*
Oberflächenanpassung…	*Anpassung*
Optionen…	*Optionen*

Unter Linux KDE finden Sie im Menü *Einstellungen* noch weitere Werkzeuge, wie *Bedienfelder* ►, *Werkzeugleisten* ► und *Vollbildmodus umschalten*.

3.1.6 Erweiterungen

Tab. 3.7: Die Einträge im Erweiterungen-Menü

Menüeintrag	Tastenkürzel	Werkzeugleiste	Referenz
Erweiterungen verwalten und installieren…			*Der Erweiterungen Dialog*
Python-Konsole	`Strg+Alt+P`	*Erweiterungswerk:*	*QGIS Python Konsole*

Wenn Sie QGIS das erste Mal starten werden nicht alle Erweiterungen geladen.

3.1.7 Vektor

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…*.

Bemerkung

The list of tools and their menu can be modified/extended with any Processing algorithms (read *Konfigurieren der Verarbeitungsumgebung*) or by some third-party *plugins*.

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

Tab. 3.8: Die Einträge im Vektor-Menü

Menüeintrag	Tastenkürzel	Werkzeu	Referenz
Geometrien prüfen…			*Die Erweiterung Geometrien prüfen*
Topologie-Prüfung		*Vektor*	*Die Topologie-prüfen-Erweiterung*
Geoverarbeitungswerkzeuge ►	`Alt+O + G`		
► *Puffer…*			*Puffer*
► *Zuschneiden…*			*Zuschneiden*
► *Konvexe Hülle…*			*Convex hull*
► *Differenz…*			*Difference*
► *Auflösen…*			*Dissolve*
► *Verschneidung…*			*Intersection*
► *Symmetrische Differenz…*			*Symmetrical difference*
► *Vereinigungen…*			*Union*
► *Ausgewählte Polygone entfernen…*			*Eliminate selected polygons*
Geometrie-Werkzeuge ►	`Alt+O + E`		
► *Zentroide…*			*Centroids*
► *Geometrien sammeln…*			*Collect geometries*
► *Stützpunkte extrahieren…*			*Extract vertices*
► *Mehr- zu einteilig…*			*Multipart to singleparts*
► *Polygone zu Linien…*			*Polygons to lines*
► *Vereinfachen…*			*Simplify*
► *Gültigkeit prüfen…*			*Check validity*
► *Delaunay-Triangulation…*			*Delaunay triangulation*
► *Nach Anzahl verdichten…*			*Densify by count*
► *Geometrieattribute hinzufügen…*			*Geometrieattribute hinzufügen*
► *Linien zu Polygonen…*			*Lines to polygons*
► *Voronoi-Polygone…*			*Voronoi polygons*
Analyse-Werkzeuge ►	`Alt+O + A`		

Fortsetzung auf der nächsten Seite

Tab. 3.8 – Fortsetzung der vorherigen Seite

Menüeintrag	Tastenkürzel	Werkzeu	Referenz
► *Linienschnittpunkte…*			*Line intersections*
► *Mittlere Koordinate(n)…*			*Mean coordinate(s)*
► *Grundstatistik für Felder…*			*Grundstatistik für Felder*
► *Punkte in Polygon zählen…*			*Count points in polygon*
► *Distanzmatrix…*			*Distance matrix*
► *Eindeutige Werte auflisten…*			*List unique values*
► *Nächster Nachbaranalyse…*			*Nearest neighbour analysis*
► *Linienlängen summieren…*			*Sum line lengths*
Datenmanagement-Werkzeuge ►	Alt+O + D		
► *Vektorlayer zusammenführen…*			*Merge vector layers*
► *Layer reprojizieren…*			*Reproject layer*
► *Räumlichen Index erzeugen…*			*Create spatial index*
► *Attribute nach Position verknüpfen…*			*Join attributes by location*
► *Vektorlayer teilen…*			*Split vector layer*
Forschungswerkzeuge ►	Alt+O + R		
► *Gitter erzeugen…*			*Create grid*
► *Layerausdehnung extrahieren…*			*Extract layer extent*
► *Zufällige Punkte in Grenzen…*			*Random points in extent*
► *Zufällige Punkte in Polygonen…*			*Random points in polygons*
► *Zufällige Punkte aus Linien…*			*Random points on lines*
► *Nach Position selektieren…*			*Auswahl nach der Lage*
► *Innerhalb einer Entfernung wählen…*			*Select within distance*
► *Zufällige Punkte in Layergrenzen…*			*Random points in layer bounds*
► *Zufällige Punkte in Polygonen…*			*Random points inside polygons*
► *Zufällige Auswahl…*			*Random selection*
► *Zufällige Auswahl in Untermengen…*			*Random selection within subsets*
► *Regelmäßige Punkte…*			*Regular points*

3.1.8 Raster

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…*.

Bemerkung

The list of tools and their menu can be modified/extended with any Processing algorithms (read *Konfigurieren der Verarbeitungsumgebung*) or by some third-party *plugins*.

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

Tab. 3.9: Die Einträge im Raster-Menü

Menüeintrag	Tastenkürzel	Werkz	Referenz
Rasterrechner…			*Rasterrechner*
Raster ausrichten…			*Align rasters*
Analyse ►			
► *Perspektive…*			*Aspect*
► *„Leerwert“ füllen…*			*Fill NoData*
► *Gitter (Beweglicher Mittelwert)…*			*Grid (Moving average)*
► *Gitter (Datenkriterien)…*			*Grid (Data metrics)*
► *Gitter (Inverse Distanz zu einer Potenz)…*			*Grid (Inverse distance to a power)*
► *Gitter (Nächster Nachbar)…*			*Grid (IDW with nearest neighbor searching)*
► *Fast schwarz…*			*Near black*
► *Schummerung…*			*Schummerung*
► *Nähe (Rasterabstand)…*			*Proximity (raster distance)*
► *Rauhigkeit…*			*Roughness*
► *Sieben…*			*Sieve*
► *Hang…*			*Slope*
► *Topographische Position (TPI)…*			*Topographic Position Index (TPI)*
► *Oberflächenrauhigkeit (TRI)…*			*Terrain Ruggedness Index (TRI)*
Projektionen ►			
► *Projektion zuweisen…*			*Projektion zuweisen*
► *Projektion extrahieren…*			*Projektion extrahieren*
► *Transformieren (Reprojizieren)…*			*Warp (reproject)*
Sonstiges ►			
► *Virtuelles Raster generieren…*			*Build virtual raster*
► *Rasterinformation…*			*Raster information*
► *Verschmelzen…*			*Vereinigen*
► *Übersichten erzeugen (Pyramiden)…*			*Build overviews (pyramids)*
► *Kachelindex…*			*Tile index*
Extraktion ►			
► *Raster auf Ausdehnung zuschneiden…*			*Clip raster by extent*
► *Raster auf Layermaske zuschneiden…*			*Clip raster by mask layer*
► *Kontur…*			*Contour*
Konvertierung ►			
► *PCT nach RGB…*			*PCT to RGB*

Fortsetzung auf der nächsten Seite

Tab. 3.9 – Fortsetzung der vorherigen Seite

Menüeintrag	Tastenkürzel	Werkz	Referenz
► *Vektorisieren (Raster nach Vektor)…*			*Polygonize (raster to vector)*
► *Rastern (Vektor nach Raster)…*			*Rasterize (vector to raster)*
► *RGB nach PCT…*			*RGB to PCT*
► *Umwandeln (Format konvertieren)…*			*Translate (convert format)*

3.1.9 Datenbank

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.

Tab. 3.10: Die Einträge im Datenbank-Menü

Menüeintrag	Tastenkürzel	Werkzeugleist	Referenz
Offline-Bearbeitung –>	`Alt+D + O`		*Offline-Bearbeitung Erweiterung*
► *In Offline-Projekt konvertieren…*		*Database*	*Offline-Bearbeitung Erweiterung*
► *Synchronisieren*		*Database*	*Offline-Bearbeitung Erweiterung*
Query History…		*Database*	*Creating SQL Query Layers*
DB-Verwaltung…		*Database*	*DB Verwaltung*

Wenn Sie QGIS das erste Mal starten werden nicht alle Erweiterungen geladen.

3.1.10 Web

Dies sind die Punkte, die Sie im *Web* Menü finden, wenn alle Kernerweiterungen aktiviert sind. Wenn keine Web-Erweiterung aktiviert ist, ist das *Datenbank* Menü nicht sichtbar.

Tab. 3.11: Die Einträge im Web-Menü

Menüeintrag	Tastenkürzel	Werkzeugleist	Referenz
MetaSearch ►	`Alt+W + M`		*MetaSearch Catalog Client*
► *Metasearch*		*Web*	*MetaSearch Catalog Client*
► *Help*			*MetaSearch Catalog Client*

Wenn Sie QGIS das erste Mal starten werden nicht alle Erweiterungen geladen.

3.1.11 Netz

Das *Netz* Menü bietet Werkzeuge, die für die Bearbeitung von *Netz-Layern* benötigt werden. Erweiterungen können diesem Menü neue Punkte hinzufügen.

Tab. 3.12: Die Einträge im Netz-Menü

Menüeintrag	Tastenkürzel	Werkzeugleist	Referenz
Netzrechner…			*Mesh Calculator*
Oberflächen und Stützpunkte reindizieren			*Reindexing meshes*

3.1.12 Processing

Tab. 3.13: Die Einträge im Verarbeitung-Menü

Menüeintrag	Tastenkürzel	Werkzeu	Referenz
Werkzeugkiste	`Ctrl+Alt+T`		*Die Werkzeugkiste*
Modellentwurf…	`Strg+Alt+G`		*Ein Modell erstellen*
Protokoll…	`Strg+Alt+H`		*Das Protokoll*
Ergebnisanzeige	`Strg+Alt+R`		*Konfiguration externer Anwendungen*
Objekte insitu ändern			*Mit Verarbeitung Objekte innerhalb eines Layers verändern*

Wenn Sie QGIS das erste Mal starten werden nicht alle Erweiterungen geladen.

3.1.13 Hilfe

Tab. 3.14: Die Einträge im Hilfe-Menü

Menüeintrag	Tastenkürzel	Werkzeugleist	Referenz
Hilfe-Übersicht	`F1`	*Direkthilfe*	
API Dokumentation			
Erweiterungen ►			
Ein Problem melden			
Brauchen Sie professionelle Unterstützung?			
QGIS-Homepage	`Strg+H`		
Check QGIS-Version überprüfen			
Über			
Dauerhafte QGIS-Fördermitgliedschaften			

3.1.14 QGIS

Dieses Menü ist nur unter macOS verfügbar und enthält einige betriebssystem-spezifische Befehle.

Tab. 3.15: Die Einträge im QGIS-Menü

Menüeintrag	Tastenkürzel	Referenz
Über QGIS		
Preferences	`Cmd+,`	*Optionen*
QGIS verbergen	`Cmd+H`	
Alle anzeigen		
Andere verbergen	`Shift+Cmd+H`	
QGIS beenden	`Cmd+Q`	

Einstellungen entspricht *Einstellungen* ► *Optionen*, *Über QGIS* entspricht *Hilfe* ► *Über* und *QGIS beenden* entspricht *Projekt* ► *QGIS beenden* in anderen Betriebssystemen.

3.2 Bedienfelder und Werkzeugkästen

Über das Menü *Ansicht* (oder *Einstellungen*) können Sie (*Bedienfelder* ►) und Werkzeugleisten (*Werkzeugkästen* ►) ein- und ausschalten. Bedienfelder und Werkzeugleisten können ebenfalls (de)aktiviert werden, indem Sie mit der rechten Maustaste auf die Menü- oder Symbolleiste klicken und das gewünschte Element auswählen. Beides lässt sich innerhalb der QGIS-Oberfläche beliebig verschieben und platzieren. Die Liste kann durch die Aktivierung von Kern- oder externen *Erweiterungen* erweitert werden.

3.2.1 Werkzeugleisten

Über die Werkzeugleisten haben Sie Zugriff auf die meisten Funktionen in den Menüs sowie auf zusätzliche Werkzeuge zur Interaktion mit der Karte. Zu jedem Element der Werkzeugleiste gibt es eine Pop-up-Hilfe: fahren Sie mit der Maus über das Element und es wird eine kurze Beschreibung des Werkzeugs angezeigt.

In der folgenden Tabelle sind die verfügbaren Werkzeugleisten aufgeführt:

Tab. 3.16: QGIS Werkzeugleisten

Name der Werkzeugleiste	Wichtigste Referenz für das Werkzeug
Erweiterte Digitalisierung	*Erweiterte Digitalisierung*
Anmerkung	*Anmerkungen hinzufügen*
Attribute	*Arbeiten mit der Attributtabelle*, *Allgemeine Werkzeuge*
Datenquellenverwaltungsleiste	*Verwaltung von Datenquellen*
Database	*DB Verwaltung*
Digitalisierung	*Einen vorhandenen Layer editieren*
GPS	*Live-GPS-Verfolgung*
GRASS	*GRASS GIS Integration*
Direkthilfe	
Beschriftung	*Die Beschriftungswerkzeugleiste*
Layerverwaltungswerkzeugleiste	*Öffnen von Daten*
Kartennavigation	*Funktionen des Kartenfensters*
Netzdigitalisierung	*Editing a mesh layer*
Erweiterungswerkzeugleiste	*Erweiterungen*
Projekt	*Arbeiten mit Projektdateien*, *Kartenlayout*, *Die Stilbibliothek*
Processing Algorithms	*Konfigurieren der Verarbeitungsumgebung*
Raster	*Erweiterungen*
Auswahl	*Objekte auswählen*
Werkzeugleiste für Formen	*Formen digitalisieren*
Einrastwerkzeugleiste	*Einrasttoleranz und Suchradius einstellen*
Vektor	*Erweiterungen*
Web	*Erweiterungen*, *MetaSearch Catalog Client*

Bemerkung

Externe Erweiterungen können die Standard-Werkzeugleisten um eigene Werkzeuge ergänzen oder auch ganz neue Werkzeugleisten zu QGIS hinzufügen.

Tipp

Werkzeugleiste wiederherstellen

Wenn Sie versehentlich eine Werkzeugleiste ausgeblendet haben, können Sie diese mit *Ansicht* ► *Werkzeugkästen* ► (oder *Einstellungen* ► *Werkzeugkästen* ►) wieder einblenden. Wenn aus irgendeinem Grund eine Werkzeugleiste (oder ein anderes Element) komplett von der Oberfläche verschwunden ist, finden Sie Tipps, wie Sie sie wiederherstellen können, unter *Wiederherstellung der ursprünglichen GUI*.

3.2.2 Bedienfelder

QGIS bietet viele Bedienfelder. Bedienfelder sind spezielle Elemente, mit denen Sie interagieren können (Auswahl von Optionen, Aktivieren von Kästchen, Eintragen von Werten…), um komplexere Aufgaben durchzuführen.

Die folgende Liste zeigt die Standard Bedienfelder, die von QGIS zur Verfügung gestellt werden:

Tab. 3.17: Bedienfelder

Name des Bedienfelds	Tastenkürzel	Referenz
Erweiterte Digitalisierung	`Ctrl+4`	*Das Bedienfeld Erweiterte Digitalisierung*
Browser	`Ctrl+2`	*Das Browser-Bedienfeld*
Browser (2)		*Das Browser-Bedienfeld*
Diagnose-/Entwicklungswerkzeuge	`F12`	*Das Diagnose/Entwicklungswerkzeuge-Bedienfeld*
Höhenprofil		*Geländehöhenprofil*
Geometrie Überprüfung		*Digitalisieren*
GPS-Information	`Ctrl+0`	*Live-GPS-Verfolgung*
GRASS Werkzeuge		*GRASS GIS Integration*
Layerreihenfolge	`Ctrl+9`	*Das Layerreihenfolge-Bedienfeld*
Layergestaltung	`Ctrl+3`	*Das Layergestaltung-Bedienfeld*
Layer	`Ctrl+1`	*Layer-Bedienfeld*
Protokoll		*Das Protokoll-Bedienfeld*
Übersicht	`Ctrl+8`	*Das Übersicht-Bedienfeld*
Verarbeitungswerkzeuge		*Die Werkzeugkiste*
Ergebnisanzeige		*Die Werkzeugkiste*
Fangen- und Digitalisierungsoptionen		*Einrasttoleranz und Suchradius einstellen*
Räumliche Lesezeichenverwaltung	`Ctrl+7`	*Räumliche Lesezeichen*
Statistik	`Ctrl+6`	*Das Statistik-Bedienfeld*
Zeitsteuerung		*Das Zeitsteuerung-Bedienfeld*
Kachelmaßstab		*Tilesets*
Rücknahme/Wiederholung	`Ctrl+5`	*Das Rücknahme/Wiederholung-Bedienfeld*
Stützpunktbearbeitung		*Das Bedienfeld Stützpunktbearbeitung*

3.3 Statusleiste

In der Statusleiste finden Sie allgemeine Informationen über die Kartenansicht, über den Status während einer Verarbeitung und zu verfügbaren Aktionen. Außerdem stehen Ihnen Werkzeuge zur Steuerung der Kartenansicht zur Verfügung.

3.3.1 Suchleiste

Auf der linken Seite der Statusleiste befindet sich die Suchleiste, ein Werkzeug mit dem Sie jedes Objekt oder jede Option in QGIS finden und ausführen können:

1. Klicken Sie in das Textfeld oder drücken Sie `Strg+K` um die Suchleiste zu aktivieren.
2. Geben Sie einen Text ein, der mit dem gesuchten Objekt verknüpft ist (Name, Tag, Schlüsselwort, …). Dabei ist es möglich, die Suche auf einen bestimmten Bereich zu beschränken, indem Sie Ihrem Text einen *Such-Filter* voranstellen (beim Aktivieren der Suchleiste werden die verfügbaren Filter angezeigt). Bei der Eingabe von `l cad` wird z.B. nur Layer zurückgeben, deren Name `cad` enthält.

   Der Such-Filter kann auch mit einem Doppelklick aus der Liste ausgewählt werden, wenn diese beim Aktivieren der Suchleiste angezeigt wird.
3. Klicken Sie auf das gewünschte Ergebnis, um eine entsprechende Aktion bzw. das entsprechende Element auszuwählen.

Tipp

Beschränkung der Suche auf bestimmte Felder in der Attributtabelle des aktiven Layers

Standardmäßig durchläuft eine Suche mit dem Filter „Objekte des aktiven Layers“ (`f`) die gesamte Attributtabelle des betreffenden Layers. Sie können die Suche auf ein bestimmtes Feld beschränken, indem Sie das Präfix `@` verwenden. So gibt z.B. `f @name sal` oder `@name sal` nur die Objekte zurück, deren Feld „name“ die Zeichenfolge ‚sal‘ enthält. Die automatische Textvervollständigung ist beim Schreiben aktiv und der Vorschlag kann mit der Taste `Tab` übernommen werden.

Eine erweiterte Kontrolle über die abgefragten Felder ist über den Reiter *Felder* der Layereigenschaften möglich. Lesen Sie das Kap. *Eigenschaften Felder* für Details.

Die Suche wird über Threads abgewickelt, so dass die Ergebnisse angezeigt werden können, sobald ein Filter auf sie stößt, auch wenn die Suche noch anhält. Das bedeutet, dass z. B. ein Dateisuchfilter die Ergebnisse nacheinander anzeigt, während der Dateibaum noch durchsucht wird. Dadurch wird sichergestellt, dass die Benutzeroberfläche immer ansprechbar ist, auch wenn eine zeitintensive Suche durchgeführt wird (z. B. wenn ein Online-Dienst durchsucht wird).

Bemerkung

Der Nominatim-Suchfilter verhält sich möglicherweise anders (keine automatische Vervollständigungssuche, Verzögerung beim Abrufen von Ergebnissen, …) da die OpenStreetMap-Nominatim-Nutzungsrichtlinie Anwendung finden.

Tipp

Schneller Zugriff auf die Konfigurationen des Locators

Klicken Sie auf das Suchen-Symbol im Textfeld der Suchleiste, um die Liste der Filter anzuzeigen, die Sie verwenden können oder klicken Sie dann auf den Eintrag *Konfigurieren* am Ende der Liste, um direkt zu den Einstellungen der Suche zu gelangen (*Einstellungen* ► *Optionen…* Registerkarte *Suche*).

3.3.2 Anzeige von Aktionen

Im Bereich neben der Suchleiste werden ggf. Informationen zu durchgeführten Aktionen angezeigt (z. B. Auswahl von Objekten in einem Layer, Entfernen von Layern, Entfernung und Richtung beim Verschieben) oder eine lange Beschreibung des Werkzeugs, über das Sie den Mauszeiger halten (nicht für alle Werkzeuge verfügbar).

Bei langwierigen Operationen, wie z. B. dem Erstellen von Statistiken von Rasterlayern, dem Ausführen von Verarbeitungsalgorithmen oder dem Rendern mehrerer Layer in der Kartenansicht, wird ein Fortschrittsbalken in der Statusleiste angezeigt.

3.3.3 Steuerung der Kartenansicht

Die Funktion *Koordinate* zeigt die aktuelle Position der Maus an und ändert sich, wenn Sie den Mauszeiger über die Kartenansicht bewegen. Sie können die Einheiten (und die Genauigkeit) dieser Anzeige im Menü *Projekt* ► *Eigenschaften…* ► *Allgemein* einstellen. Wenn Sie auf die kleine Schaltfläche links neben dem Textfeld klicken, können Sie zwischen den Optionen *Koordinate* und *Ausdehnung* umschalten. Für die Ausdehnung werden die aktuellen Koordinaten der unteren linken und oberen rechten Ecke der Kartenansicht in den eingestellten Karteneinheiten anzeigt.

Neben der Koordinatenanzeige wird der jeweils aktuelle *Maßstab* der Kartenansicht angezeigt. Es gibt einen Maßstabsselektor, mit dem Sie zwischen *vordefinierten Maßstäben* wählen können.

Klicken Sie auf der rechten Seite der Maßstabsanzeige auf die Schaltfläche , um den Maßstab zu sperren. Wenn Sie jetzt mit Lupe oder Mausrad die Ansicht Vergrößern oder Verkleinern können Sie in die Kartenansicht hineinzoomen, ohne den Maßstab zu verändern. Das ist hilfreich, um z.B. die Positionen von Beschriftungen und Symbolen genauer sehen und einstellen zu können. Die Vergrößerungsstufe wird neben dem Maßstab in Prozent angegeben. Wenn die *Vergrößerung* einen Wert von 100% hat, wird die aktuelle Karte nicht vergrößert, d.h. sie wird maßstabsgetreu im Verhältnis zur Auflösung des Monitors (DPI) dargestellt. Ein Standardwert für die Vergrößerung kann unter *Einstellungen* ► *Optionen* ► *Darstellung* ► *Zeichenverhalten* definiert werden, was bei hochauflösenden Bildschirmen sehr nützlich sein kann, um kleine Symbole zu vergrößern. Zusätzlich kann unter *Einstellungen* ► *Optionen* ► *Karte & Legende* ► *DPI* eingestellt werden, ob QGIS die physikalische DPI des jeweiligen Monitors berücksichtigt oder die logische DPI des Gesamtsystems verwendet.

Rechts neben dem Vergrößerungswerkzeug können Sie eine Drehung (im Uhrzeigersinn) für die Kartenansicht in Grad festlegen.

Auf der rechten Seite der Statusleiste kann das Kontrollkästchen *Zeichnen* verwendet werden, um das Rendering der Kartenansicht vorübergehend auszuschalten (siehe Abschnitt *Steuerung der Kartendarstellung*).

Rechts neben dem *Zeichnen* Kontrollkästchen befindet sich die Schaltfläche *EPSG:code*, die das aktuelle Projekt-KBS anzeigt. Wenn Sie darauf klicken, öffnet sich die *Projekteigenschaften* mit dem Reiter *KBS* und Sie können die Kartenansicht neu projizieren (oder auch andere Projekteigenschaften anpassen).

Tipp

Die richtige Maßstabseinheit im Kartenfenster einstellen

Wenn Sie QGIS starten, ist das Standard KBS `WGS 84 (EPSG 4326)` und die Einheiten sind Grad. Das bedeutet, dass QGIS die Koordinate in Grad anzeigt. Wenn Sie eine andere Anzeige wünschen (oder auch um ggf. eine korrekte Anzeige zu erhalten) können Sie diese Einstellung entweder manuell im Menü *Projekt* ► *Eigenschaften…* unter dem Reiter *Allgemein* ändern (z.B. in Meter). Oder Sie können über das Symbol *EPSG:code* (siehe oben) das korrekte KBS einstellen. Dadurch werden die Einheiten so gesetzt, wie es die Projektion vorgibt (z.B. `+units=us-ft`).

Beachten Sie, dass das Standard-KBS für neu angelegte Projekte in *Einstellungen* ► *Optionen* ► *KBS und Transformationen* ► *KBS-Behandlung* eingestellt werden kann.

3.3.4 Meldungen

Die Schaltfläche

ganz rechts in der Statusleiste öffnet das *Protokoll* Bedienfeld, das Informationen über Hintergrundprozesse enthält (QGIS-Start, Laden von Plugins, Verarbeitungswerkzeuge…).

KAPITEL 4

Das Browser Bedienfeld

- *Ressourcen, die über den Browser geöffnet/ausgeführt werden können*
- *Die obersten Einträge des Brower-Bedienfelds*
  - *Favoriten*
  - *Räumliche Lesezeichen*
  - *Projektverzeichnis*
  - *Laufwerke und Dateisystem*
  - *Datentyp*
  - *Datenbankeinträge*
  - *Kacheln und Internetdienste*
  - *Projekt und unterstützte Dateitypen*

Das *Browser* Bedienfeld eignet sich sehr gut, um Ressourcen zu Suchen, zu Prüfen, zu Kopieren und zu Laden. Es werden nur Ressourcen angezeigt, die von QGIS genutzt werden können.

Sie können hier Daten finden, untersuchen und hinzufügen, weitere Informationen finden Sie auch im Kapitel *Das Browser-Bedienfeld*. Darüber hinaus unterstützt das *Browser* Bedienfeld das Ziehen und Ablegen vieler QGIS-Ressourcen, wie Projektdateien, Python-Skripte, Processing-Skripte und Processing-Modelle.

Python-Skripte, Processing-Skripte und Processing-Modelle können auch zur Bearbeitung in einem externen Editor bzw. dem grafischen Modellierer geöffnet werden.

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

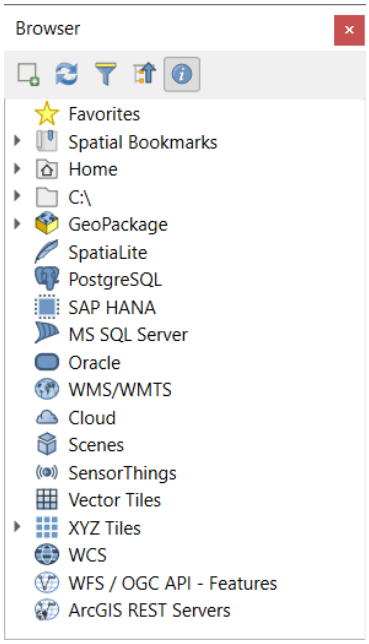


Abb. 4.1: Das Browser Bedienfeld

Das Browser-Bedienfeld (Abb. 4.1) ist als erweiterbarer, hirachischer Ordnerbaum organisiert, wobei es eine Reihe von Einträgen für bestimmte Ressourcen gibt, die immer vorhanden sind. Wenn für einen Eintrag Ressourchen vorhanden sind, wird ein links neben dem Eintrag angezeigt und dieser kann durch klicken darauf erweitert werden. Durch Klicken auf kann ein Zweig wieder eingeklappt werden. Die Schaltfläche Alle einklappen oben im Bedienfeld reduziert alle Einträge auf die oberste Ebene.

Im Menü *Einstellungen* ► *Oberflächenanpassung* ist es möglich, die Anzeige von Ressourcen zu deaktivieren. Wenn Sie z.B. keine Python-Skripte im Browser angezeigt haben möchten, können Sie den Eintrag *Browser* ► *py* deaktivieren oder wenn Ihr Home-Verzeichnis nicht angezeigt werden soll, können Sie den Eintrag *Browser* ► *special:Home* deaktivieren.

Mit der Schaltfläche (Browser filtern) kann ein Names-Filter für die im Browser angezeigten Ressourcen gesetzt werden. Wenn der Filter aktiviert ist, erscheint neben einem Eingabefeld auch eine Schaltfläche mit Optionen - mit dem Pulldown-Menü hier können Sie:

- *Groß-/Kleinschreibung beachten* ein- und ausschalten
- Die Syntax des Filters auf einen der folgenden Werte setzen:
  - *Normal*
  - *Platzhalter*
  - *Regulärer Ausdruck*

Über die Schaltfläche Eigenschaften-Element ein-/ausschalten kann ein Werkzeug aktiviert werden, welches am unteren Rand des Bedienfelds nützliche Informationen über Einträge und Ressourcen anzeigt. Wenn es aktiviert ist (und Informationen verfügbar sind), öffnet es sich wie in Abb. 4.2 gezeigt.

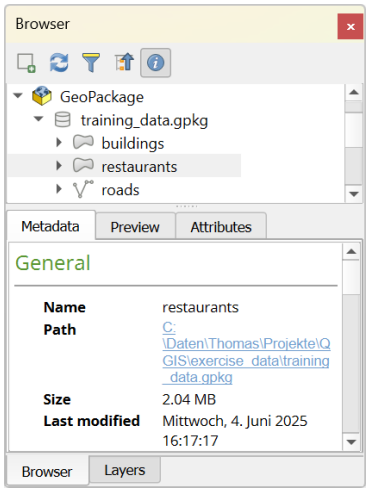


Abb. 4.2: The properties widget

Ein zweites Browser-Bedienfeld kann durch Aktivieren des *Browser (2)* Bedienfelds im Menü *Ansicht* ► *Bedienfelder* geöffnet werden. Ein zweites Bedienfeld kann nützlich sein, wenn Layer zwischen Ressourcen kopiert werden sollen, die sich tief unten in verschiedenen Zweigen der Browser-Hierarchie befinden.

4.1 Ressourcen, die über den Browser geöffnet/ausgeführt werden können

Mit dem *Browser* Bedienfeld können viele Aufgaben bewältigt werden:

- Layer durch verschiedene Methoden zur Kartenansicht hinzufügen: durch Doppelklick, die Layer auf die Kartenansicht ziehen oder, nach Auswahl des Layers, auf die Schaltfläche Gewählte Layer hinzufügen klicken
- Ausführen von Python-Skripten (einschließlich Verarbeitungsalgorithmen) durch Doppelklick oder Ziehen auf die Kartenansicht
- Ausführen von Modellen durch Doppelklick oder Ziehen auf die Kartenansicht
- Über das Kontextmenü aus QGIS-Projektdateien *Symbole extrahieren…*
- Über das Kontextmenü von Dateien diese mit der entsprechenden Standardanwendungen öffnen (Kontextmenü **<file type>* extern öffnen…*). Beispiele: HTML-Dateien, Tabellenkalkulationen, Bilder, PDFs, Textdateien, …
- Einträge kopieren
- Umbenennen und Löschen (mehrerer) Layer (Kontextmenü: *Verwalten* ►)
- Über das Kontextmenü kann mit *In Dateien anzeigen* ein Datei-Explorer-Fenster des entsprechenden Ordners geöffnet werden

Darüber hinaus kann es im Kontextmenü ressourcenspezifische Aktionen geben, z.B. „Projekt öffnen“ oder „Layer exportieren“.

4.2 Die obersten Einträge des Brower-Bedienfelds

4.2.1 Favoriten

Häufig verwendete Speicherorte können über das Kontextmenü als Favoriten gekennzeichnet werden, damit sie dort angezeigt werden.

Ebenfalls über das Kontextmenü kann mit *Favorit umbenennen…* ein Alias vergeben oder mit *Favorit entfernen* ein Eintrag wieder aus den Favoriten gelöscht werden.

4.2.2 Räumliche Lesezeichen

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

Tab. 4.1: available actions for spatial bookmarks

Ebene	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 Projektverzeichnis

Nachdem die Projektdatei zum ersten Mal gespeichert wurde, steht im *Browser*-Bedienfeld der *Projektverzeichnis* Eintrag zur Verfügung. Dieser soll den schnellen Zugriff auf Daten und andere Dateien des Projekts ermöglichen.

Standardmäßig ist dies der Ordner, in dem das Projekt gespeichert wurde, kann aber über die Option *Projekt* ► *Eigenschaften…* ► *Allgemein* ► *Projekt-Stammverzeichnis* oder durch einen Rechtsklick auf den Eintrag *Projektverzeichnis* und Auswahl von *Projektverzeichnis wählen…* geändert werden. Das Anpassen dieses Ordners ist besonders nützlich, wenn QGIS-Projekte nicht im Stammordner eines organisatorischen „Projekts“ zusammen mit Datensätzen gespeichert sind.

4.2.4 Laufwerke und Dateisystem

Standardmäßig werden im *Browser* Bedienfeld auch verschiedene Ordner angezeigt. Welche dies sind, ist vom verwendeten Betreibssystem abhängig.

Diese Ordner sind:

- Unter Windows wird als *Home* das Benutzerverzeichnis des aktuell angemeldeten Benutzers gezeigt
- auf Unix-basierten Systemen wird auf das Root-Verzeichnis `/` verwiesen
- Alle verbundenen Laufwerke, sowohl lokale als auch Netzlaufwerke, werden angezeigt. Je nach Betriebssystem werden sie direkt aufgelistet (z.B. `C:\`, `D:\`) oder über den Eintrag `/Volumes`.

Über das Kontextmenü haben Sie für jeden Ordner (auch Unterordner) und jedes Laufwerk folgende Optionen:

- *Aktualisieren* des Inhalts
- Mit *Neu* ► können Sie ein *Verzeichnis*, ein *GeoPackage* oder ein ESRI *Shapefile* neu erstellen

- Das Verzeichnis *Im Browser verstecken*; dies kann mittels eines Rechtsklicks auf ein anderes Verzeichnis über *Verborgene Elemente* ► wieder sichtbar gemacht werden
- Mit *Farbe setzen…*: können sie die Farbe, in der das Verzeichnis angezeigt wird, anpassen. Dies kann die schnelle Navigation in komplexen Ordnerstrukturen erleichtert
- *Durchsuche* ► aktivieren:
  - *Änderungen verfolgen*: ermöglicht es manuell festzulegen, ob ein bestimmtes Verzeichnis auf Änderungen überwacht und automatisch aktualisiert werden soll oder nicht. Diese Einstellung gilt für das ausgewählte Verzeichnis und alle Unterverzeichnisse. Standardmäßig werden lokale Verzeichnisse automatisch überwacht, Remote- oder Netzlaufwerke nicht. Damit haben sie z.B die Möglichkeit, die Überwachung von Netzlaufwerken zu aktivieren, wenn Sie wissen, dass es keine Probleme gibt, oder die Überwachung großer Verzeichnisse zu deaktivieren, um lange Ladezeiten zu vermeiden.
  - *Verzeichnis schnell durchsuchen*
- *Verzeichnis öffnen…* öffnet das Verzeichnis im Dateibrowser
- *In Terminal öffnen …* öffnet das Verzeichnis in einem Terminalfenster
- mit *Eigenschaften…* können sie spezielle Informationen über das Verzeichnis abrufen, die in QGIS zur Verfügung stehen; mit *Verzeichniseigenschaften…* rufen sie die Informationsseite des Betriebssystems auf

4.2.5 Datentyp

Tab. 4.2: Available actions for plain layer files

Level and Context menu	Arten von Layern Raster (*Reference*)	Vector (*Reference*)	Point clouds (*Reference*)	Mesh (*Reference*)
File				
Lösche <dataset name>	☑	☑		
In Dateien anzeigen	☑	☑	☑	☑
Verwalten	☑	☑	☑	☑
File Properties…	☑	☑	☑	☑
Open <file type> Externally…	☑			
Layer				
Layer zu Projekt hinzufügen	☑	☑	☑	☑
Layer exportieren	☑	☑		
Layer Properties…	☑	☑	☑	☑
Open with Data Source Manager…	☑	☑		
Execute SQL…		☑		

4.2.6 Datenbankeinträge

Abhängig von Ihrem Betriebssystem und den installierten Treibern haben Sie ggf. Zugriff auf verschiedene Datenbanktypen, die Sie in QGIS verwenden können. In der nachfolgenden Tabelle sind die verschiedenen Möglichkeiten im Kontextmenüs für jede Ebene der Datenbankstruktur aufgeführt.

Tab. 4.3: Available actions for database entries

Level and Context menu	Datenbank-Typ					
	GeoPackage ([1])	SpatiaLite	PostgreSQL	SAP HANA	MS SQL Server	Oracle
Top menu						
Eine *Neue Verbindung…* zu einer existierenden Datenbank erstellen	☑	☑	☑	☑	☑	☑
einen neue *Datenbank erzeugen…*	☑	☑				
Datenbank und Layer erzeugen…	☑					
in einer Datei *Verbindungen speichern…*			☑	☑	☑	
aus einer Datei *Verbindungen laden…*			☑	☑	☑	
Connection / Database						
Eine Verbindung *Aktualisieren*			☑	☑	☑	☑
Die Einstellungen einer *Verbindung bearbeiten…*			☑	☑	☑	☑
Verbindung duplizieren			☑	☑	☑	☑
Verbindung löschen…	☑	☑	☑	☑	☑	☑
Import Vector Layer…	☑	☑	☑	☑	☑	
Neues Domänenfeld ► Neuer Domänenbereich…	☑					
Neues Domänenfeld ► Neue Schlüsselwertdomäne…	☑					
Neues Domänenfeld ► Neue Globdomäne…	☑					
<Name_Datenbank> löschen	☑	☑				
Datenbank aufräumen (VACUUM)	☑					
Ein *Neues Schema…* erstellen			☑	☑	☑	
Eine *Neue Tabelle…* erstellen	☑	☑	☑	☑		
SQL ausführen…	☑	☑	☑	☑		
Schema						

Fortsetzung auf der nächsten Seite

Tab. 4.3 – Fortsetzung der vorherigen Seite

Ein Schema *aktualisieren*			✓	✓	✓
Schema-Operationen ► Schema umbenennen…			✓	✓	✓
Schema-Aufgaben ► Schema löschen…			✓	✓	✓
Eine *Neue Tabelle…* erstellen			✓	✓	
SQL ausführen…			✓	✓	
Table / Layer					
Tabellen-Operationen ► Tabelle umbenennen…			✓	✓	✓
Tabellen-Operationen ► Tabelle leeren… (trunkieren)			✓		✓
SQL ausführen…	✓	✓	✓		
Layer exportieren ► in Datei…	✓	✓	✓	✓	✓
Verwalten ► Layer <Name_Layer> umbenennen…	✓	✓			✓
Verwalten ► Layer <Name_Layer> löschen…	✓	✓	✓	✓	✓
Verwalten ► ausgewählte Layer löschen…	✓	✓	✓	✓	✓
Verwalten ► Layer zu Projekt hinzufügen	✓	✓	✓	✓	✓
Verwalten ► ausgewählte Layer zu Projekt hinzufügen	✓	✓	✓	✓	✓
Manage ► Move to Another Schema			✓		✓
Layer Eigenschaften… Dialog öffnen	✓	✓	✓	✓	✓
Datei Eigenschaften… Dialog öffnen	✓				
Mit Datenquellenverwaltung öffnen…	✓	✓			
Fields					
Neues Feld hinzufügen…	✓	✓	✓	✓	
Field					
Felddomäne festlegen…	✓				
Rename Field ([2])	✓		✓		✓
Alias setzen…	✓				
Kommentar setzen…	✓		✓		
Feld löschen…	✓	✓	✓	✓	

[1] Die verschiedenen Einträge können auch für einige von GDAL unterstützte Vektordateiformate wie ESRI File Geodatabase, FlatGeobuf, GeoParquet, NetCDF usw. verfügbar sein, sofern diese kompatibel sind.

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

Bemerkung

Beim Laden von Vektorlayern in QGIS werden Felder mit *Felddomänen* (wie sie beispielsweise in einem GeoPackage oder einer ESRI-Dateigeodatenbank definiert sind) automatisch erkannt. Diese Domänen sind Einschränkungen auf Datenbankebene, d. h. sie werden von der Datenbank selbst erzwungen und gelten für verschiedene Anwendungen, nicht nur für QGIS.

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 *DB Verwaltung*).

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.

Bemerkung

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.

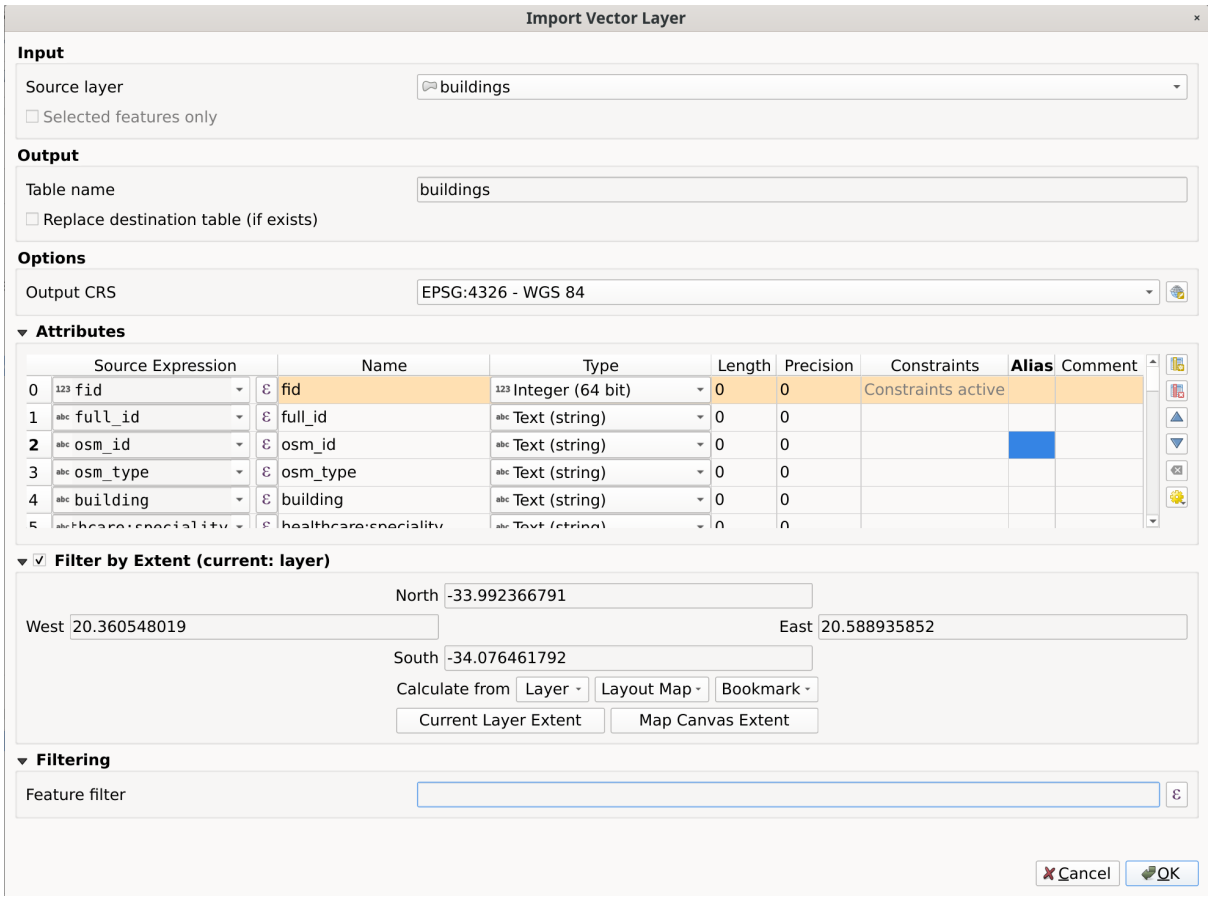


Abb. 4.3: Import Vector Layer dialog

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 Kacheln und Internetdienste

Tab. 4.4: Available actions for web services (1)

Level and Context menu	Art des Dienstes				
	WMS / WMTS (*Referenz*)	XYZ Tiles (*Referenz*)	WCS (*Referenz*)	WFS / OGC API - Features (*Referenz*)	ArcGIS REST Servers (*Referenz*)
Top menu					
Eine *Neue Verbindung…* erstellen	☑	☑	☑	☑	☑

Fortsetzung auf der nächsten Seite

Tab. 4.4 – Fortsetzung der vorherigen Seite

Level and Context menu	Art des Dienstes				
	WMS / WMTS (*Referenz*)	XYZ Tiles (*Referenz*)	WCS (*Referenz*)	WFS / OGC API - Features (*Referenz*)	ArcGIS REST Servers (*Referenz*)
in einer Datei *Verbindungen speichern…*	☑	☑	☑	☑	☑
aus einer Datei *Verbindungen laden…*	☑	☑	☑	☑	☑
Connection					
Verbindung *Aktualisieren*	☑	☑	☑	☑	☑
Die Einstellungen einer *Verbindung bearbeiten…*	☑	☑	☑	☑	☑
Verbindung duplizieren	☑	☑	☑	☑	☑
Verbindung löschen…	☑	☑	☑	☑	☑
Im Browser *Dienst-Informationen anzeigen*					☑
Table / Layer					
Layer exportieren ► in Datei…	☑	☑	☑	☑	☑
Layer zu Projekt hinzufügen	☑	☑	☑	☑	☑
Add Filtered Layer to Project…					☑
Layereigenschaften… Dialog öffnen	☑	☑	☑	☑	☑
Im Browser *Dienst-Informationen anzeigen*					☑

Tab. 4.5: Available actions for web services (2)

Level and Context menu	Art des Dienstes				
	Vector Tiles (*Referenz*)	Scene (*Referenz*)	Cloud (*Referenz*)	SensorThings (*Referenz*)	STAC (*Reference*)
Top menu					
Eine *Neue Verbindung…* erstellen		☑	☑	☑	☑
Eine *Neue allgemeine Verbindung…* erstellen	☑				
Eine *Neue ArcGIS-Vektorkacheldienst-Verbindung…* erstellen	☑				
Save Connections… details to a file	☑	☑	☑	☑	☑
aus einer Datei *Verbindungen laden…*	☑	☑	☑	☑	☑
Connection					

Fortsetzung auf der nächsten Seite

Tab. 4.5 – Fortsetzung der vorherigen Seite

Level and Context menu	Art des Dienstes				
	Vector Tiles (*Referenz*)	Scene (*Referenz*)	Cloud (*Referenz*)	SensorThings (*Referenz*)	STAC (*Reference*)
Die Einstellungen einer *Verbindung bearbeiten…*	☑	☑	☑	☑	☑
Verbindung duplizieren	☑	☑	☑	☑	
Delete Connection…	☑	☑	☑	☑	☑
Refresh					☑
Details…					☑
Table / Layer					
Layer exportieren ► in Datei…				☑	
Layer zu Projekt hinzufügen	☑	☑		☑	
Layereigenschaften… Dialog öffnen	☑	☑		☑	
Refresh					☑
Download assets…					☑
Details…					☑

4.2.8 Projekt und unterstützte Dateitypen

Tab. 4.6: Available actions for various supported files :class: longtable

Datentypen	Kontextmenü
QGIS-Projektdatei (.qgz, .qgs)	*Projekt öffnen*
	Symbole extrahieren… (öffnet die Stilverwaltung, um Symbole als XML zu exportieren, zum Standardstil hinzuzufügen oder als PNG/SVG zu exportieren)
	Dateieigenschaften…
QGIS-Layerdefinitionsdatei (.qlr)	*Layer exportieren ► in Datei…*
	Layer zum Projekt hinzufügen
	Layereigenschaften…
Processing Model (.model3)	*Model ausführen…*
	Model bearbeiten…
Layout-Vorlage (.qpt)	*Neues Layout aus Vorlage*
Python script (.py)	*Script ausführen…*
	In externem Editor öffnen

KAPITEL 5

QGIS anpassen

Sie können QGIS auf vielfache Art und Weise an Ihre Bedürfnisse anpassen. Im Menü *Einstellungen* finden Sie verschiedene Werkzeuge hierfür:

- *Stilverwaltung…*: erstellen und verwalten von *Symbolen, Farbverläufen, etc*.
- *Benutzerprojektionen…*: definieren Sie Ihre eigenen *Koordinatenbezugssystem*.
- *Tastenkürzel…*: legen Sie ihre eigenen <shortcuts> fest. Sie können auch während jeder QGIS-Sitzung durch die *Projekteigenschaften* überschrieben werden (zu finden im Menü *Projekt*).
- *Oberflächenanpassung…*: passen Sie die *Oberfäche* an, indem Sie Dialoge oder Werkzeuge verstecken, die Sie für Ihre Arbeit nicht benötigen.
- *Optionen…*: hier können Sie verschiedene <gui_options> festlegen, die in unterschiedliche Bereiche gegliedert sind. Diese werden in den Einstellungen des aktuellen *Benutzerprofils* gespeichert und sind immer dann wirksam, wenn Sie mit diesem Profil arbeiten.

5.1 Optionen

Einige grundlegende Einstellungen für QGIS können Sie mit dem Dialog *Optionen* vornehmen. Wählen Sie im Menü *Einstellungen* ► *Optionen* aus. Hier können Sie die Einstellungen an Ihre Bedürfnisse anpassen. Einige Änderungen können einen Neustart von QGIS erfordern, um wirksam zu werden.

Im Nachfolgenden werden die einzelnen Reiter beschrieben, in denen Sie die Optionen ihren Bedürfnissen anpassen können.

Bemerkung

Erweiterungen können durch ihre eigenen Einstellungsmöglichkeiten die unten beschriebenen Optionen ergänzen

Bitte beachten Sie, dass im Folgenden nur die Optionen, die in der Kernanwendung vorhanden sind, beschrieben werden. Durch die Installation von *Erweiterungen*, ist es möglich, dass die Standard-Optionen durch Einträge dieser Erweiterung ergänzt werden. Damit wird verhindert, dass jede Erweiterung einen eigenen Konfigurationsdialog mit eigenen Menüeinträgen mitbringt.

5.1.1 Allgemein

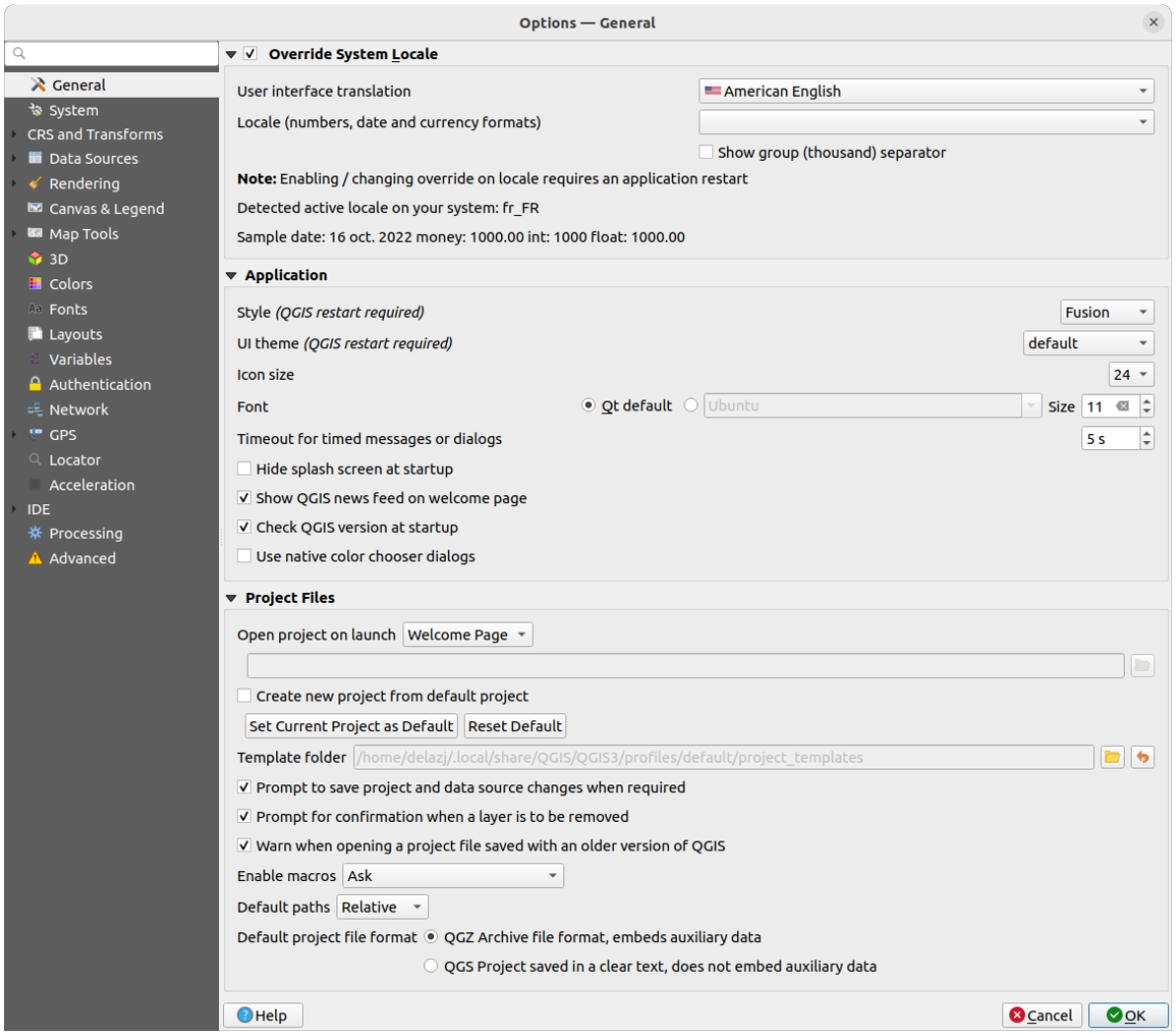


Abb. 5.1: Allgemein

System-Locale übersteuern

Standardmäßig verwendet QGIS die Konfiguration Ihres Betriebssystems für die Sprach- und Zahleinstellungen. Wenn Sie diese Gruppe aktivieren, können Sie dieses Verhalten anpassen:

- Wählen Sie in *Benutzeroberflächenübersetzung* die Sprache aus, die für die Benutzeroberfläche verwendet werden soll
- Wählen Sie in *Locale (Zahlen-, Daten- Währungsformate)* das System, in dem Datum und numerische Werte eingegeben und dargestellt werden sollen
- *Gruppen-/Tausendertrennung anzeigen*

Am unteren Rand des Bereichs wird eine Zusammenfassung der ausgewählten Einstellungen und deren Interpretation angezeigt.

Anwendung

- Wählen Sie den *Stil (QGIS-Neustart erforderlich)*, d.h. das Aussehen und die Anordnung der Elemente in Dialogen. Die Auswahlmöglichkeiten hängen von Ihrem Betriebssystem ab.

- Legen Sie das *Oberflächenthema (QGIS-Neustart erforderlich)* fest . Mögliche Werte sind ‚default', ‚Night Mapping', und ‚Blend of Gray'.
- Legen Sie die *Icon-Größe* fest
- Legen Sie die *Schriftart* und ihre *Größe* fest. Die Auswahl kann *Qt Vorgabe* oder eine benutzerdefinierte Schrift sein.
- Ändern Sie die *Dauer von zeitweiligen Meldungen und Dialogen*
- *Splashscreen beim Start nicht anzeigen*
- *QGIS-Neuigkeiten auf Willkommensseite anzeigen*: zeigt einen QGIS-Newsfeed auf der Startseite an, der Sie über Projektneuigkeiten informiert (Datum und Zusammenfassung von Benutzer-/Entwicklertreffen, Community-Umfragen, Ankündigungen von Releases, verschiedene Tipps…)
- mit *QGIS-Version beim Start überprüfen* werden Sie informiert, wenn eine neue QGIS-Version veröffentlicht wurde
- *Native Farbauswahldialoge verwenden* (siehe *Farbauswahl*)

Projektdateien

- *Projekt beim Start öffnen* bietet die folgenden Möglichkeiten:
  - ‚Willkommensseite' (Standardeinstellung): kann den „Newsfeed" anzeigen, die Projektvorlagen und die zuletzt verwendeten Projekte des aktuellen *Benutzerprofils* (mit Vorschau); standardmäßig wird kein Projekt geöffnet
  - ‚Neu': öffnet ein neues Projekt, das auf der Standardvorlage basiert
  - ‚Zuletzt verwendetes': öffnet das zuletzt verwendete Projekt
  - ‚Bestimmtes': öffnet ein festgelegtes Projekt; benutzen sie die … Schaltfläche, um das Projekt festzulegen
- Unter *Neues Projekt aus Vorgabeprojekt erstellen* können Sie das *Aktuelle Projekt als Vorgabe speichern* oder die *Vorgabe zurücksetzen*. In der Zeile darunter können Sie ein Verzeichnis festlegen, in dem Sie ihre benutzerdefinierten Projektvorlagen speichern. Diese werden dann dem Menü *Projekt* ► *Neu aus Vorlage* hinzugefügt.
- Wählen Sie *Bei Bedarf nachfragen, ob geänderte Projekte und Datenquellen gespeichert werden sollen*, um zu verhindern, dass Ihre Änderungen verloren gehen.
- *Beim Löschen von Layern Bestätigung anfordern*
- *Warnung ausgeben, wenn QGIS-Projekt einer früheren Version geöffnet wird*: Sie können Projekte, die mit einer älteren QGIS-Version gespeichert wurden, jederzeit öffnen; das Öffnen einer neueren Projektdatei mit einer älteren Version kann aber scheitern, weil neuere Entwicklungen in dieser Version noch nicht verfügbar sind.
- *In Projekt eingebetteten Python-Code aktivieren* . Diese Option regelt die Ausführung von Makros, die geschrieben wurden, um eine Aktion bei Projekt-Ereignissen auszuführen, sowie von benutzerdefinierten Python-Funktionen, die als Ausdrücke verwendet werden sollen. Sie können zwischen „Nie", „Fragen", „Nur in diese Sitzung" und „Immer (nicht empfohlen)" wählen.
- *Vorgabepfade*: legt fest, ob Pfade zu Dateien und Layern, die in neuen Projekten verwendet werden, als ‚Absolut' oder ‚Relativ' zur Projektdatei gespeichert werden. Diese Einstellung kann auf Projektebene überschrieben werden.
- *Vorgabeprojektdateiformat*
  - *QGZ-Archivdateiformat, enthält Zusatzdaten* (siehe *Hilfsspeicher*)
  - *QGS-Projekt in Klartext speichern, unterstützt keine Zusatzdaten*: die Zusatzdaten werden in einer separaten `.qgd`-Datei zusammen mit der Projektdatei gespeichert

5.1.2 System

SVG-Pfade

Es können *Suchpfad(e) für SVG-Symbole (Scalable Vevtor Graphic)* hinzugefügt oder entfernt werden. Alle SVG-Dateien, die sich in diesen Verzeichnissen befinden, stehen dann zur Verfügung, um die Objekte ihrer Karte zu symbolisieren, zu beschriften oder um Dekorationen hinzuzufügen (z.B. in der Layer-Symbolisierung oder der Stilverwaltung).

Lesen Sie auch das Kapitel *Entfernte oder eingebettete Dateiauswahl* für verschiedene Möglichkeiten, auf SVG-Dateien in einem QGIS-Pfad zu verweisen.

Erweiterungspfade

Fügen Sie *Nach zusätzlichen C++-Erweiterungsbibliotheken zu durchsuchende Pfad(e)* hinzu oder entfernen Sie diese.

Dokumentationspfade

Hinzufügen oder Entfernen von *Dokumentationspfad(e)* für die QGIS-Hilfe. Standardmäßig wird ein Link zum offiziellen Online-Benutzerhandbuch hinzugefügt, das der verwendeten Version entspricht (d. h. `https://docs.qgis.org/$qgis_short_version/$qgis_locale/docs/user_manual/`). Sie können jedoch weitere Links hinzufügen und diese von oben nach unten priorisieren: Jedes Mal, wenn Sie in einem Dialogfeld auf die Schaltfläche *Hilfe* klicken, wird der oberste Link überprüft. Wenn keine entsprechende Seite gefunden wird, wird der nächste Link versucht usw.

> **Bemerkung**
>
> Die Dokumentation wird nur für die QGIS Long Term Releases (LTR) versioniert und übersetzt. Das bedeutet, dass Sie, wenn Sie in einer regulären Version (z.B. QGIS 3.0) auf die Schaltfläche Hilfe klicken, als Voreinstellung das nächste LTR-Benutzerhandbuch angezeigt bekommen (d.h. 3.4 LTR), das aber Beschreibungen von Programmeigenschaften enthalten kann, die erst in neueren Versionen (3.2 und 3.4) enthalten sind. Falls keine LTR-Dokumentation verfügbar ist, wird die *testing*-Dokumentation angezeigt, die Programmeigenschaften von neueren und Entwicklerversionen enthält.

Einstellungen

Sie können die *Benutzeroberflächeneinstellungen zurücksetzen (Neustart erforderlich)* falls Sie eigene *Anpassungen* vorgenommen haben, diese aber wieder verwerfen wollen.

Umgebung

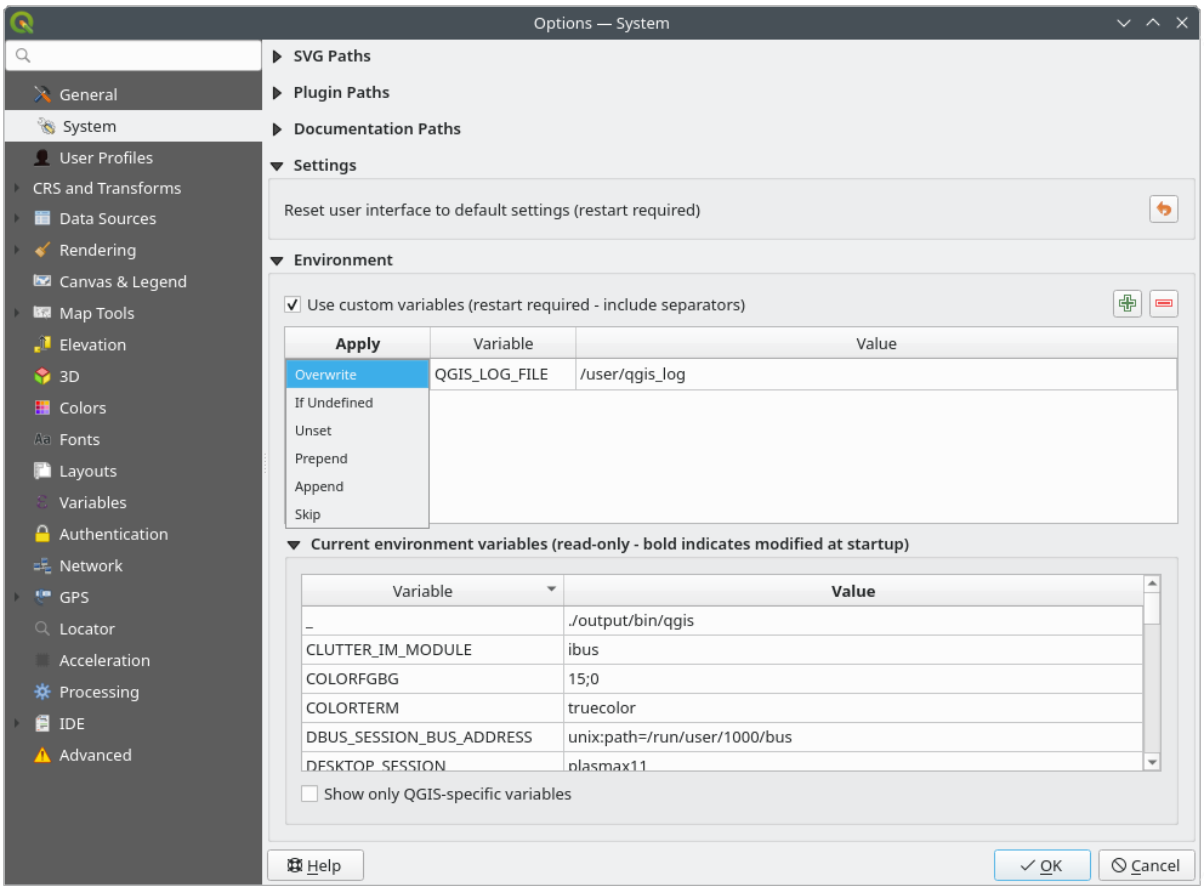


Abb. 5.2: Systemumgebungsvariablen

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.

Wenn Sie das Kontrollkästchen *Benutzerdefinierte Umgebungsvariablen verwenden (Neustart erforderlich - Trennzeichen einschließen)* aktiviert haben, können Sie eine neue Variable hinzufügen oder eine Variable entfernen. Für jedes neue Element können Sie einen *Variable*-Namen, einen *Wert* und die zu verwendende *Anwenden*-Methode konfigurieren, wobei die folgenden Methoden zur Verfügung stehen:

- *Überschreiben*: ersetzt einen bereits existierenden Wert der Variablen
- *wenn definiert*: Verwenden Sie diesen Wert für die Variable, wenn er nicht bereits auf einer höheren Ebene (z. B. auf Betriebssystem- oder Anwendungsebene) definiert ist
- *Nicht gesetzt*: die Variable aus der Umgebung entfernen (der Parameter *Wert* wird nicht verwendet)
- *Voranstellen*: den Wert dem bereits bestehenden Wert der Variablen voranstellen
- *Anhängen*: den Wert dem bereits bestehenden Wert der Variablen anfügen
- *Überspringen*: der Eintrag wird in der Liste für zukünftige Referenzzwecke aufbewahrt, aber nicht benutzt

Bereits definierte Umgebungsvariablen werden in *Aktuelle Umgebungsvariablen* angezeigt, wobei es möglich ist, sie durch Aktivierung der Checkbox *Nur QGIS-spezifische Variablen anzeigen* zu filtern.

5.1.3 Benutzerprofile

Bemerkung

Für weitere Informationen zur Verwaltung von Benutzerprofilen lesen Sie bitte den entsprechenden Abschnitt unter *Arbeiten mit Benutzerprofilen*.

5.1.4 KBS und Transformation

Bemerkung

Für weitere Informationen darüber, wie QGIS mit der Projektion von Layer umgeht, lesen Sie bitte den Abschnitt *Arbeiten mit Projektionen*.

KBS-Behandlung

Auf dem *KBS-Behandlung*-Reiter können Sie festlegen, welches Koordinatenbezugssystem für ein neues Projekt oder einen neuen Layer verwendet werden soll.

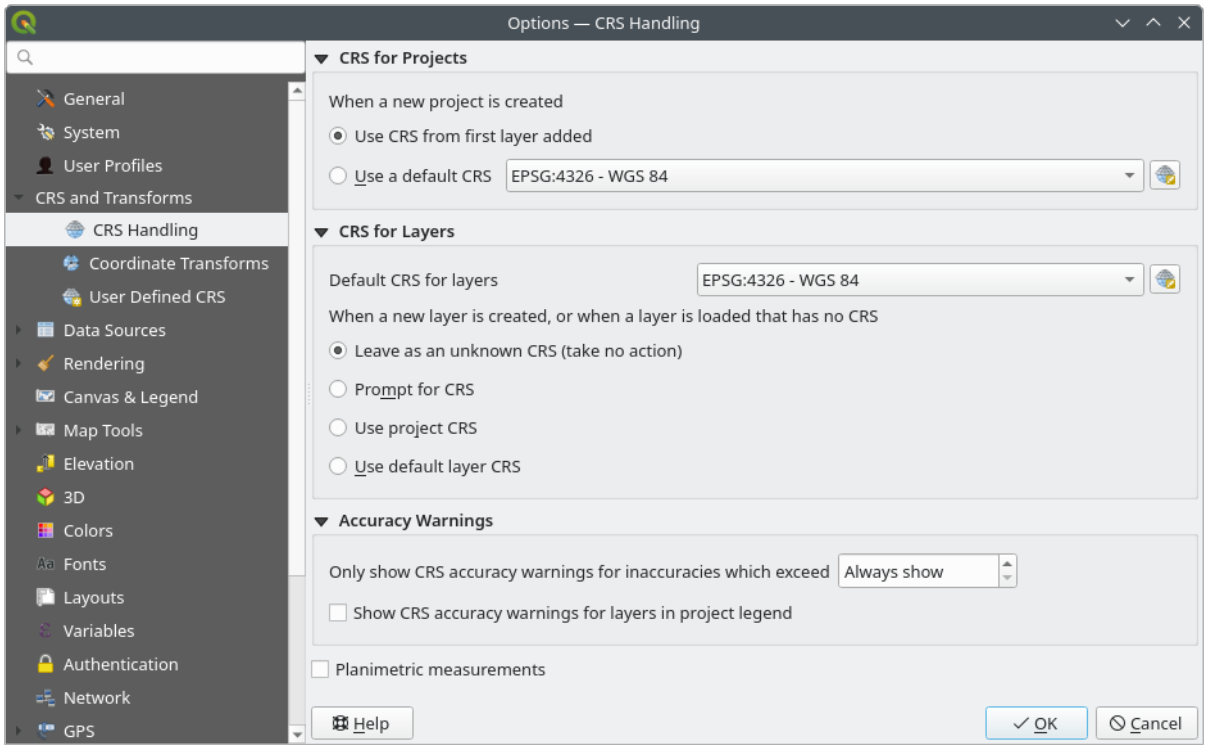


Abb. 5.3: KBS Einstellungen

KBS für Projekte

Um einem neu erzeugten Projekt ein KBS zuzuweisen, gibt es zwei Möglichkeiten:

- *KBS des ersten hinzugefügten Layer verwenden*: das KBS des Projekts wird auf das KBS des ersten geladenen Layers gesetzt
- *Voreingestelltes KBS verwenden*: Ein vorausgewähltes KBS wird standardmäßig auf jedes neue Projekt angewendet und bleibt beim Hinzufügen von Layern zum Projekt unverändert.

Die Auswahl wird für die Verwendung in späteren QGIS-Sitzungen gespeichert. Das Koordinatenbezugssystem des Projekts kann weiterhin über das Menü *Projekt* ► *Eigenschaften…* ► *KBS* angepasst werden.

KBS für Layer

Vorgabe-KBS für Layer: hier können Sie ein Standard-KBS auswählen, das beim Erstellen eines Layers verwendet werden soll (wenn unten ausgewählt)

Sie können die Aktion festlegen, die ausgeführt werden soll, wenn ein neuer Layer erstellt wird oder wenn ein Layer ohne KBS geladen wird:

- *Bei unbekanntem KBS belassen (nichts tun)*
- *KBS abfragen*
- *Projekt KBS verwenden*
- *Projekt KBS verwenden*

Genauigkeitswarnungen

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.

KBS-Genauigkeitswarnung für Layer in Projektlegende anzeigen: Wenn diese Option aktiviert ist, wird für alle Layer mit einem KBS mit Genauigkeitsproblemen (d. h. einem dynamischen KBS ohne verfügbare Koordinatenepoche oder einem KBS, das auf einem Datumsensemble basiert, dessen inhärente Ungenauigkeit den vom Benutzer festgelegten Grenzwert überschreitet) das Warnsymbol im *Layer*-Bedienfeld angezeigt, um darauf hinzuweisen, dass es sich um einen Layer mit geringer Genauigkeit handelt.

Dies wurde für den Einsatz in den Bereichen Ingenieurwesen, BIM, Asset Management und anderen Bereichen entwickelt, in denen Ungenauigkeiten dieser Art potenziell sehr gefährlich oder teuer sind!

Planimetrische Messungen: legt den Standardwert für die Eigenschaft von *planimetrischen Messungen* für neu erstellte Projekte fest.

Koordinatentransformationen

Auf dem *Koordinatentransformationen* Reiter können Sie Koordinatentransformationen und Operationen festlegen, die beim Laden eines Layers in ein Projekt oder bei der Reprojektion eines Layers angewendet werden sollen.

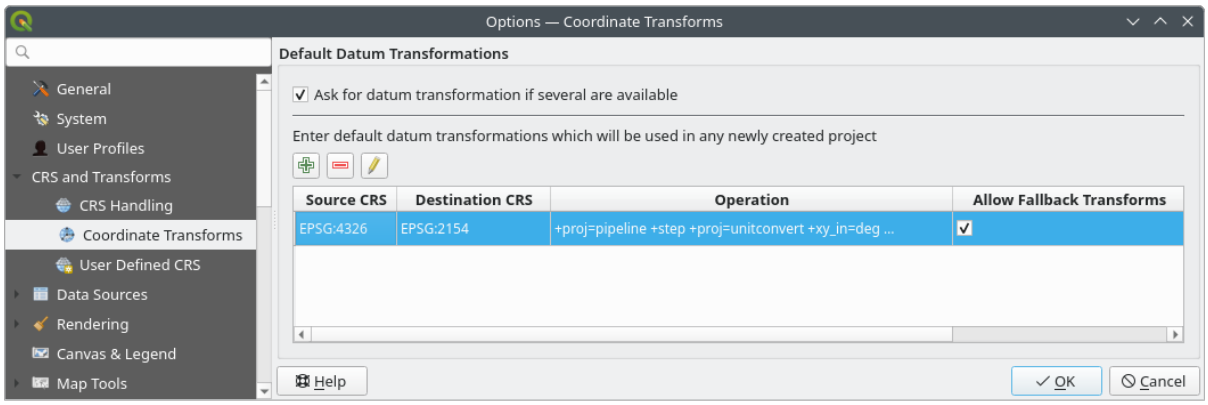


Abb. 5.4: Transformationseinstellungen

Datumstransformationsvorgabe

Hier können Sie einstellen, wie verfahren werden soll, wenn Layer in ein anderes KBS projiziert werden müssen:

- automatisch die QGIS-Standardeinstellungen für Transformationen anwenden
- oder Sie können individuelle Optionen einstellen:
  - *Datumstransformation erfragen, wenn mehrere verfügbar sind*
  - aus einer vordefinierte Liste von Datumstransformationen eine bestimmte auswählen, die standardmäßig angewendet werden soll; für Details dazu siehe *Datumstransformationen*

Sie können Vorgabedatumstransformationen hinzufügen, entfernen oder bearbeiten, die für alle neu erzeugten Projekte verwendet werden sollen.

Benutzerdefinierte KBS

Auf dem Reiter *Benutzerdefinierte KBS* können Sie Ihre eigenen KBS festlegen. Diese müssen dem WKT- oder Proj-Format entsprechen.

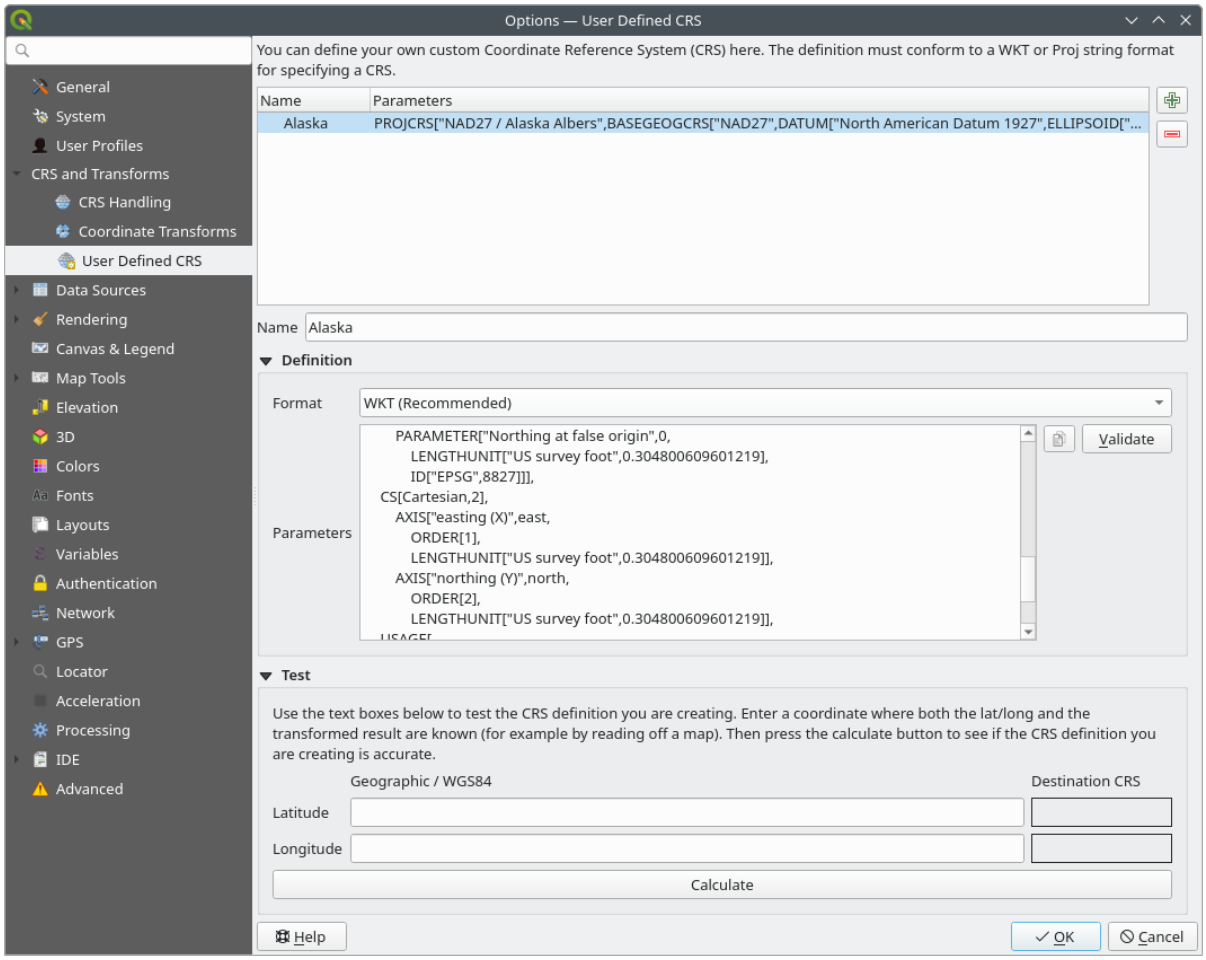


Abb. 5.5: Benutzerdefinierte KBS

Tragen Sie unter der Liste im Textfeld *Name* den gewünschten Namen ein und verwenden Sie die Neues KBS hinzufügen Schalfläche um ein neues KBS anzulegen. Wenn Sie ein vorhandenes KBS löschen möchten, können Sie die KBS entfernen Schlatfläche verwenden.

Definition

- ***Format***
  - WKT (empfohlen)
  - Proj-Zeichenkette (Altlast - nicht empfohlen)
- ***Parameter***
  - Sie können die Parameter eines vorhandenem KBS kopieren
  - mit *Validieren* können Sie testen, ob Ihr Ausdruck korrekt ist

Test

Hier können Sie Ihre erstellte KBS-Definition anhand einer Umrechnung von Breiten- und Längengrad testen. Geben Sie eine bekannte Koordinate, um zu prüfen, ob die Umrechnung das korrekte Ergebnis liefert.

5.1.5 Datenquellen

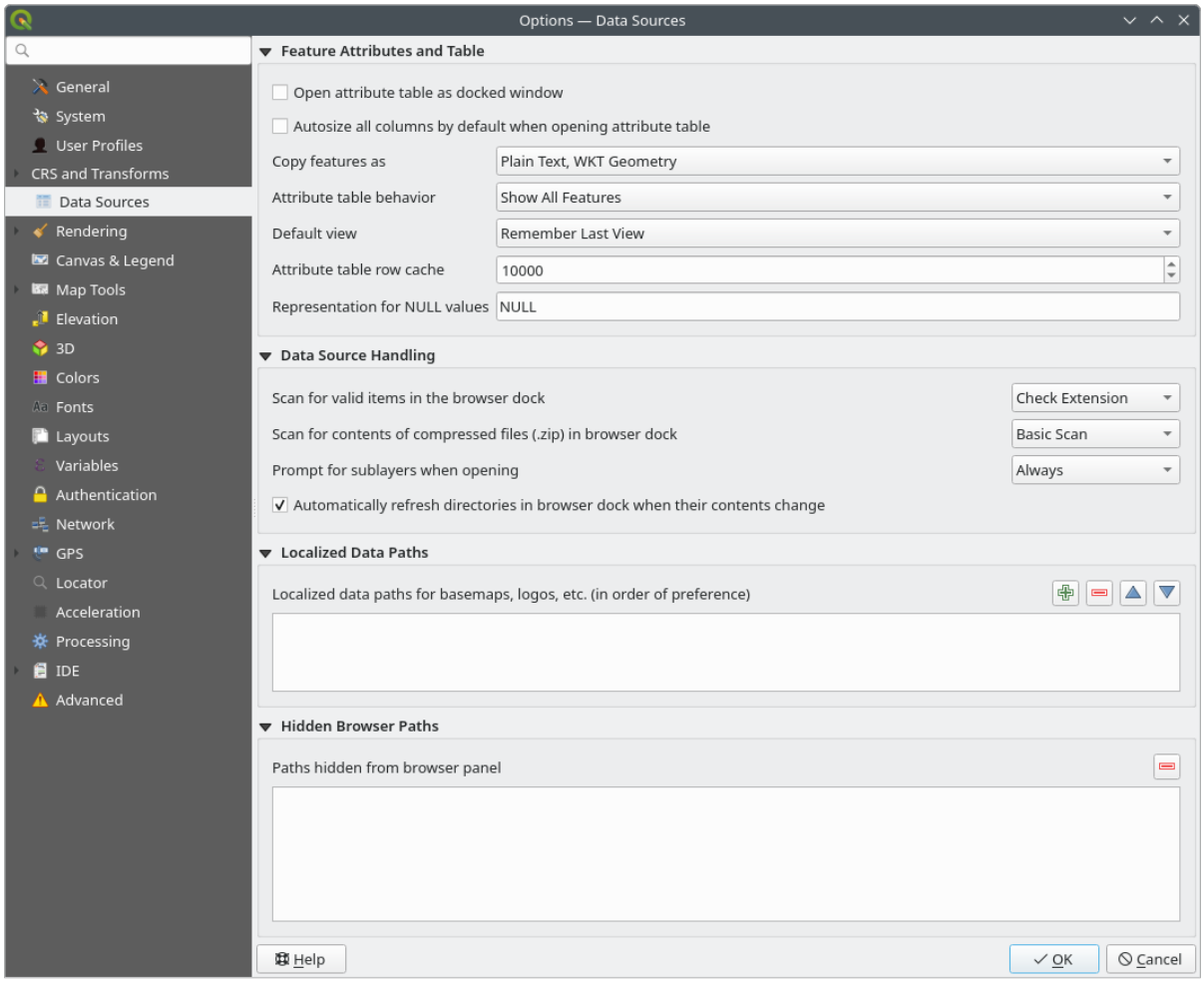


Abb. 5.6: Datenquellen

Objektattribute und -tabelle

- *Attributtabelle im gedockten Fenster öffnen*
- *Beim Öffnen der Attributtabelle die Spaltengröße automatisch anpassen*

- *Objekte kopieren als*: Legt das Format fest, das für Daten verwendet werden soll, wenn Funktionen in andere Anwendungen eingefügt werden.
  - *Einfacher Text, keine Geometrie*: Attribute werden als Text kopiert, aber die Geometrie wird übersprungen
  - *Einfacher Text, WKT-Geometrie*: Attribute werden als Text kopiert und die Geometrie wird in WKT zurückgegeben
  - *Einfacher Text, WKT-Geometrie*: Attribute werden als Text kopiert und die Geometrie wird in WKT zurückgegeben (Hexadezimalformat)
  - *GeoJSON*: Attribute und Geometrie werden als GeoJSON-Daten formatiert
- *Attributtabellenverhalten* : setzt beim Öffnen einen Filter auf die Attributtabelle; es gibt die folgenden Möglichkeiten: ‚Alle Objekte anzeigen', ‚Alle gewählten Objekte anzeigen' and ‚Sichtbare Objekte der Karte anzeigen'.
- *Voreingestellte Ansicht*: definiert den Ansichtsmodus der Attributtabelle beim Öffnen; mögliche Einstellungen sind ‚Letzte Ansicht merken', ‚Tabellenansicht' oder ‚Formularansicht'
- *Attributtabellenzeilencache* : hier wird die Anzahl der Zeilen der zuletzte geladenen Tabelle festgelegt, die im Cache gespeichert werden; der Cache wird gelöscht, wenn die Attributtabelle geschlossen wird
- *Repräsentation für NULL-Werte*: Hier können Sie einen Wert festlegen, der in Datenfeldern angezeigt wird, die einen NULL-Wert enthalten

Tipp

Schnelleres Öffnen von Attributtabellen mit sehr vielen Zeilen

Wenn Sie mit Layern mit einer großen Anzahl von Datensätzen arbeiten, kann das Öffnen der Attributtabelle lange dauern, da für die Anzeige alle Zeilen des Layers abfragt werden. Wenn Sie das *Attributtabellenverhalten* auf **Sichtbare Objekte der Karte anzeigen** setzen, wird QGIS beim Öffnen der Tabelle nur die Features in der aktuellen Kartenansicht abfragen, was ein schnelleres Laden der Daten ermöglicht.

Beachten Sie, dass bei dieser Einstellung die Attributtabelle immer nur Objekte enthält, die in der Kartenansicht sichtbar waren, als die Tabelle geöffnet wurde. Das bedeutet, dass auch nach dem Verschieben des Kartenausschnitts mit **Alle Objekte anzeigen** in der Tabelle keine neuen Objekte angezeigt werden. Sie können jedoch die angezeigten Objekte aktualisieren, indem Sie die Option **Features auf der Karte sichtbar machen** auswählen.

Datenquellenbehandlung

- *Nach gültigen Element im Browser suchen* : hier können Sie zwischen ‚Erweiterung prüfen' und ‚Dateiinhalt prüfen' wählen.
- *Inhalt komprimierter Dateien (.zip) im Browser durchsuchen* : legt fest, wie detailliert die Suche nach Informationen (für das Browser Bedienfeld) innerhalb solcher Dateien durchgeführt wird; die möglichen Einstellungen sind ‚Nein', ‚Grundsuche' and ‚Vollsuche'.
- *Beim Öffnen nach Unterlayern fragen*: einige Raster unterstützen Unterlayer, diese werden in GDAL auch „subdatasets" genannt; ein Beispiel sind netCDF-Dateien. Mit dieser Option können Sie festlegen, wie mit Unterlayern verfahren werden soll, wenn eine Datei mit Unterlayern geöffnet wird. Sie haben die folgenden Auswahlmöglichkeiten:
  - ‚Immer': Immer fragen, ob es Unterlayer gibt
  - 'Wenn nötig': Fragen, wenn der Layer keine Bänder, aber möglicherweise Unterlayer hat
  - ‚Nie': Nicht fragen; es wird dann nichts geladen
  - ‚Alle laden': Nicht fragen, aber immer alle Unterlayer laden

- *Verzeichnisse im Browser-Dock automatisch aktualisieren, wenn sich der Inhalt ändert*: Ermöglicht es Ihnen, die Überwachung von Verzeichnissen im *Browser* Bedienfeld zu deaktivieren (z. B. um eine mögliche Verlangsamung aufgrund von Netzwerklatenz zu vermeiden).

Lokalisierungsdatenpfade

Es ist möglich, lokalisierte Pfade für alle Arten von dateibasierten Datenquellen zu verwenden. Dabei handelt es sich um eine Liste von Pfaden, die zur Abstraktion des Speicherorts der Datenquelle verwendet werden. Wenn z. B. `C:\my_maps` in den lokalisierten Pfaden aufgeführt ist, wird ein Layer mit der Quelle `C:\my_maps\my_country\ortho.tif` im Projekt unter `localized:my_country\ortho.tif` gespeichert.

Die Pfade sind in der Reihenfolge ihrer Präferenz aufgelistet, d.h. QGIS sucht die Datei zuerst im ersten Pfad, dann im zweiten usw.

Ausgeblendeter Browserpfad

Hier werden alle Ordner aufgelistet, die Sie aus dem *Browser* Bedienfeld über das Kontextmenü ausgeblendet haben. Wenn Sie einen Ordner aus der Liste entfernen, wird er im *Browser* Bedienfeld wieder sichtbar.

GDAL

GDAL ist eine Datenaustauschbibliothek für Geodaten, die eine Vielzahl von Vektor- und Rasterformaten unterstützt. Sie stellt Treiber zum Lesen und (oft auch) Schreiben von Daten in diesen Formaten bereit. Die Registerkarte *GDAL* zeigt die Treiber für Raster- und Vektorformate mit ihren Funktionen an.

GDAL Raster- und Vektortreiber

Die Reiter *Rastertreiber* und *Vektortreiber* ermöglichen es Ihnen, festzulegen, welcher GDAL-Treiber zum Lesen und/oder Schreiben von Dateien verwendet werden soll, wenn mehr als ein GDAL-Treiber verfügbar ist.

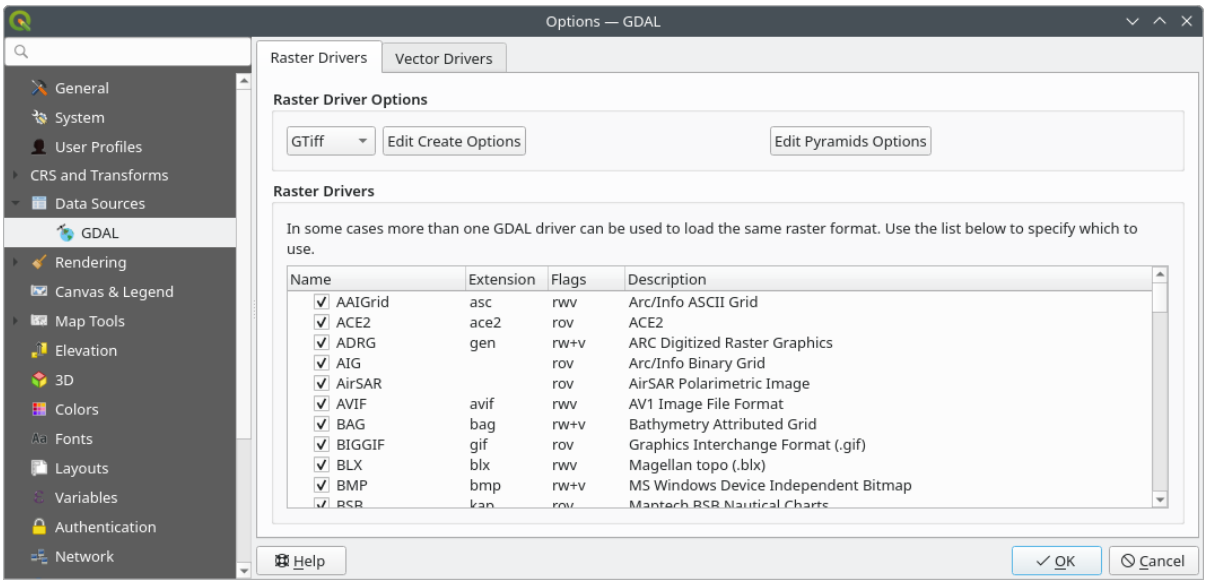


Abb. 5.7: GDAL - Rastertreiber

Tipp

Mit einem Doppelklick auf einen Rastertreiber für den Lese- und Schreibrechte bestehen (`rw+(v)`) öffnen Sie den *Erzeugungsoptionen* Dialog zur Anpassung.

Rastertreiberoptionen

Dieser Bereich bietet Möglichkeiten zur Anpassung des Verhaltens von Rastertreibern, auf die Lese- und Schreibzugriff besteht:

- *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.

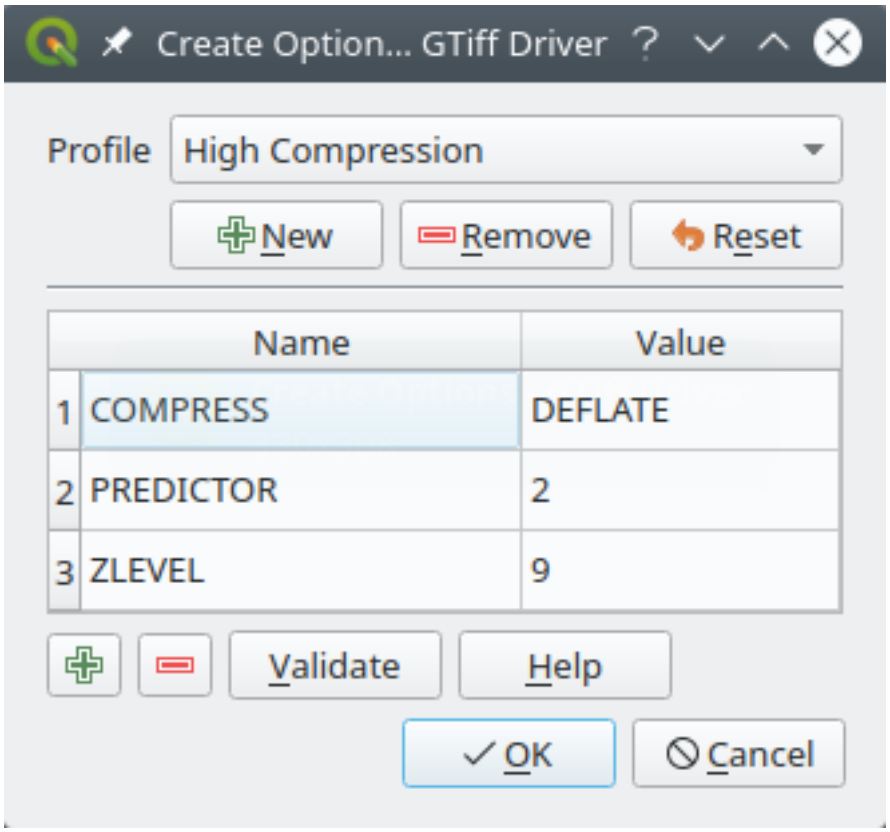


Abb. 5.8: Beispiel für die Erstellung eines Optionsprofils (für GeoTiff)

Im oberen Teil des Dialogfensters werden die aktuellen Profile aufgelistet, und Sie können neue Profile hinzufügen oder eines entfernen. Sie können das Profil auch auf seine Standardparameter zurücksetzen, wenn Sie diese geändert haben. Einige Treiber (z. B. GeoTiff) haben einige Beispielprofile, mit denen Sie arbeiten können.

Im unteren Bereich des Dialofensters finden Sie folgende Optionen:

  - Mit der Schaltfläche können Sie neue Zeilen einfügen, in die Sie den Parameternamen und den Wert eintragen
  - Mit der Schaltfläche können Sie den ausgewählten Parameter löschen
  - Mit der *Prüfen* Schaltfläche können Sie prüfen, ob die für das angegebene Format eingegebenen Erstellungsoptionen gültig sind
  - Verwenden Sie die Schaltfläche *Hilfe*, um die zu verwendenden Parameter zu finden, oder lesen Sie die GDAL-Rastertreiber-Dokumentation.

- *Pyramiden-Optionen bearbeiten*

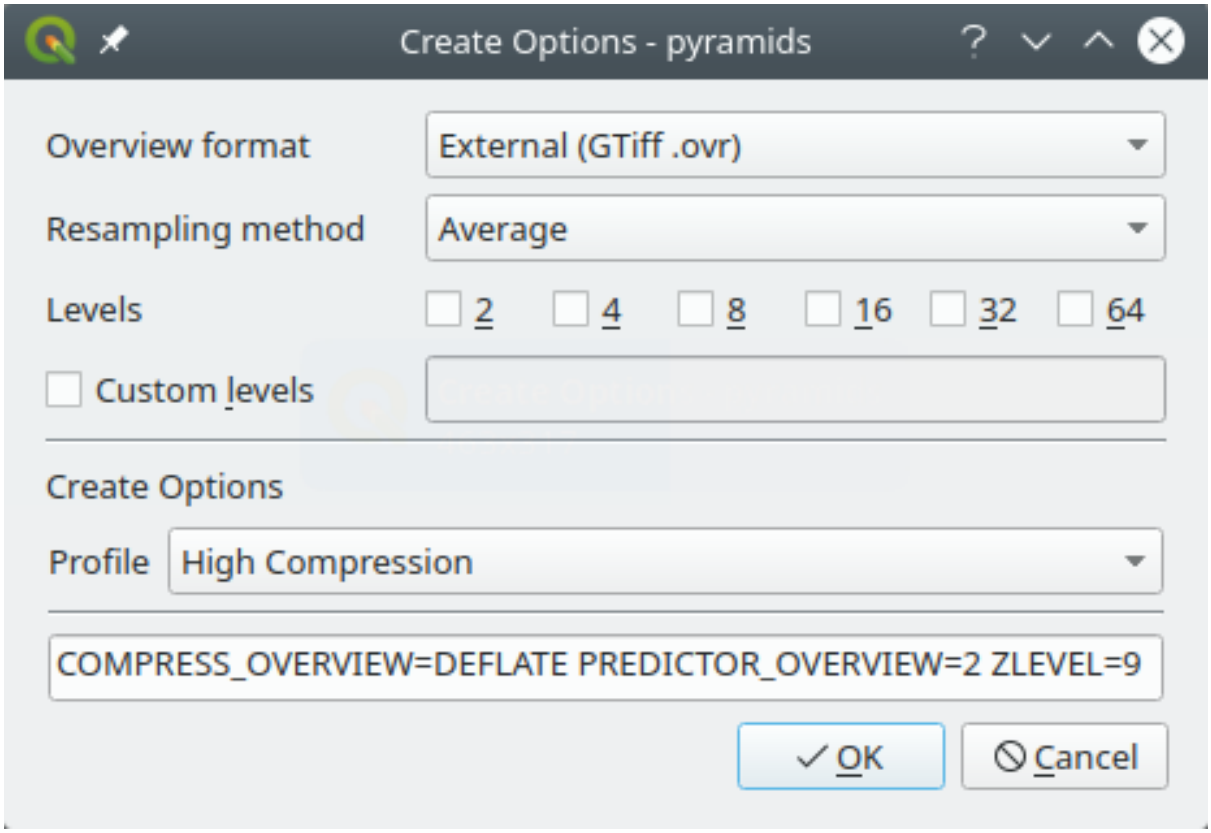


Abb. 5.9: Beispiel eines Pyramidenprofils

5.1.6 Darstellung

Der Reiter *Darstellung* bietet Einstellungen zur Steuerung der Darstellung von Layern in der Kartenansicht.

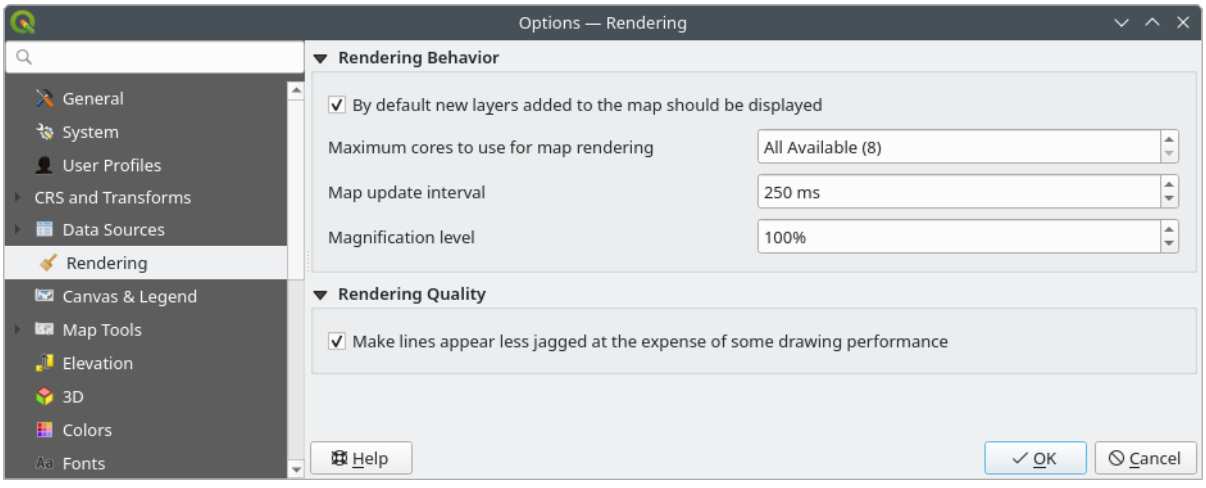


Abb. 5.10: Übersicht des Darstellungs-Reiters

Zeichenverhalten

- *Normalerweise werden alle neuen Layer im Kartenfenster angezeigt*: Die Deaktivierung dieser Option kann beim Laden mehrerer Layer nützlich sein, um zu vermeiden, dass jeder neue Layer in der Kartenansicht

gerendert wird und so das Laden verlangsamt wird

- Legen Sie die Anzahl der *Maximal zu benutzenden Kerne* fest
- Die Kartenansicht wird im Hintergrund in einem separaten Image gerendert, und bei jedem *Kartenaktualisierungsintervall* (Standardwert 250 ms) wird der Inhalt dieses (nicht auf dem Bildschirm angezeigten) Images zur Aktualisierung der sichtbaren Kartenansicht verwendet. Wenn das Rendering der Kartenansicht jedoch schneller als diese Zeitspanne abgeschlossen ist, wird es sofort angezeigt.
- *Vergrößerungsgrad* (siehe Abschnitt: *Vergrößerung*)

Zeichenqualität

- *Linie auf Kosten der Zeichengeschwindigkeit weniger gezackt zeichnen*

Einstellungen zur Vektor-Darstellung

Der Reiter *Vektor* enthält spezifische Einstellungen für die Darstellung von Vektor-Layern.

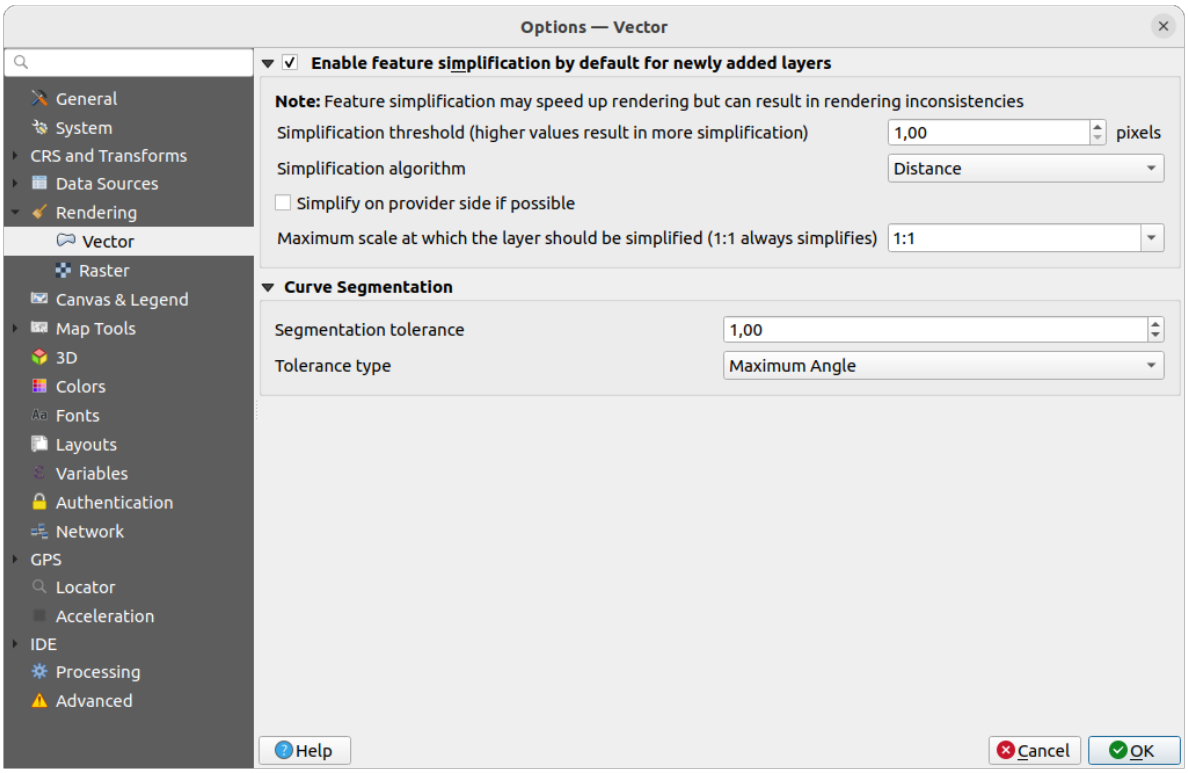


Abb. 5.11: Einstellungen zur Vektor-Darstellung

- *Geometrievereinfachung für neue Layer voreinstellen*: Sie können die Geometrie von Objekten in Vektorlayern vereinfachen (weniger Knoten), was zu einer schnelleren Darstellung führt. Beachten Sie jedoch, dass dies zu Inkonsistenzen bei der Darstellung führen kann. Verfügbare Einstellungen sind:
  - *Vereinfachungsschwelle (höhere Werte führen zu stärkerer Vereinfachung)*
  - *Vereinfachungsalgorithmus*: Diese Option führt eine lokale „On-the-Fly“-Vereinfachung der Features durch und beschleunigt das Rendering der Geometrien. Die von den Datenanbietern abgerufenen Geometrien werden dabei nicht verändert. Dies ist wichtig, wenn Sie Ausdrücke haben, die die Feature-Geometrie verwenden (z. B. Flächenberechnung) – es stellt sicher, dass diese Berechnungen auf der ursprünglichen Geometrie und nicht auf der vereinfachten Geometrie durchgeführt werden. Zu diesem Zweck bietet QGIS drei Algorithmen: „Entfernung“ (Standard), „SnapToGrid“ und „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
  - *Größter Maßstab bis zu dem der Layer vereinfacht werden soll (1:1 vereinfacht immer)*

> **Bemerkung**
>
> Neben der globalen Einstellung kann die Vereinfachung von Objekten für jeden einzelnen Layer über *Layereigenschaften* ► *Darstellung* eingestellt werden.

- *Kurvensegmentierung*
  - *Segmentierungstoleranz*: Diese Einstellung steuert die Art und Weise, wie Kreisbögen gerendert werden. **Je kleiner** der maximale Winkel (zwischen den beiden aufeinanderfolgenden Scheitelpunkten und dem Kurvenzentrum in Grad) oder die maximale Differenz (Abstand zwischen dem Segment der beiden Scheitelpunkte und der Kurvenlinie in Karteneinheiten), desto **mehr geradlinige** Segmente werden beim Rendern verwendet.
  - *Toleranztyp*: kann ‚Maximalwinkel' oder ‚Maximaldifferenz' zwischen Annäherung und Kurve sein.

Raster Einstellungen

Der Reiter *Raster* enthält spezifische Einstellungen für die Darstellung von Raster-Layern.

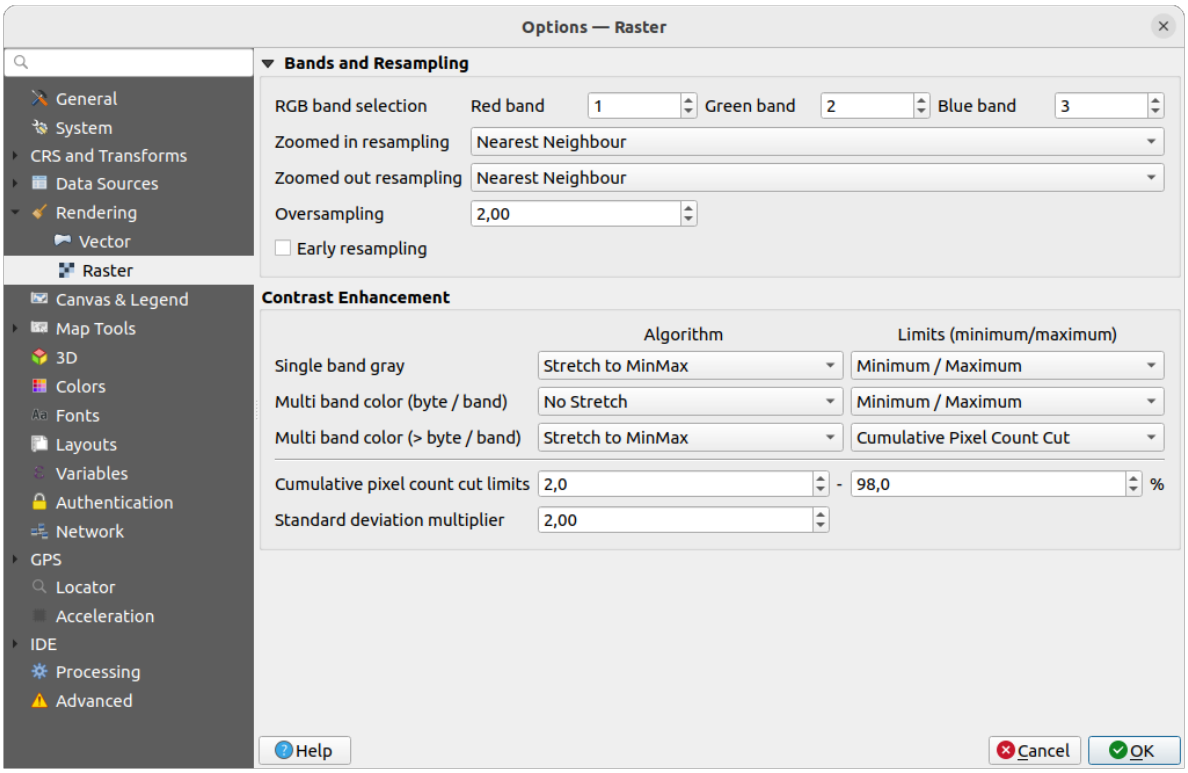


Abb. 5.12: Raster Einstellungen

Bereich *Kanäle und Abtastung*:

- Mit *RGB Kanalauswahl* können Sie die Nummer für den Roten, Grünen und Blauen Kanal festlegen.
- Bei der *Abtastung für Hineinzoomen* und der *Abtastung für Hinauszoomen* können verschiedene Methoden definiert werden; für *Abtastung für Hineinzoomen* können Sie zwischen drei Resampling-Methoden wählen:

‚Nächster Nachbar', ‚Bilinear (2x2 Kern)' und ‚Kubisch (4x4 Kern)'. Für *Abtastung für Hineinzoomen* können Sie zwischen ‚Nächster Nachbar' und ‚Durchschnitt' wählen. Sie können auch einen Wert für die *Überabtastung* einstellen: zwischen 0,0 und 99,99 - je höher der Wert, umso länger dauert die Abtastung - der Standardwert ist 2,0.

- *Frühe Abtastung*: ermöglicht die Berechnung des Raster-Renderings auf der Seite des Datenanbieters (Datenquelle), wo die Auflösung der Quelle bekannt ist, und sorgt für ein besseres Zoomen beim Rendering mit QGIS benutzerdefiniertem Stil. Dies ist sehr praktisch für Vektor Tiles, die mit einer *Interpretationsmethode* geladen werden. Die Option kann auch auf Layer-Ebene gesetzt werden (*Symbolisierung* Eigenschaften).

Kontrastverbesserung kann für *Einkanalgraustufen*, *Multikanalfarbe (Byte/Kanal)* oder *Multikanalfarbe (> Byte/Kanal)* angewendet werden. Für jeden Punkt können Sie folgendes festlegen:

- den zu verwendenden *Algorithmus*, welcher die Werte ‚Kein Strecken', ‚Strecken auf MinMax', ‚Auf MinMax strecken und zuschneiden' oder ‚Zuschneiden auf MinMax' haben kann
- die *Grenzen (Minimum/Maximum)* festlegen, welche die Werte ‚Kumulativer Pixelanzahl-Schnitt', ‚Minimum/Maximum' oder ‚Mittlere +/- Standardabweichung' haben kann.

Die *Kontrastverbesserung* enthält darüber hinaus folgende Optionen:

- *Kommulative Pixelanzahl-Schnittgrenzen*
- *Standardabweichungsfaktor*

5.1.7 Karte & Legende

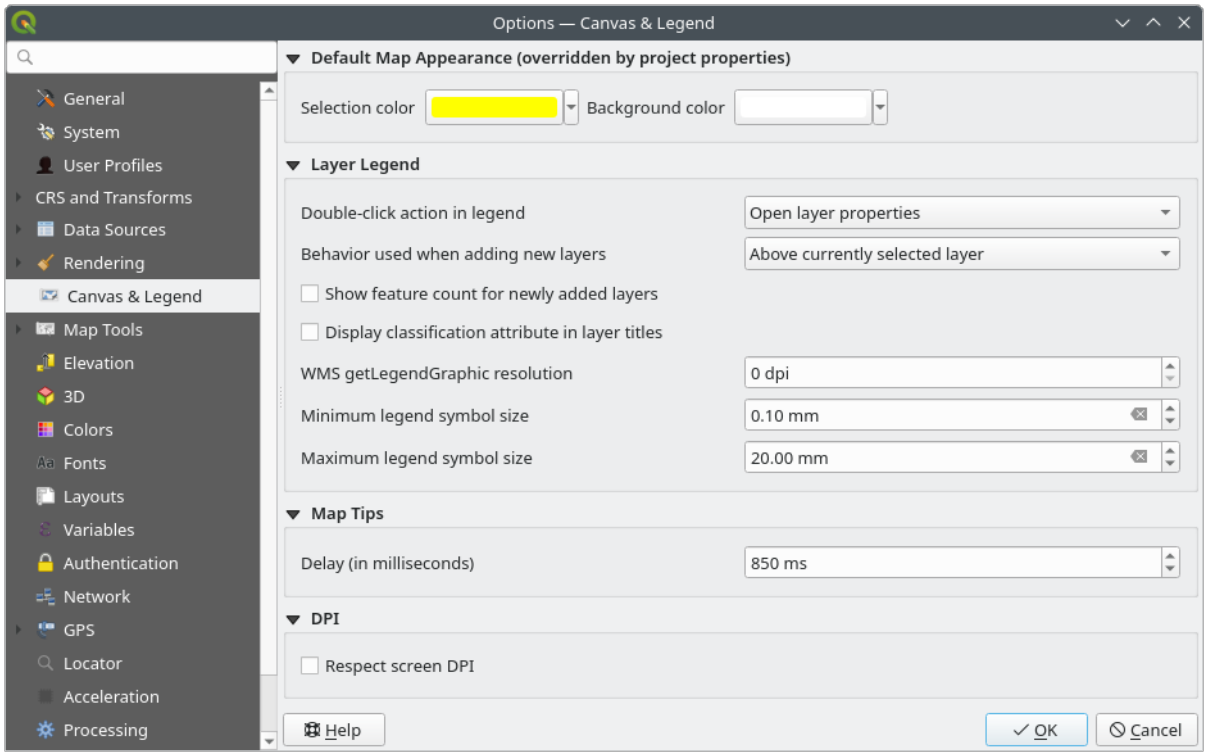


Abb. 5.13: Optionien auf dem Karte und Legende Reiter

Hier haben Sie die folgenden Einstellungsmöglichkeiten:

- unter **Voreingestelltes Kartenaussehen (Projekteigenschaften überschreiben)**: können Sie die *Auswahlfarbe* und die *Hintergrundfarbe* die verwendet werden sollen einstellen.
- Anpassungsmöglichkeiten für die **Layerlegende** im Layer-Bedienfeld:

  - *Doppelklick-Aktion in Legende*: hier kann eingestellt werden, ob ein Doppelklick auf einen Layer entweder die *Layereigenschaften öffnen* (default), die *Attributtabelle öffnen* oder das *Kartengestaltungsdock öffnen* soll.
  - *Beim Anlegen von neuen Layern verwendetes Verhalten*: bestimmt, wo der Layer im *Layer* Bedienfeld platziert wird, wenn er in QGIS geladen wird. Das kann sein:
    * *Über dem derzeitig gewählten Layer*
    * *Immer oben im Layerbaum*
    * *Optimaler Index in der aktuellen Layerbaumgruppe*: Im Gegensatz zu den anderen Optionen, bei denen die neuen Layer direkt untereinander an der gewünschten Stelle platziert werden, wird bei dieser Option die Sortierlogik auf den gesamten Layerbaum (bzw. die aktive Gruppe) ausgedehnt und neue Layer werden „optimal“ eingefügt, d.h. Punktlayer werden auf Punktlayer, Linienlayer auf Linienlayer, Polygonlayer auf Polygonlayer usw. platziert.
  - ☐ *Objektanzahl für neu hinzugefügte Vektorlayer anzeigen*: zeigt im *Layer* Bedienfeld die Anzahl der Objekte neben dem Layernamen an. Die Anzahl der Objekte in den Klassen (wenn vorhanden), wird ebenfalls angezeigt. Sie können mit der rechten Maustaste auf einen Layer klicken, um die Anzahl der Objekte ein- oder auszuschalten.
  - ☐ *Klassifikationssattribute in Layertiteln anzeigen* im Layer-Bedienfeld, z. B. beim Anwenden eines kategorisierten oder regelbasierten Renderers (weitere Informationen finden Sie unter *Symbolisierung*).
  - die *WMS-getLegendGraphic-Auflösung*
  - mit *Minimale Legendensymbolgröße* und *Maximale Legendensymbolgröße* können Sie die Größe, mit der Symbole im ,:guilabel:*Layer* Bedienfeld angezeigt werden, einstellen
- die *Verzögerung (in Millisekunden)* mit der *Kartenhinweise* eines Layers angezeigt werden
- ☑ *Bildschirmauflösung berücksichtigen*: Wenn diese Option aktiviert ist, versucht QGIS, die Kartenansicht mit physikalisch genauer Skalierung auf dem Bildschirm anzuzeigen, abhängig von der physikalischen DPI des Monitors. Symbole mit einer bestimmten Anzeigegröße werden ebenfalls genau dargestellt, z. B. wird ein 10-mm-Symbol auf dem Bildschirm als 10 mm angezeigt. Die Schriftgrößen der Beschriftungen in der Kartenansicht können jedoch von denen in der Benutzeroberfläche von QGIS oder anderen Anwendungen abweichen. Wenn diese Einstellung deaktiviert ist, verwendet QGIS die logische DPI des Betriebssystems, die mit anderen Anwendungen auf dem System übereinstimmt. Der Maßstab der Kartenansicht und die Größe der Symbole können jedoch auf dem Bildschirm physikalisch ungenau sein. Insbesondere auf Bildschirmen mit hoher DPI-Auflösung werden Symbole wahrscheinlich zu klein angezeigt.

  Es wird empfohlen, die Option ☑ *Bildschirmauflösung berücksichtigen* zu aktivieren, wenn Sie mehrere oder unterschiedliche Monitore verwenden und wenn Sie optisch hochwertige Kartenausdrucke erstellen wollen. Die Deaktivierung von ☑ *Bildschirmauflösung berücksichtigen* erzeugt eine Ausgabe, die in der Regel besser für Karten geeignet ist, die nur auf dem Bildschirm verwendet werden sollen, insbesondere wenn die Schriftgrößen anderen Anwendungen entsprechen sollen.

Bemerkung

Die Darstellung in Drucklayouts wird von der Einstellung *Bildschirmauflösung berücksichtigen* nicht beeinflusst; hier wird immer die angegebenen Auflösung verwendet. Beachten Sie auch, dass diese Einstellung die vom Betriebssystem gemeldete physische Bildschirm-Auflösung verwendet, der möglicherweise nicht bei allen Bildschirme korrekt ist.

5.1.8 Kartenwerkzeuge

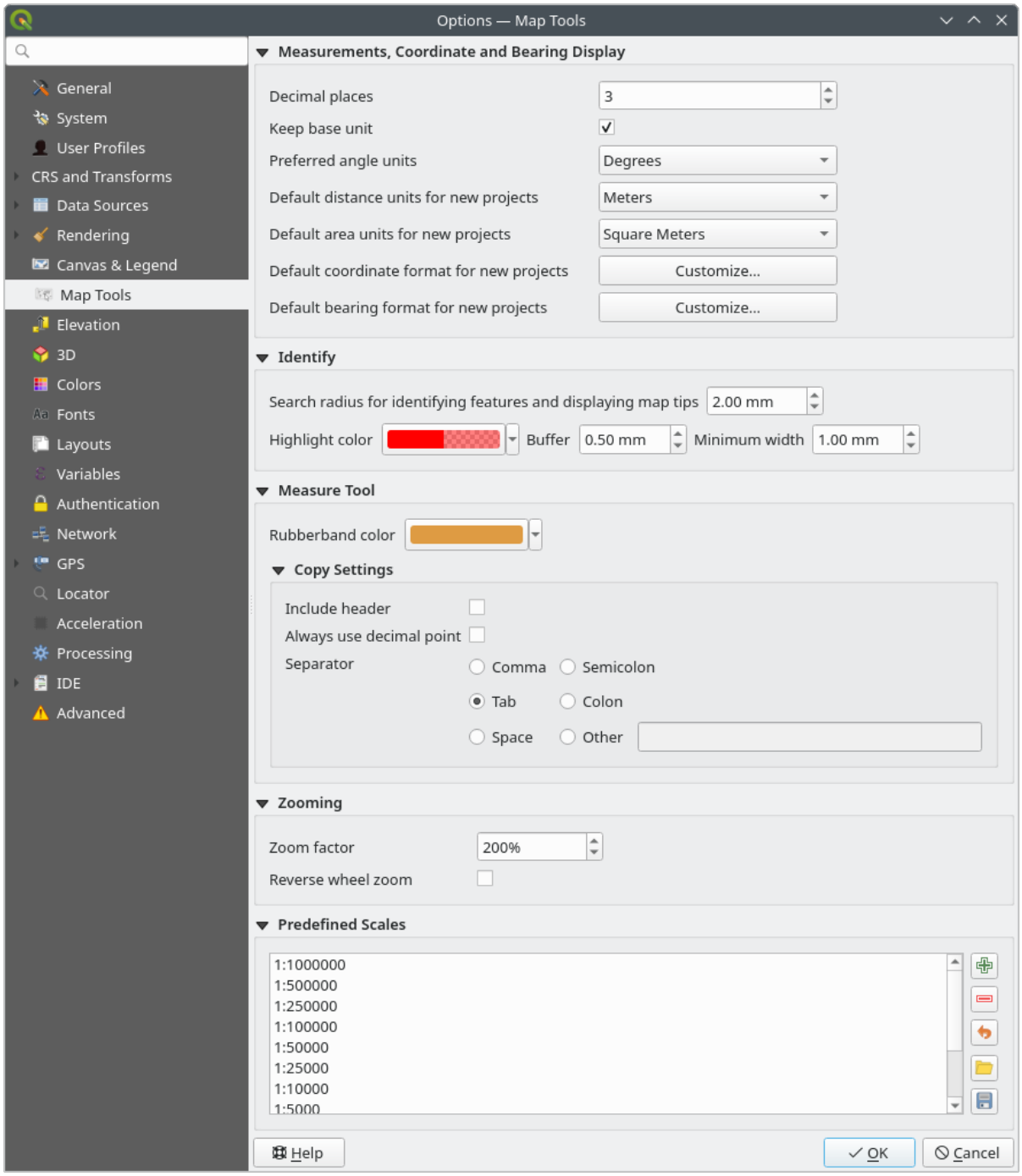


Abb. 5.14: Einstellungen Kartenwerkzeuge

Measurements, Coordinate and Bearing Display

This section provides ways to configure default units parameters:

- Definieren Sie die Anzahl der *Dezimalstellen*, die beim Messen angezeigt werden.
- Wenn *Basiseinheit beibehalten* aktiviert ist, werden große Zahlen nicht automatisch umgewandelt (z. B. Meter in Kilometer).

- Für das *Bevorzugte Winkelmaß* sind die folgenden Optionen verfügbar: ‚Grad', ‚Bogenmaß', ‚Gon/Neugrad', ‚Bogenminuten', ‚Bogensekunden', ‚Umdrehungen', ‚Milliradianten (SI Definition)' und ‚Mil (NATO/Militärische Definition)'.
- *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)
- *Vorgabekoordinatenformat für neue Projekte*, wie es im Koordinatenfeld der QGIS-Statusleiste und im Bereich *Koordinaten* des Objekte abfragen Werkzeugs angezeigt wird.
- *Kursformat für neue Projekte*, wie es in der Richtungsanzeige der Statusleiste und vom Kurs messen Werkzeug angezeigt wird.

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*.

- Mit *Suchradius für die Objektidentifikation und zur Anzeige von Kartenhinweisen* kann der Abstand eingestellt werden, innerhalb dessen das Identifizierungswerkzeug Ergebnisse anzeigt.
- Mit der *Hervorhebungsfarbe* kann eingestellt werden, wie abgefragte Objekte angezeigt werden sollen.
- *Puffer* legt die Größe des Puffers fest, mit der die Hervorhebung dargestellt wird.
- *Minimalbreite* legt fest, wie dick die Umrisslinie eines hervorgehobenen Objektes sein soll.

Messwerkzeug

- Definieren Sie die *Gummibandfarbe* für das Messwerkzeug.

Messwerkzeugkopiereinstellungen

Mit diesen Einstellungen können Sie das Verhalten der Schaltfläche *Kopieren* im Fenster Linie messen steuern. Sie können *Kopf einschließen* auswählen, um die Spaltennamen beizubehalten, und Sie können auch ein Trennzeichen Ihrer Wahl verwenden. Sie können auch das Kontrollkästchen *Immer Dezimalpunkt verwenden* aktivieren, um Ihre in die Zwischenablage kopierten Koordinaten mit einem Punkt als Dezimaltrennzeichen beizubehalten, selbst wenn Ihre Spracheinstellungen in den QGIS-Optionen auf die Verwendung eines Kommas eingestellt sind. Wenn Sie in diesem Fall die Option *Immer Dezimalpunkt verwenden* nicht aktiviert haben, können Sie kein Komma als Feldtrennzeichen auswählen.

Zoomen

- Sie können einen *Zoomfaktor* für Zoom-Werkzeuge und Mausrad festlegen.
- *Mausradzoom umkehren* ermöglicht es Ihnen, die Scrollrichtung des Mausrades anzupassen.

Vordefinierte Maßstäbe

Hier finden Sie eine Liste vordefinierter Maßstäbe, die standardmäßig in den maßstabsbezogenen Dropdown-Menüs angezeigt werden (z. B. in der *Maßstab* Statusleiste, in der Auswahl der Sichtbarkeitsmaßstäbe oder in den Einstellungen für weitere 2D-Kartenansichten). Mit den Schaltflächen und können Sie Ihre eigenen Maßstäbe hinzufügen oder entfernen. Sie können die Liste auch in eine `XML`-Datei exportieren oder daraus importieren. Auf der rechten Seite steht eine Schaltfläche zur Verfügung, um die Liste wieder auf die Standardeinträge zurückzusetzen.

Im Dialogfeld der Projekteigenschaften können Sie auch eine eigene Liste von Maßstäben festlegen, die diese globale Liste in den Dropdown-Menüs überschreibt.

Digitalisieren

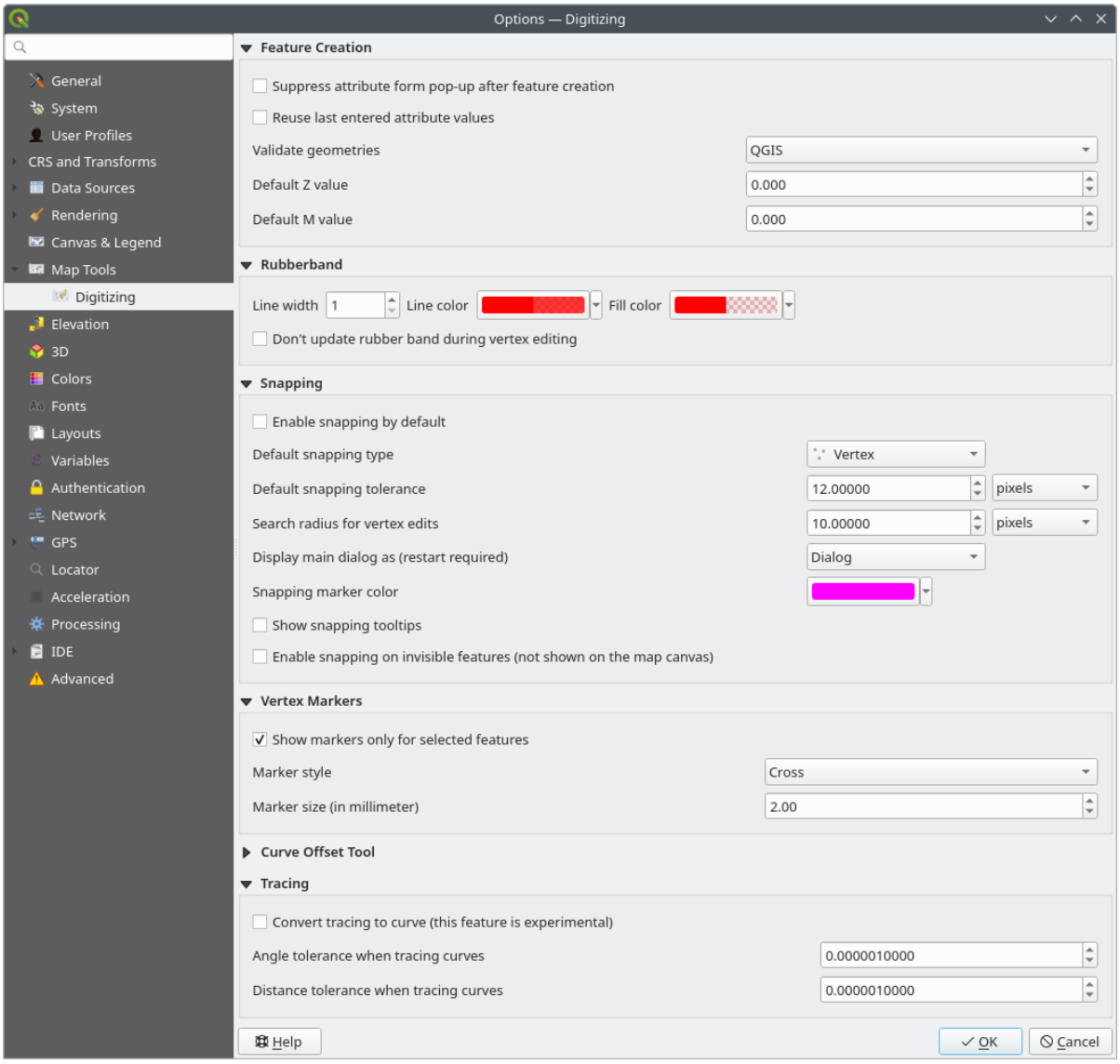


Abb. 5.15: Digitalisieren

Hier können Sie grundsätzlich Einstellungen für die *Bearbeitung eines Vektorlayers* festlegen (Geometrien und Attribute).

Objekterzeugung

- Mit *Attributeingabeformular bei der Objekterstellung unterdrücken* können Sie verhindern, dass das Attributeingabeformular automatisch bei der Erstellung eines neuen Objekts erscheint (Achtung: diese Auswahl kann für jeden Layer in den Layer-Eigenschaften überschrieben werden!).
- *Letzte Attributwerteingaben wiederverwenden*: Speichert den zuletzt verwendeten Wert jedes Attributs und verwendet ihn als Standardwert für das nächste zu digitalisierende Objekt. Funktioniert innerhalb eines Layers. Dieses Verhalten kann auch pro Feld gesteuert werden (siehe *Einstellung der Feldfunktionsweisen*).
- *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.

Beachten Sie, dass je nach Auswahl, die Berichte über Geometriefehler unterschiedlich ausfallen können (s. *Types of error messages and their meanings*)

- Der *Z-Vorgabewert* wenn neue 3D Objekte erstellt werden kann hier festgelegt werden.

Gummiband

- Hier können Sie die *Linienstärke*, *Linienfarbe* and die *Füllfarbe* festlegen.
- *Gummiband bei der Stützpunktbearbeitung nicht aktualisieren*.

Einrasten

- Mit *Einrasten voreinstellen* legen Sie fest, dass die Einrastfunktion aktiviert ist, wenn ein Projekt geöffnet wird.
- Sie können den *Vorgabeeinrasttyp* festlegen: (‚Stützpunkt', ‚Segment', ‚Fläche', ‚Zentroid', ‚Segmentmitte', ‚Linienendpunkte')
- Definieren Sie die *Einrasttoleranzvorgabe* in Karteneinheiten oder Pixeln.
- Definieren Sie den *Suchradius für Stützpunktbearbeitung* in Karteneinheiten oder Pixeln.
- Mit *Hauptfenster anzeigen in (erfordert Neustart)* können Sie festlegen, ob das Einrasten-Fenster als ‚Dialog' oder als ‚Dock' angezeigt wird.
- Die *Einrastmarkierfarbe* kann ausgewählt werden.
- Wenn *Einrasthinweise anzeigen* aktiviert ist, wird z.B. der Name des Layers angezeigt, an den gefangen wird. Das ist hilfreich, wenn mehrere Layer übereinander liegen.
- Sie können auch das *Einrasten auf verborgene (nicht auf der Karte sichtbare) Objekte aktivieren*.

Stützpunktmarkierungen

- Sie können *Markierungen nur für gewählte Objekte anzeigen* lassen.
- Legen Sie für die Stützpunktmarker den *Markierungsstil* (‚Kreuz' (standard), ‚Teiltransparenter Kreis' oder ‚Keine') fest
- Definieren Sie die *Markierungsgröße (in Millimeter)*

Werkzeug zum Versetzen von Linien

Die nächsten Optionen beziehen sich auf das Werkzeug Linie versetzen in *Erweiterte Digitalisierung*. Durch die verschiedenen Einstellungen kann die Form des Linienversatzes beeinflusst werden.

- *Verbindungsstil*: Legt fest, ob beim Versetzen von Ecken die Verbindungen „rund", „eckig" oder „abgefast" sein sollen.
- *Quadrantensegmente*: Steuert die Anzahl der Liniensegmente, die zur Annäherung an einen Viertelkreis beim Erstellen abgerundeter Versätze verwendet werden sollen.
- *Eckgrenze*: Legt den maximalen Abstand von der versetzten Geometrie fest, der beim Erstellen einer Gehrungsverbindung als Faktor des Versatzabstands verwendet werden soll (gilt nur für Gehrungsverbindungen).
- *Endstil*: Legt fest, wie Zeilenenden behandelt werden; sie können „rund", „flach" oder „quadratsich" sein.

Spurverfolgung

Wenn *Spurverfolgung in Kurve umwandeln* aktiviert ist, können Sie während der Digitalisierung automatisch Kurvensegemente erstellen. Bitte beachten Sie, dass es sich um eine experimentelle Funktion handelt und dass der Datenanbieter/Datentyp, den Sie verwenden, diese Funktion unterstützen muss.

5.1.9 Höhe

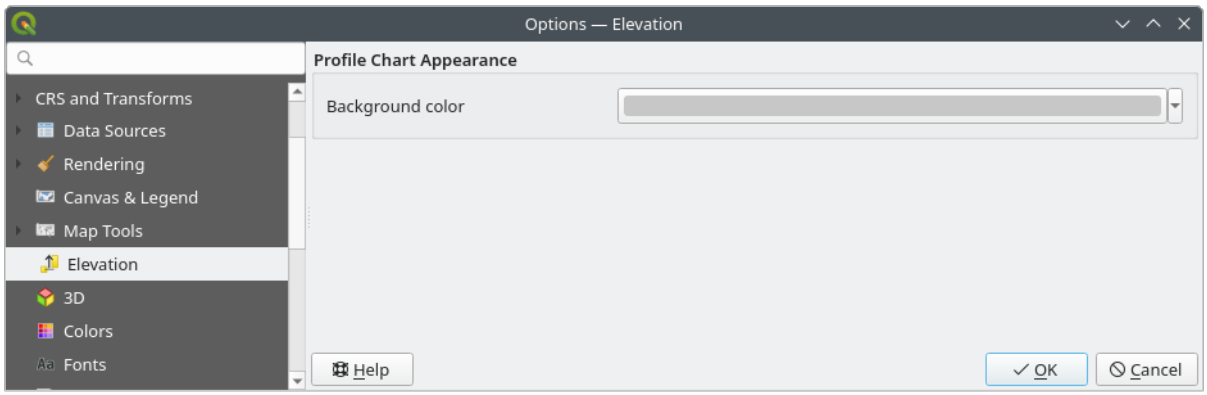


Abb. 5.16: Höhe

Im Menü *Höhe* können Sie eine bestimmte Farbe als *Hintergrundfarbe* für *Höhenprofile* festlegen. Dadurch kann das Diagramm für bestimmte Datensätze besser lesbar werden, z. B. für Punktwolken mit RGB-Farbgebung, bei denen die Standard-Hintergrundfarbe den Punktfarben zu ähnlich ist, um gut erkennbar zu sein. Wenn Sie keine Änderung vornehmen, werden die Höhenprofile weiterhin mit der Standard-Hintergrundfarbe des Systems angezeigt.

5.1.10 3D-Einstellungen

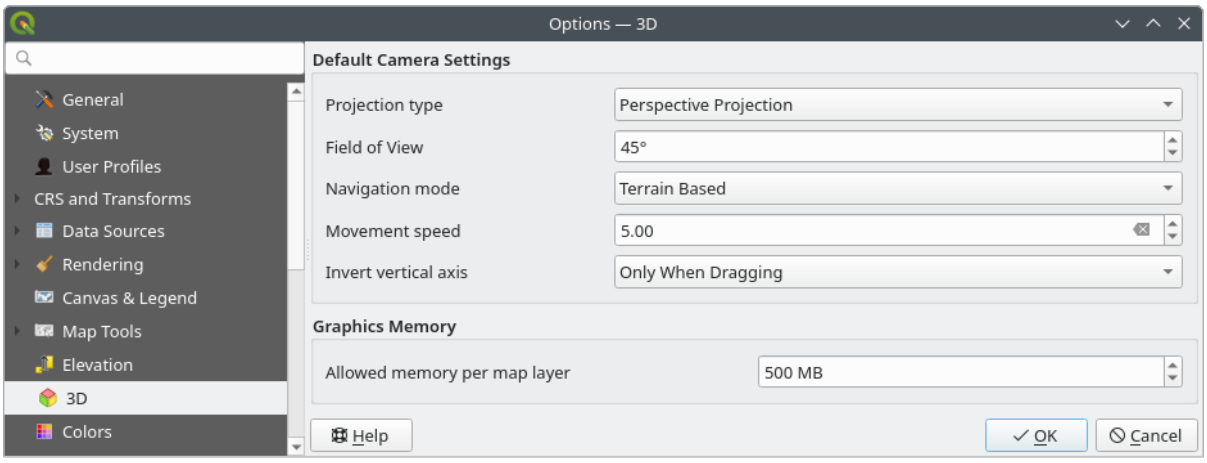


Abb. 5.17: 3D-Einstellungen

Der *3D* Reiter hilft Ihnen, einige Standardeinstellungen für *3D Kartenansichten* zu konfigurieren. Dies sind:

- *Kameravoreinstellungen*:
  - *Projektionstyp*: ermöglicht die Betrachtung der 3D-Kartenansicht in einer:
    - *Perspektivische Projektion* (Standard): Parallele Linien scheinen sich in der Ferne zu treffen. Objekte scheinen zu schrumpfen, je weiter sie von der Kamera entfernt sind.
    - oder eine *Orthogonale Projektion*: Parallele Linien erscheinen parallel. Objekte erscheinen unabhängig von der Entfernung in der gleichen Größe.
  - *Sichtfeld* der Kamera (nur bei perspektivischer Projektion relevant): legt einen Wert für das vertikale Sichtfeld in Grad an und bestimmt dadurch, wie viel der Ansicht für die Kamera sichtbar ist. Der Standardwert ist 45°.

  - *Navigationsmodus*: bietet verschiedene Möglichkeiten zur Interaktion mit der 3D-Kartenansicht. Verfügbare Modi sind:
    * *Gelände basierend*: Die Kamera folgt einer festen Position auf der Oberfläche des Geländes, während durch die Ansicht navigiert wird.
    * *Gehmodus (Erste Person)*

    Je nach gewähltem Modus unterscheiden sich die *Navigationsbefehle*.
  - *Bewegungsgeschwindigkeit*
  - *Vertikale Achse invertieren*: Legt fest, ob die Bewegungen der vertikalen Achse gegenüber ihrem normalen Verhalten invertiert werden sollen. Wirkt sich nur auf Bewegungen im *Gehmodus* aus. Die möglichen Einstellungen sind:
    * *Nie*
    * *Nur beim Ziehen*: bewirkt, dass die vertikale Bewegung nur dann invertiert wird, wenn eine Kameradrehung durch Klicken und Ziehen durchgeführt wird.
    * and *Immer*: bewirkt, dass die Bewegungen invertiert werden, sowohl beim Klicken und Ziehen als auch wenn die Kamerabewegung an den Cursor gebunden ist (durch Drücken der ~-Taste).

- Unter *Grafikspeicher* können Sie mit der Option *Gestatteter Speicher pro Kartenlayer* das GPU-Speicherlimit für jeden Layer festlegen. Dies kann nützlich sein, wenn Sie 3D-Kartenansichten mit großem GPU-Speicherbedarf erstellen. Wenn das eingestellte Limit erreicht wird, wird eine Warnung angezeigt, die bei der Problemlösung helfen soll.

5.1.11 Farben

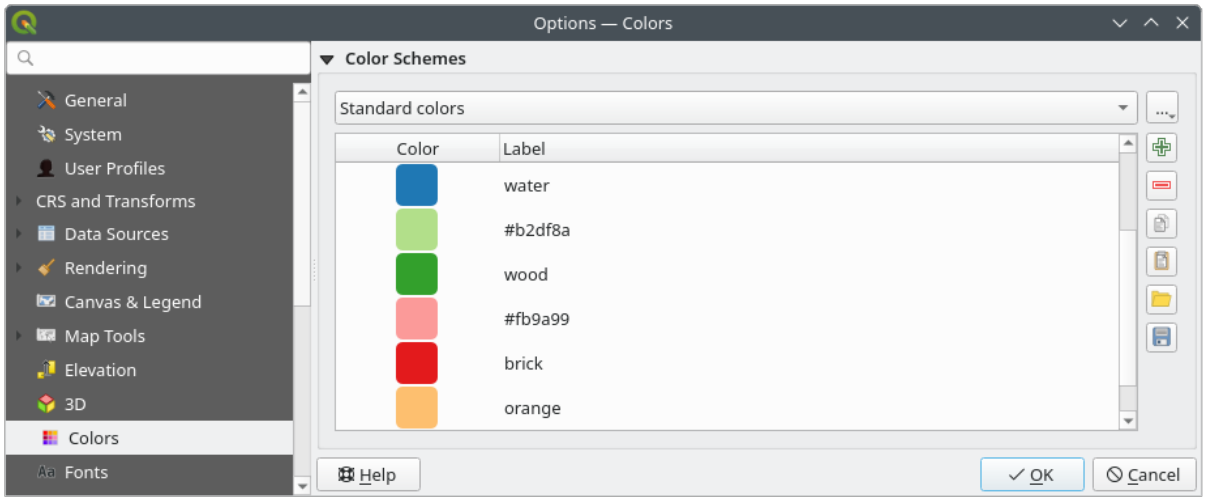


Abb. 5.18: Farben

Mit diesem Menü können Sie Farbpaletten erstellen oder verändern, die in der gesamten Anwendung über die *Farbauswahl* aufgerufen werden können. Über das Dropdownmenü können Sie die folgenden Farbschemata aufrufen:

- *Zuletzt verwendete Farben* zeigt eben diese
- :guilabel:`Standardfarben“
- *Projektfarben*, eine Reihe von Farben, die für das aktuelle Projekt festgelegt sind (siehe *Stile* für weitere Details)
- *Farben für neue Layer*, ein Farbschema, das standardmäßig verwendet wird, wenn neue Layer zu QGIS hinzugefügt werden

- oder benutzerdefinierte Palette(n), die Sie mit der Schaltfläche … neben der Dropdown-Menü erstellen oder importieren können.

Die Paletten *Zuletzt verwendete Farben*, *Standardfarben* und *Projektfarben* können nicht entfernt werden und erscheinen immer als Farbknöpfe im Dropdown-Menü der Farbauswahl. Über die Option *Anzeigen in Farbschaltflächen* können auch benutzerdefinierte Paletten an dieser Stelle zur Verfügung gestellt werden.

Für jede der Paletten können Sie die Liste der Farben mit den Werkzeugen auf der rechten Seite verwalten:

- Eine *Farbe hinzufügen* oder eine *Farbe entfernen*
- Eine *Farbe kopieren* oder eine *Farbe einfügen*
- Die ganze Farbpalette als `.gpl` Datei *Importieren* oder *Exportieren*.

Doppelklicken Sie auf eine Farbe in der Liste, um sie mit dem *Farbauswahl Dialog* zu ändern. Mit einem Doppelklick in die Spalte *Beschriftung* können Sie die Farbe auch benennen bzw. umbenennen.

5.1.12 Schriftarten

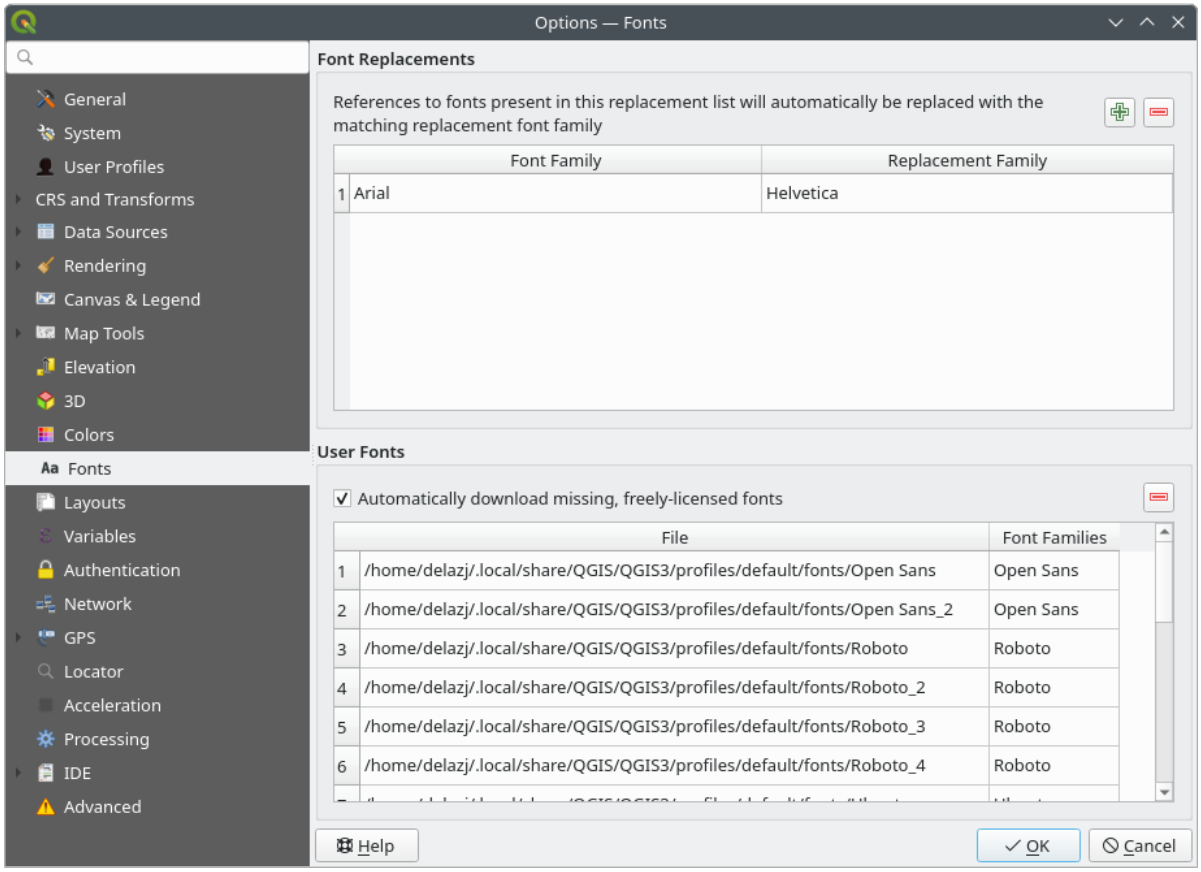


Abb. 5.19: Der Schriftarten-Reiter

Der *Schriftarten* Reiter bietet Unterstützung bei der Verwaltung der projektübergreifend verwendeten Schriftarten:

- *Schriftartenersetzungen*: Ermöglicht das Erstellen einer Liste mit automatischen Schriftartersetzungen, die beim Laden von Projekten oder Stilen angewendet werden, wodurch Projekte und Stile, die auf verschiedenen Betriebssystemen verwendet werden, besser unterstützt werden (z. B. Ersetzen von „Arial” durch „Helvetica”).
- *Benutzerschriftarten*: Ermöglicht es Ihnen, TTF- oder OTF-Schriftarten im Unterordner `fonts` des *Benutzerprofils* abzulegen. Diese Schriftarten können beim Start von QGIS automatisch geladen werden. Auf

diese Weise können Sie Schriftarten verwenden, ohne dass sie auf Betriebssystemebene installiert sein müssen, was in Unternehmensumgebungen häufig blockiert wird. Das Fenster listet alle installierten Benutzerschriften auf und ermöglicht Ihnen die Verwaltung (d. h. das Entfernen) zuvor installierter Benutzerschriften.

Es ist auch möglich, das Kontrollkästchen „Automatisch fehlende frei lizenzierte Schriftarten herunterladen“ zu aktivieren: Wenn Sie beispielsweise ein Projekt oder einen Stil öffnen oder versuchen, eine Vektorkachel-Ebene zu laden, die auf derzeit nicht verfügbare Schriftarten verweist, wird eine fest codierte Liste frei lizenzierter Schriftarten zum Herunterladen über eine URL konsultiert, um festzustellen, ob die Schriftart automatisch in das Schriftartenverzeichnis des Benutzerprofils heruntergeladen werden kann (mit Benachrichtigung über die Schriftartlizenz).

5.1.13 Layouts

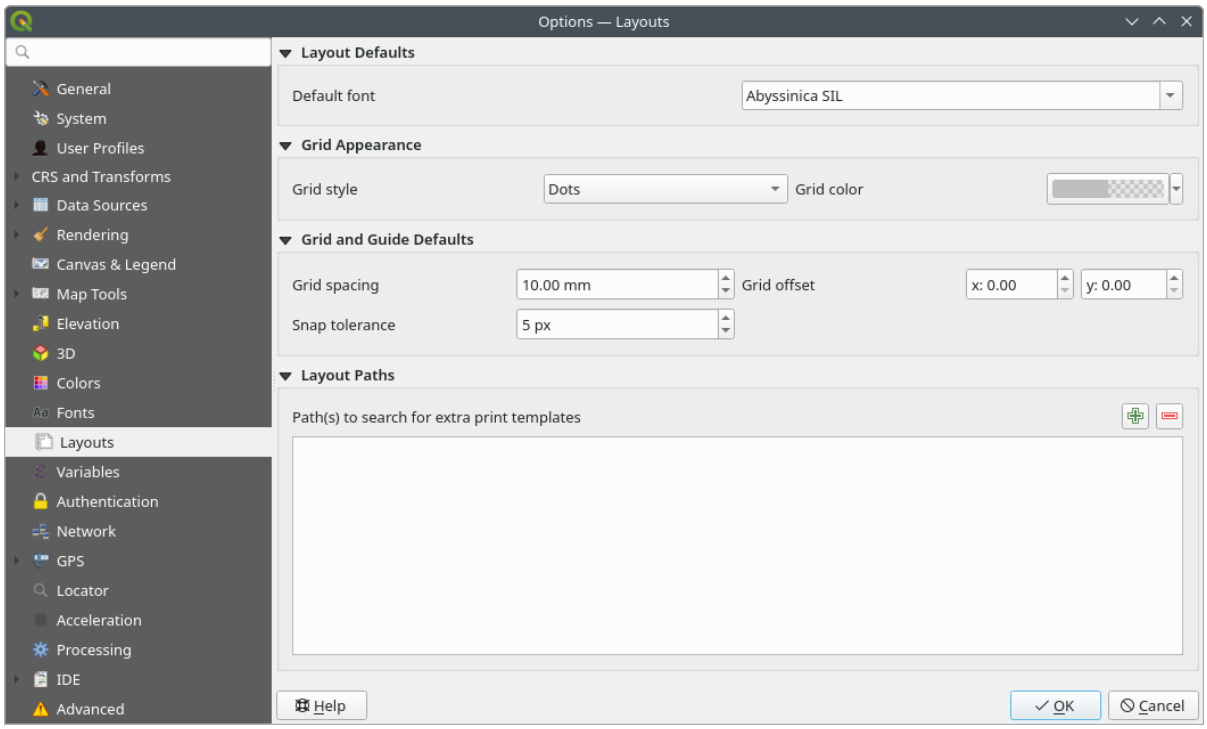


Abb. 5.20: Layouts

Layoutvorgaben

Sie können die *Vorgabeschriftart* definieren, die für *Kartenlayouts* verwendet wird.

Gitterdarstellung

- Definieren Sie den *Gitterstil*, die möglichen Optionen sind ‚Ausgefüllt‘, ‚Punkte‘ (Standard) und ‚Kreuze‘.
- Sie können auch die *Gitterfarbe* defnieren.

Gitter- und Führungsvoreinstellungen

- Definieren Sie den *Gitterabstand*
- Definieren Sie den *Gitterversatz* jeweils für die x- und die y-Achse getrennt
- Definieren Sie die *Einrasttoleranz*, mit der am Gitter gefangen wird

Layoutpfade

- Sie können *Pfad(e), um nach zusätzlichen Druckvorlagen zu suchen*: hinzufügen; wenn Sie die Layout-Verwaltung öffnen, werden auch die Dateien, die sich in den hinzugefügten Ordnern befinden, als Vorlagen angezeigt.

5.1.14 Variablen

Der Reiter *Variablen* listet alle auf der globalen Ebene verfügbaren Variablen auf.

Es ist auch möglich, eigene Variablen zu definieren. Klicken Sie auf die Schaltfläche , um eine neue benutzerdefinierte Variable auf globaler Ebene hinzuzufügen. Wenn Sie eine benutzerdefinierte Variable aus der Liste auswählen, können Sie diese bearbeiten oder mit der Schaltfläche wieder entfernen.

Mehr Informationen über Variable erhalten sie im Abschnitt *Speichern von Werten in Variablen*.

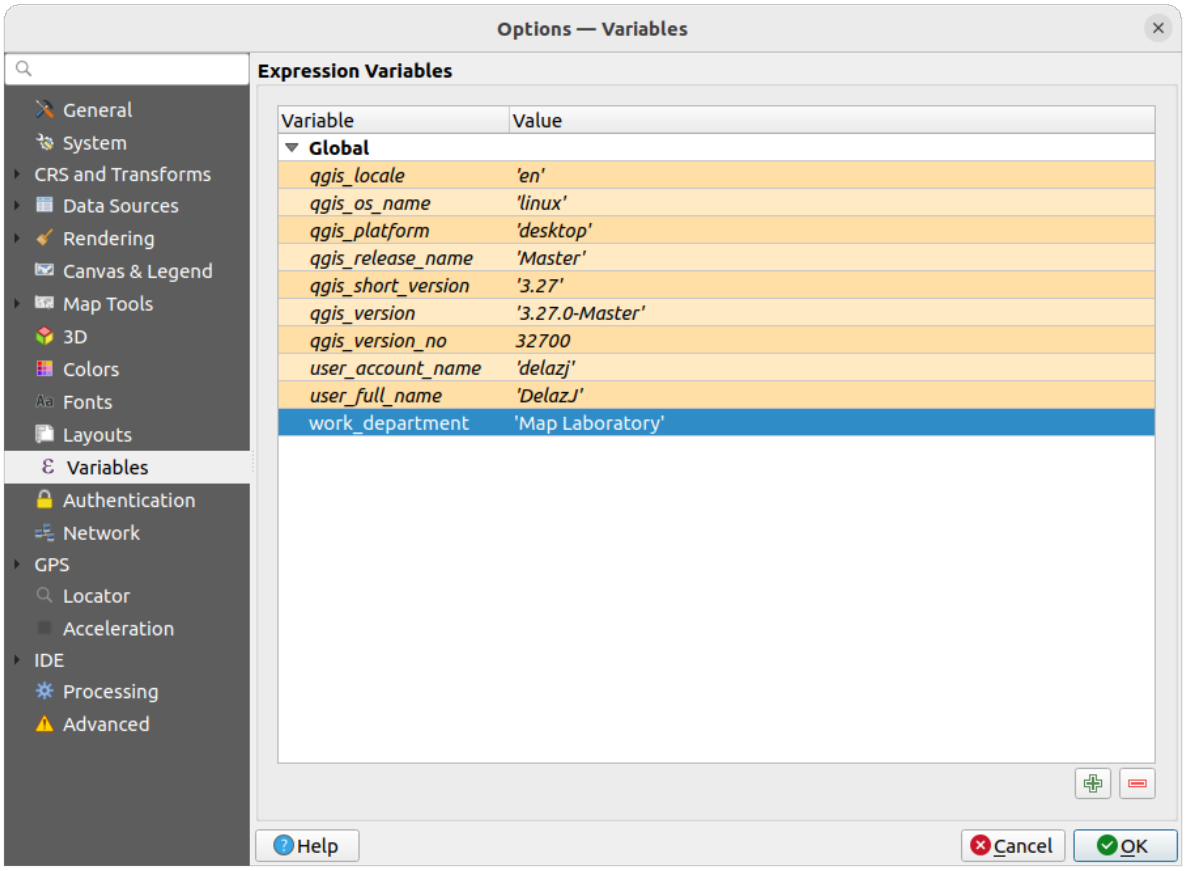


Abb. 5.21: Variablen

5.1.15 Authentifizierung

Auf dem Reiter *Authentifizierung* können Sie Authentifikationskonfigurationen vornehmen und PKI Zertifikate verwalten. Für mehr Details siehe *Authentifizierungssystem*.

Zur Verwaltung der Authentifizierungen können Sie die Liste der Werkzeuge neben dem Rahmen verwenden:

- Neue Authentifizierungskonfiguration hinzufügen
- Gewählte Authentifizierungskonfiguration löschen
- Gewählte Authentifizierungskonfiguration bearbeiten

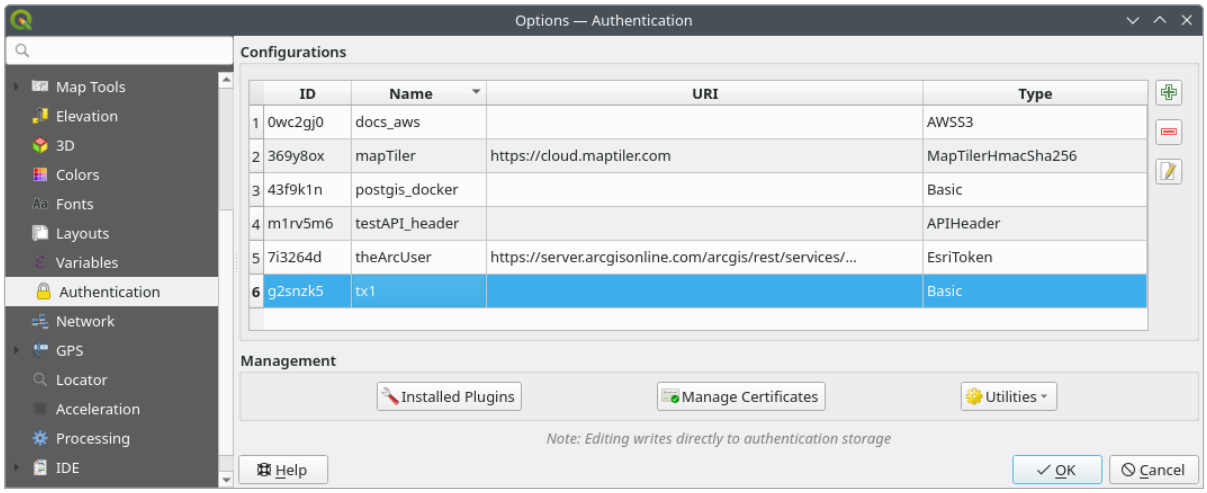


Abb. 5.22: Authentifizierung

5.1.16 Netzwerk

Allgemein

- Definieren Sie die *Zeitüberschreitung bei Netzwerkanfragen (ms)* - der Standardwert ist 60.000ms.
- Definieren Sie die *Verfallszeitraumvorgabe für WMS-Eigenschaften (Stunden)* - der Standardwert ist 24 Stunden.
- Definieren sie die *Verfallszeitraumvorgabe für WMS-C-/WMTS-Kacheln (Stunden)* - der Standardwert ist 24 Stunden.
- Definieren Sie die *Max. Wiederholungen bei Kachel- oder Objektabfragefehlern* - der Standardwert ist 3.
- Definieren Sie *User-Agent-Präfix*, dessen Standardwert `Mozilla/5.0` lautet. Dieser Wert wird sowohl der QGIS- als auch der Betriebssystemversion vorangestellt, z. B. `Mozilla/5.0 QGIS/33801/Ubuntu 22.04.4 LTS`, um den User-Agent zu formen.

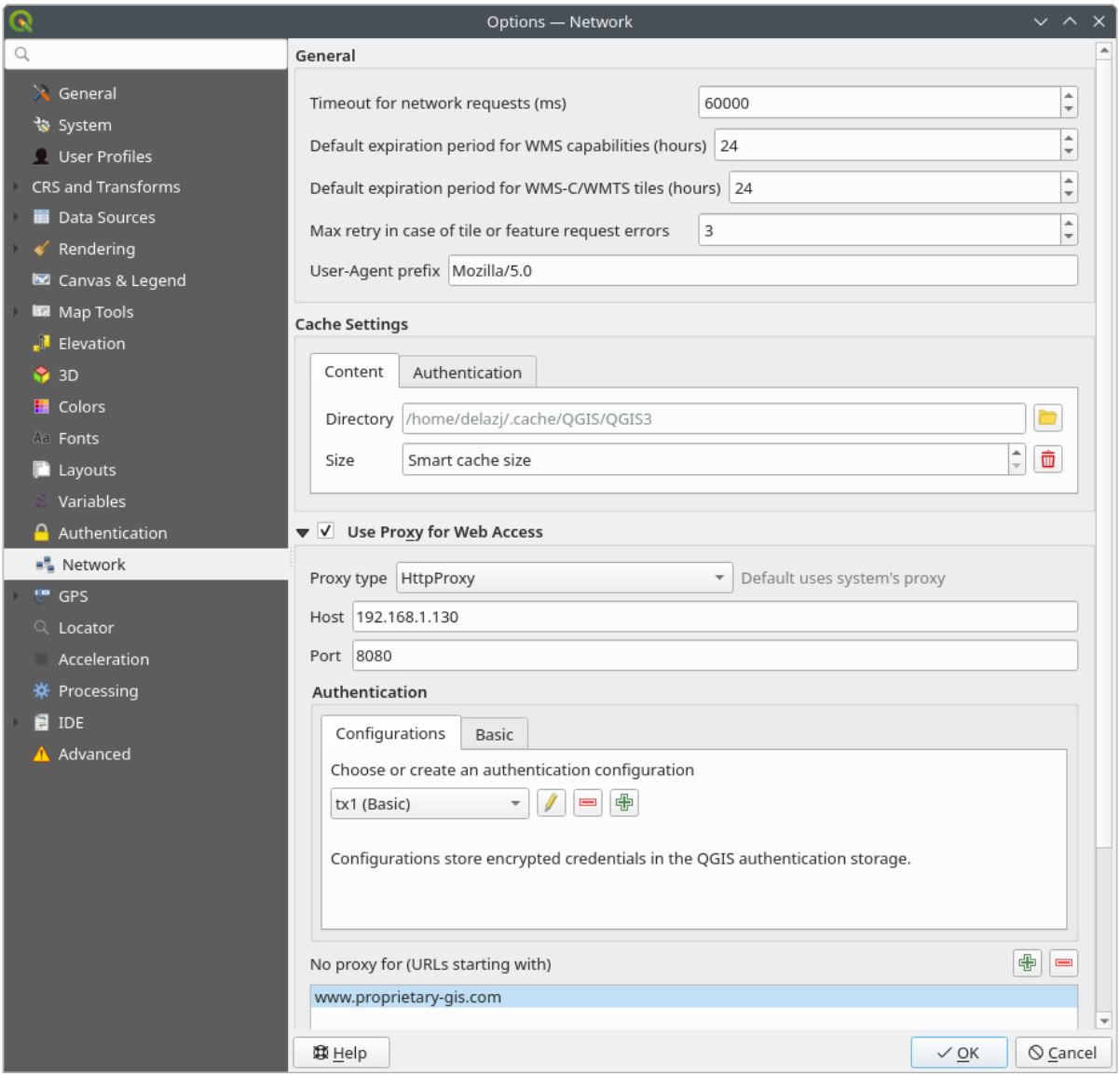


Abb. 5.23: Optionen auf dem Netzwerk-Reiter

Cache-Einstellungen

Definiert das *Verzeichnis* und eine *Größe* für den Cache in Megabyte. Sie können den Wert löschen, um die Option *Angepaßte Cachegröße* zu aktivieren, die die maximale Cachegröße basierend auf dem verfügbaren Speicherplatz festlegt. Bietet außerdem Tools zum *automatischen Löschen des Verbindungsauthentifizierungs-Caches bei SSL-Fehlern (empfohlen)*.

Proxy-Einstellungen

- *Proxy für Webzugriff verwenden*
- Stellen Sie den *Proxytyp* entsprechend Ihren Bedürfnissen ein und definieren Sie ‚Host' und ‚Port'. Verfügbare Proxy-Typen sind:
  - *DefaultProxy*: Der Proxy wird auf der Grundlage des System-Proxys bestimmt
  - *Socks5Proxy* ►: Proxy für jede Art von Verbindung. Unterstützt TCP, UDP, Bindung an einen Port (eingehende Verbindungen) und Authentifizierung.
  - *HttpProxy* ►: Umgesetzt mit dem „CONNECT"-Befehl, unterstützt nur ausgehende TCP-Verbindungen; unterstützt Authentifizierung.

- *HttpCachingProxy*: Umgesetzt mit normalen HTTP Befehlen ist dies nur im Zusammenhang mit HTTP Befehlen sinnvoll einzusetzen.
- *FtpCachingProxy*: Mit einem FTP-Proxy umgesetzt ist dies nur sinnvoll im Zusammenhang mit FTP-Anforderungen anzuwenden

Die Berechtigungsnachweise des Proxys werden mit dem *Authentifizierungs-Werkzeug* gesetzt.

Im Textfeld unter den Proxy-Einstellungen kann der Ausschluss bestimmter URLs eingetragen werden (siehe Abb. 5.23). Es wird kein Proxy verwendet, wenn die Ziel-URL mit einer der in diesem Textfeld aufgeführten Zeichenfolgen beginnt.

Wenn Sie detailliertere Informationen über die verschiedenen Proxy-Einstellungen benötigen, lesen Sie bitte das Handbuch der zugrundeliegenden QT-Bibliotheksdokumentation unter https://doc.qt.io/archives/qt-5.9/qnetworkproxy.html#ProxyType-enum.

Tipp

Proxy richtig anwenden

Die Verwendung von Proxys kann manchmal schwierig sein. Es ist sinnvoll, im Zweifel die oben genannten Proxy-Typen auszuprobieren, um herauszufinden, ob sie in Ihrem Fall funktionieren.

5.1.17 GPS

Der *GPS* Reiter hilft Ihnen bei der Konfiguration der Verbindungen mit GPS-Geräten und dem Einlesen von GPS-Daten. Er bietet auch Einstellungen für GPS-Tracking und Daten-Digitalisierung.

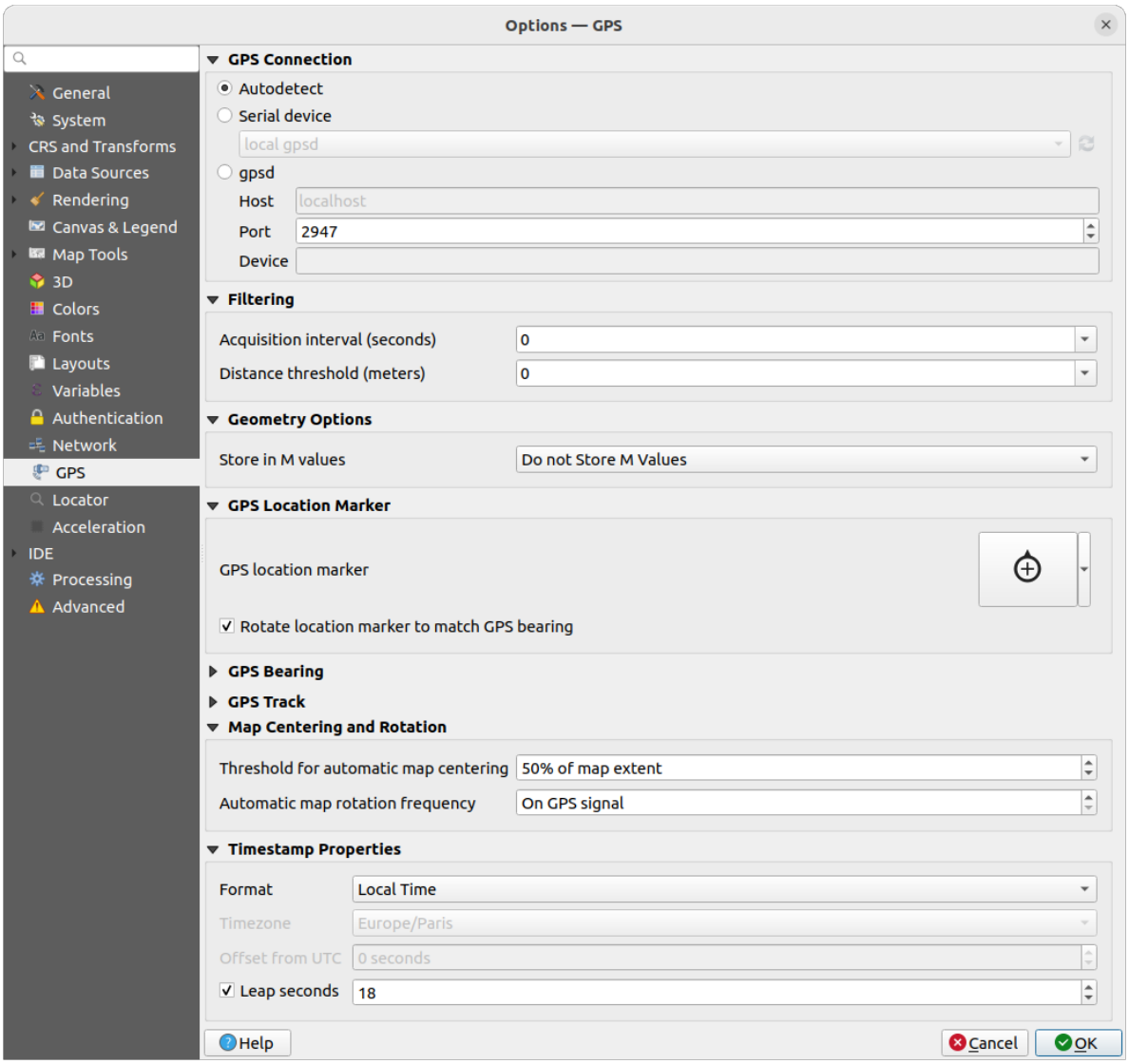


Abb. 5.24: GPS

Es gibt die folgenden Optionen:

- *GPS-Verbindungen* bietet verschiedene Möglichkeiten, das Gerät mit QGIS zu verbinden:
  - *Automatisch feststellen*
  - *Serielles Gerät* erfordert ein Neuladen, wenn ein neues GPS-Gerät angeschlossen wird.
  - *gpsd*: Wählen Sie den *Host* und den *Port* aus, mit denen Ihr GPS verbunden ist; Sie können auch einen Namen für das *Gerät* eingeben.

  Im Falle von Verbindungsproblemen können Sie versuchen, die Verbindungsart zu wechseln.
- *Filterung*: Sie können einen *Akquisitionsintervall (Sekunden)* und/oder eine *Abstandsschwelle (Meter)* einstellen, um den Cursor auch dann aktiv zu halten, wenn sich der Empfänger in einem statischen Zustand befindet.
- Unter *Geometrieoptionen* können mithilfe von *In M-Werte speichern* die M-Werten aus den eingebauten GPS-Logging-Tools übernommen werden. Dies gilt sowohl für Features, die aus GPS-Logs digitalisiert wurden, als auch für die *In Geopackage/SpatiaLite protokollieren…* Funktionalität. Die Möglichkeiten zur Speicherung von M-Werten sind u.a. die Speicherung von Zeitstempeln (in ms seit Epoche), Geschwindigkeit über Grund, Höhen, Peilungen und Genauigkeitskomponenten.

- *GPS-Positionsmarkierung*
  - Sie können das Symbol für den *GPS-Standortmarker* konfigurieren, das die aktuelle GPS-Position angibt.
  - Mit *Positionsmarkierung in GPS-Kurs drehen* können Sie auswählen, ob das Markierungssymbol entsprechend der GPS-Richtung gedreht werden soll.
- *GPS-Kurs*:
  - Sie können einen *Kurslinienstil* unter der Verwendung der QGIS Liniensymboleigenschaften konfigurieren.
  - Wenn die Option *Richtung aus Reiserichtung berechnen* aktiviert ist, wird die vom GPS-Gerät gemeldete Peilung ignoriert und die Peilung wird stattdessen anhand des Winkels zwischen den beiden vorherigen GPS-Positionen berechnet.
- Unter *GPS-Spur* können Sie ein Symbol für den *Spurlinienstil* setzen.
- Im Bereich *Kartenzentrierung und -drehung* können sie festlegen, wann die Kartenansicht aktualisiert wird:
  - *Schwelle für automatische Kartenzentrierung*: definiert die minimale Abweichung der GPS-Position von der Kartenmitte, um eine *automatische Kartenzentrierung* auszulösen.
  - *Frequenz der automatischen Kartendrehung*: legt fest, wie oft die Karte gedreht wird, um mit der GPS-Peilung übereinzustimmen; dies kann für eine benutzerdefinierte Dauer (in Sekunden) oder *bei GPS-Signal* erfolgen.
- Unter *Zeitstempeleigenschaften* können sie konfigurieren, wie Zeitwerte in den Daten angezeigt und gespeichert werden. Die Einstellungen umfassen das *Format*, für welches es die folgenden Optionen gibt:
  - *Lokale Zeit*
  - *UTC*
  - *UTC mit Versatz*: damit ist es möglich, Sommerzeitverschiebungen oder anderen komplexen Zeitzonenproblemen zu berücksichtigen
  - *Zeitzone*: es kann eine bestimmte Zeitzone ausgewählt werden

  Mit der Option *Schaltsekunde* können die Zeiten korrigiert werden, indem die Sekunden zum GPS-Zeitstempel addiert werden.

GPSBabel

Mit GPSBabel können Wegpunkte, Tracks und Routen zwischen gängigen GPS-Empfängern wie Garmin oder Magellan und Kartenprogrammen wie Google Earth oder Basecamp konvertiert werden. Von GPSBabel werden sehr viele GPS-Empfänger und Programme unterstützt. QGIS greift auf die GPSBabel Datenbank zurück, um mit den Geräten zu kommunizieren und die *Daten zu bearbeiten*.

Einzelheiten zur Vorgehensweise finden Sie unter *Laden auf oder von einem Gerät*.

5.1.18 Suche

Auf dem *Suche* Reiter können Sie die <locator_bar> konfigurieren, die sich unten in der Statusleiste befindet. Es gibt die folgenden Standardfilter, für welche Sie verschiedene Einstellungen (s.u.) vornehmen können (in Klammern jeweils der Präfix):

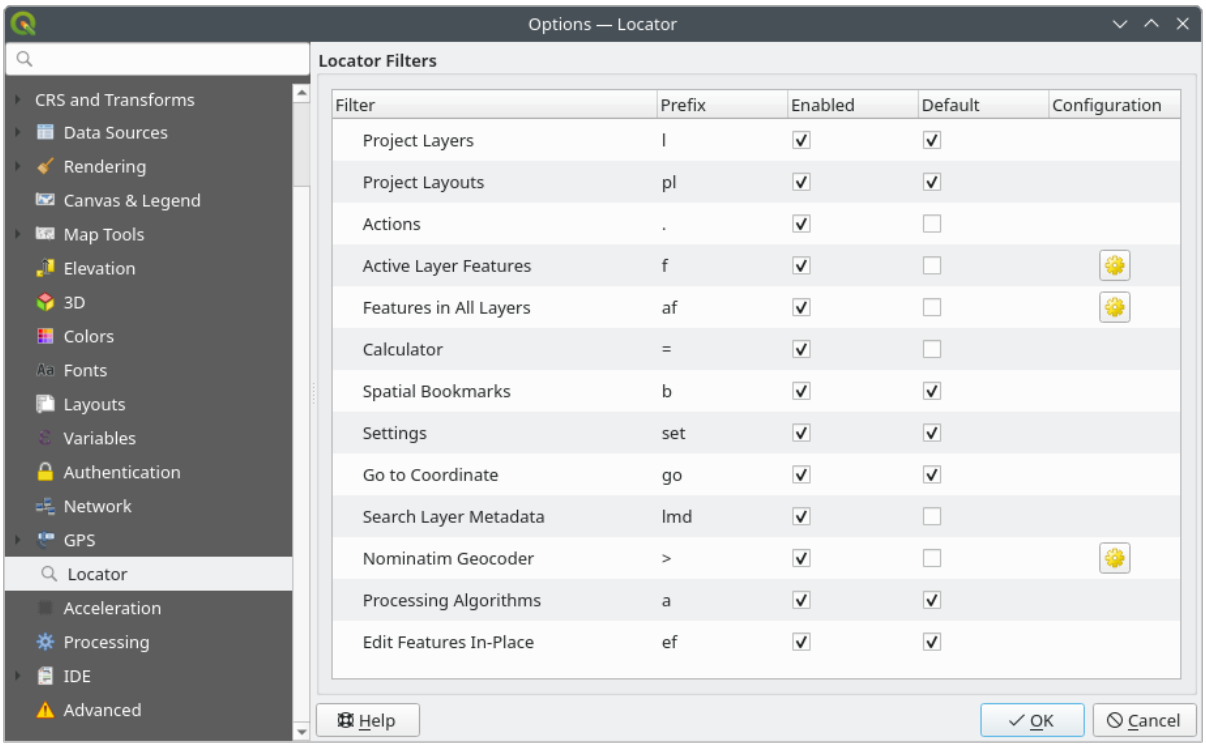


Abb. 5.25: Suche

- *Projektlayer* (`l`): sucht einen Layer im *Layer* Bedienfeld und wählt ihn aus.
- *Projektlayouts* (`pl`): findet und öffnet ein Drucklayout.
- Aktionen (`.`): findet und führt eine QGIS-Aktion aus; Aktionen können jedes Werkzeug oder Menü in QGIS sein, z.B. auch das Aktivieren eines Bedienfeldes.
- *Objekte des aktiven Layers* (`f`): sucht nach übereinstimmenden Attributen in allen Feldern des aktiven Layers und zoomt dann auf das ausgewählte Objekt. Drücken Sie am Ende der Zeile, um die Anzahl der maximal angezeigten Ergebnisse einzustellen.
- *Objekte in allen Layern* (`af`): sucht nach übereinstimmenden Attributen in allen Feldern aller *durchsuchbarer Layer* und zoomt dann auf das ausgewählte Objekt. Drücken Sie am Ende der Zeile, um die Anzahl der maximal angezeigten Ergebnisse und die maximale Anzahl von Ergebnissen pro Layer einzustellen.
- *Rechner* (`=`): ermöglicht die Evaluation eines beliebigen QGIS-Ausdrucks und bietet die Möglichkeit, das Ergebnis in die Zwischenablage zu kopieren, wenn es gültig ist.
- *Räumliche Lesezeichen* (`b`): sucht in den räumlichen Lesezeichen und zoomt dann auf den entsprechenden Bereich.
- *Einstellungen* (`set`): durchsucht und öffnet projekt- und anwendungsweite Eigenschaftsdialoge.
- *Zu Koordianten gehen* (`go`): schwenkt die Kartenansicht zu einem Punkt, der angegeben wird durch x- und y-Koordinaten (getrennt durch Komma oder Leerzeichen) oder eine formatierte URL (z.B. OpenStreetMap, Leaflet, OpenLayer, Google Maps, …). Die Koordinaten werden in WGS 84 (`epsg:4326`) und/oder im KBS der Karte erwartet.
- *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.
- Verarbeitungsalgorithmen (`a`): durchsucht die Verarbeitungsalgorithmen und öffnet den entsprechenden Dialog.

- *Ausgewählte Objekte bearbeiten* (`ef`): bietet Schnellzugriff auf kompatible *insitu* Verarbeitungsalgorithmen des aktiven Layers.

Im Dialog können Sie:

- den Filter anpassen mit: guilabel:*Prefix*, d. h. das Schlüsselwort, mit dem der Filter ausgelöst werden soll.
- Festlegen, ob der Filter *Eingeschaltet* ist: der Filter kann bei der Suche verwendet werden und eine Verknüpfung ist im Menü der Suchleiste verfügbar.
- *Voreinstellung* aktivieren/deaktivieren: Eine Suche, bei der kein Filter-Präfix vorangestellt wird, liefert nur Ergebnisse aus den Kategorien, bei denen hier ‚Voreinstellung' aktiviert ist.
- Einige Filter bieten die Möglichkeit, die Anzahl der Ergebnisse einer Suche zu begrenzen.

Der Umfang der Standard-Suchleisten-Filter kann durch Plugins erweitert werden, z.B. für OSM-Nominatim-Suchen, direkte Datenbanksuchen, Layer-Katalog-Suchen, usw.

5.1.19 Beschleunigung

OpenCL-Beschleunigungseinstellungen.

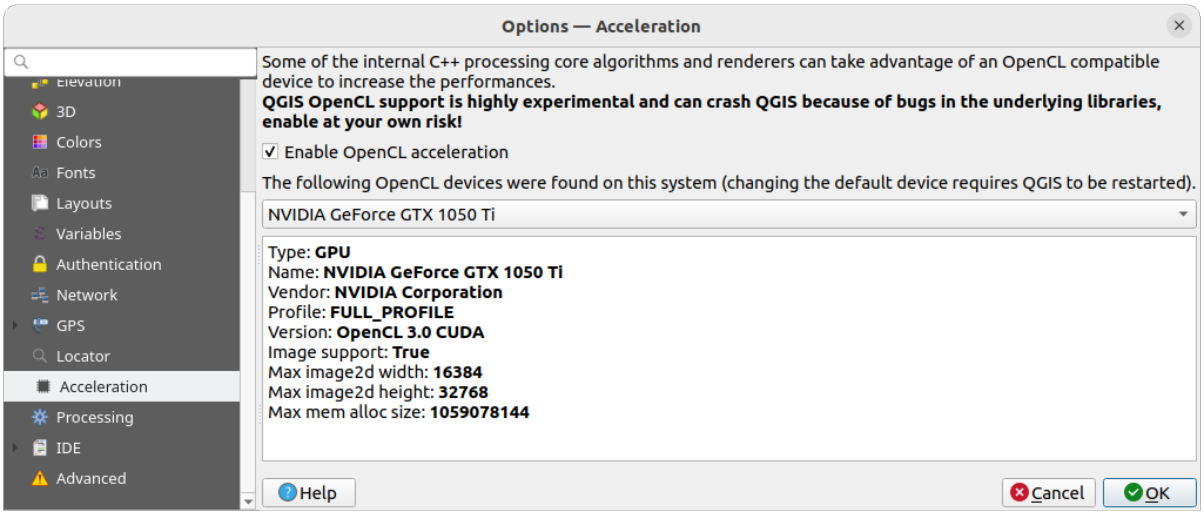


Abb. 5.26: Beschleunigung

Abhängig von Ihrer Hard- und Software müssen Sie möglicherweise zusätzliche Bibliotheken installieren, um die OpenCL-Beschleunigung aktivieren zu können.

5.1.20 IDE

Unter *GitHub-Zugriffstoken* können Sie ein persönliches Token generieren, mit dem Sie Codeausschnitte aus dem Python-Code-Editor heraus teilen können. Weitere Details zur GitHub-Authentifizierung

Code-Editor

Auf dem *Code Editor* Reiter können Sie das Aussehen und Verhalten der Code-Editor-Fenster (interaktive Python-Konsole und -Editor, Ausdrucksgenerator, Funktions-Editor, …) steuern.

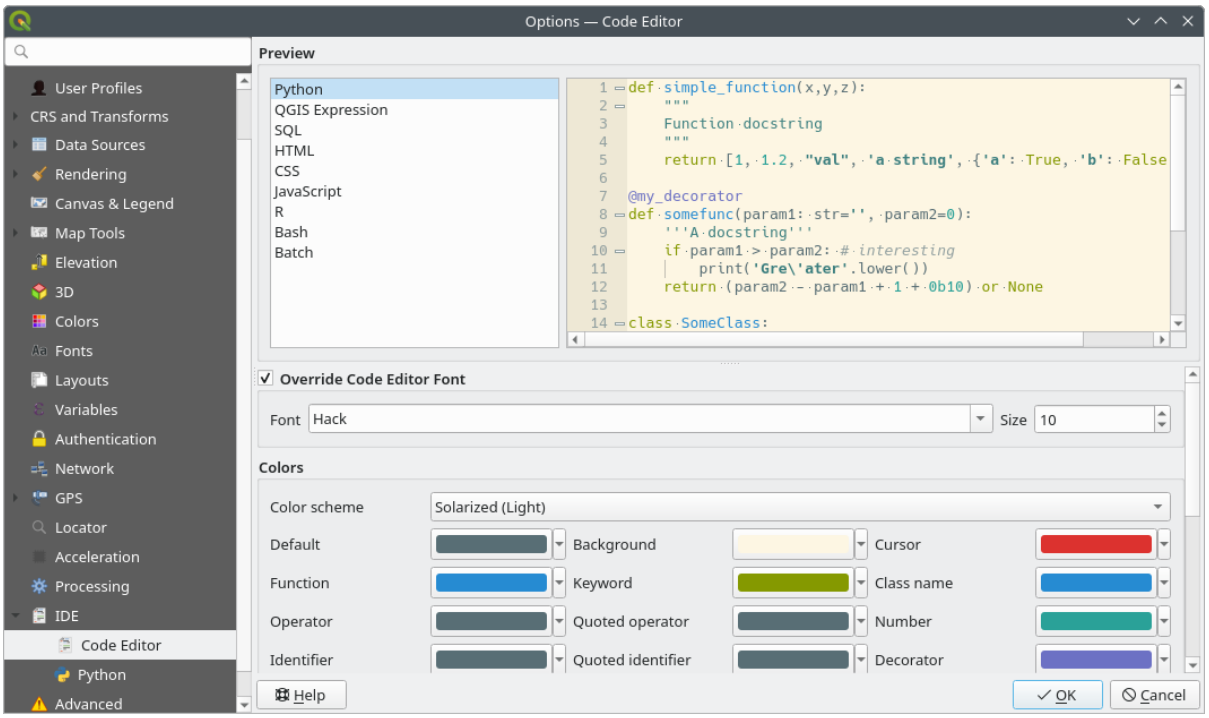


Abb. 5.27: Code-Editor

Am oberen Rand des Reiters bietet ein Fenster eine Live-Vorschau der aktuellen Einstellungen in verschiedenen Programmiersprachen (Python, QGIS-Expression, HTML, SQL, JavaScript).

- Aktivieren Sie *Code-Editorschriftart übersteuern*, um die *Schriftart* und die *Size* zu ändern.
- Im Bereich *Farben* haben Sie die folgenden Einstellungsmöglichkeiten:
  - Wählen Sie ein *Farbschema*: Vordefinierte Schemata sind `Voreinstellungen`, `Solarized (Dunkel)` und `Solarized (Hell)`. Ein `Benutzerdefiniertes` Schemata wird aktiviert, sobald Sie eine Farbe ändern und kann durch Auswahl eines vordefinierten Schemas zurückgesetzt werden.
  - Ändern Sie die *Farbe* für jedes Element beim Schreiben von Code, z.B. für Kommentare, Anführungszeichen, Funktionen, Hintergrund, usw.

Python

Die Einstellungen unter *Python* helfen Ihnen bei der Verwaltung und Steuerung des Verhaltens der Python-Editoren (*interaktive Konsole*, *Code-Editor*, *Projektmakros*, *benutzerdefinierte Ausdrücke*, …). Diese Einstellungen können auch über die Optionen…-Schaltfläche an den folgenden Stellen aufgerufen werden:

- der *Python-Konsole* Werkzeugleiste,
- über das Kontextmenü der *Python-Konsole*,
- über das Kontextmenü des Code-Editors.

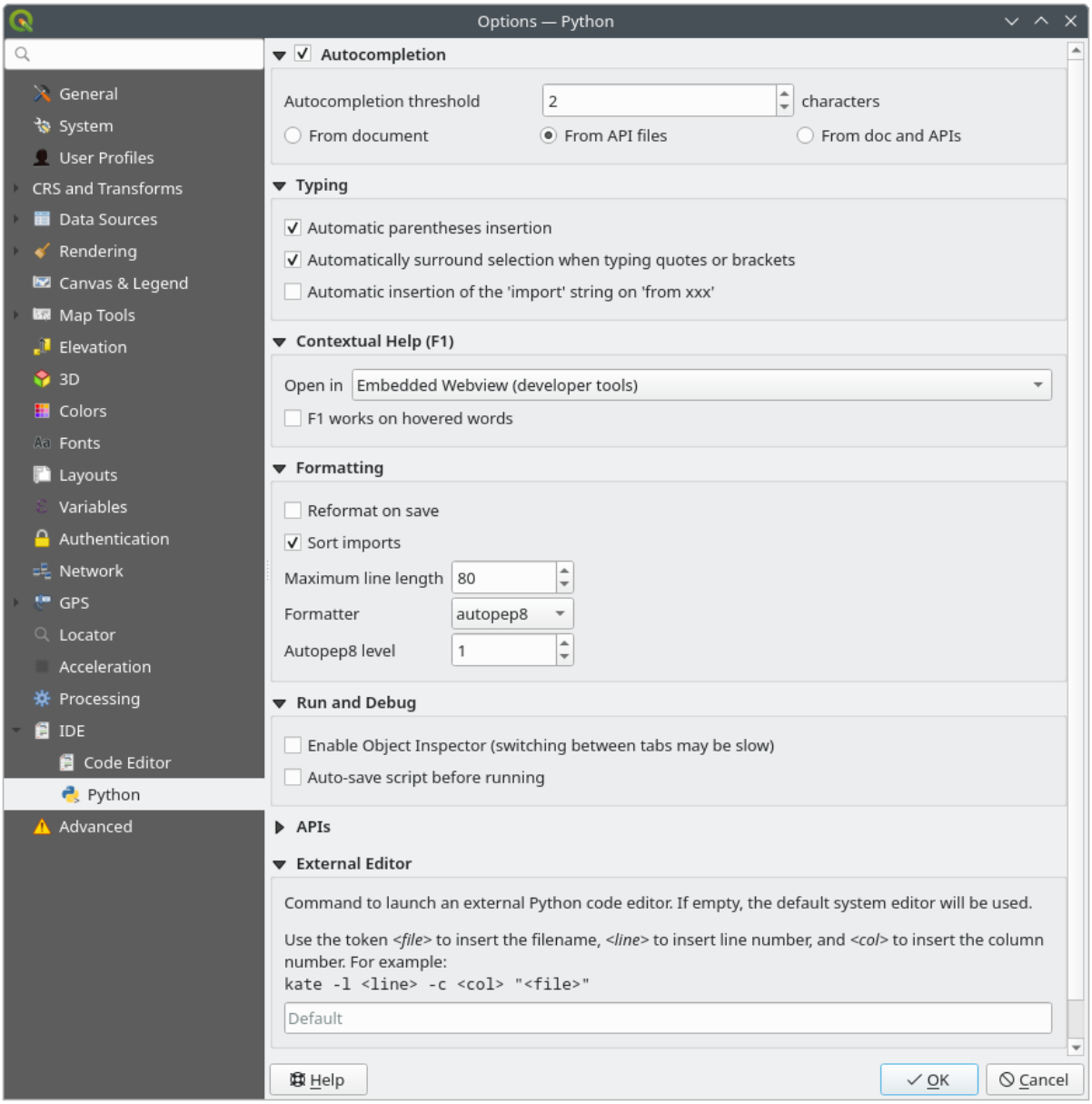


Abb. 5.28: Python

Es gibt die folgenden Optionen:

- *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)
- *Tippen*
  - *Automatisch Klammern einfügen*: Wenn kein Text ausgewählt ist, wird bei der Eingabe eines öffnenden Zeichens (Klammern, Anführungszeichen, eckige Klammern, …) das passende schließende Zeichen direkt hinter dem Cursor eingefügt. Beachten Sie, dass dieses Verhalten deaktiviert ist, wenn sich der aktuelle Cursor innerhalb einer Zeichenfolge oder eines Kommentars befindet.
  - *Auswahl automatisch beim Tippen von Anführungszeichen oder Klammern umschließen*: Wenn ein öffnendes Zeichen eingegeben wird, wird der ausgewählte Text in das öffnende/schließende Paar

eingeschlossen. Die Auswahl bleibt erhalten, sodass es möglich ist, ein ausgewähltes Wort zu zitieren und es in Klammern zu setzen, indem man einfach `"` und dann `(` eingibt.

Sonderfall für mehrzeilige Auswahl mit Anführungszeichen und doppelten Anführungszeichen: Auswahl wird in dreifache einfache/doppelte Anführungszeichen eingeschlossen.

- ☐ *Automatisches Einfügen der Zeichenkette ‚import' bei ‚from xxx'*: Aktiviert das Einfügen von ‚import' bei der Angabe von Importen.

- *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*.

- unter *Formatierung* können Sie Werkzeuge einstellen, um den von Ihnen geschriebenen Code automatisch zu formatieren:
  - ☐ *Beim Speichern neuformatieren*: Formatierung wird unmittelbar vor dem Speichern des Codes angewendet
  - ☑ *Importe sortieren*: sortiert ‚import'-Anweisungen unter Verwendung der isort-Bibliothek
  - *Maximale Zeilenlänge*: steuert, wie der Formatierer die Zeilen umbricht, und steuert das Editor-Lineal
  - *Formatierer*: unterstützte Werkzeuge sind *autopep8* und *black*, mit jeweils eigenen Option:
    - *Autopep8-Stufe* - weitere Einzelheiten unter „Autopep8 advanced usage <https://pypi.org/project/autopep8/#more-advanced-usage>".
    - ☐ *Anführungszeichen normieren*: ersetzt alle einfachen Anführungszeichen durch doppelte Anführungszeichen, wenn möglich

- *Ausführen und Fehler suchen*
  - ☐ *Objektinspektor einschalten (Umschaltung zwischen Reitern kann langsam sein)*
  - ☐ *Skript vor Ausführung automatisch speichern*: Speichert das Skript automatisch bei der Ausführung. Bei dieser Aktion wird eine temporäre Datei (im Systemverzeichnis) gespeichert, die nach der Ausführung automatisch gelöscht wird.

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.

- *Vorbereitete API-Dateien benutzen*: Wenn aktiviert, wird die gewählte `*.pap`-Datei für die Code-Vervollständigung verwendet. Um eine vorbereitete API-Datei zu generieren, müssen Sie mindestens eine `*.api`-Datei laden und diese dann durch Anklicken der Schaltfläche *Compile APIs…* kompilieren.

Die Gruppe *Externer Editor* ermöglicht es Ihnen, Befehlszeilenanweisungen zum Starten eines externen Python-Code-Editors unter Angabe eines Dateinamens, einer Zeilen- und Spaltennummer zu geben. Wenn dieses Feld leer ist, wird der Standard-Systemeditor verwendet.

5.1.21 Verarbeitung

Die Registerkarte *Verarbeitung* enthält allgemeine Einstellungen für Werkzeuge und Datenanbieter, die im QGIS-Verarbeitungsumgebung verwendet werden. Weitere Informationen finden Sie unter *Konfigurieren der Verarbeitungsumgebung*.

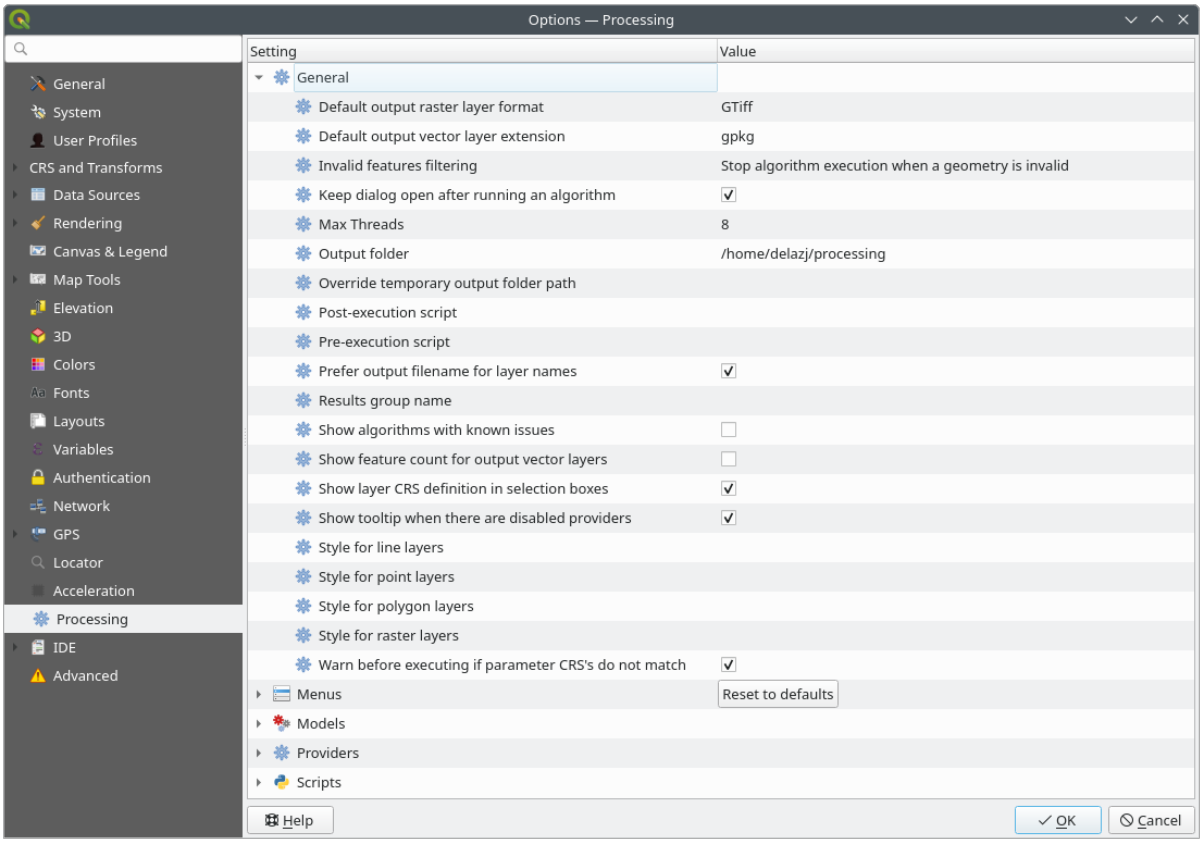


Abb. 5.29: Verarbeitung

5.1.22 Erweitert

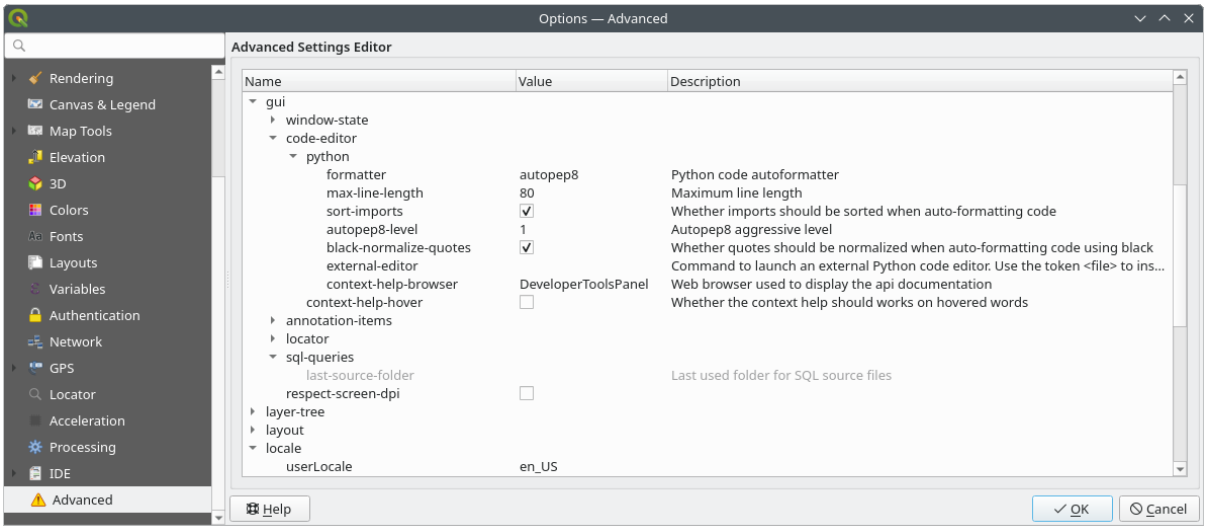


Abb. 5.30: Erweitert

Alle Einstellungen in Bezug auf QGIS (Benutzeroberfläche, Werkzeuge, Datenanbieter, Verarbeitungskonfigurationen, Standardwerte und -pfade, Plugin-Optionen, Ausdrücke, Geometrieprüfungen…) werden in der Datei `QGIS/QGIS3.ini` im Verzeichnis für das aktuelle *Benutzerprofil* gespeichert. Durch das Kopieren und Einfügen dieser Datei können die Einstellungen auf anderen Installationen übertragen werden.

Der Reiter *Erweitert* bietet die Möglichkeit, diese Einstellungen über die *Erweiterte Einstellungsbearbeitung* zu verwalten. Nachdem Sie bestätigt haben, vorsichtig zu sein, werden in dem Bereich alle vorhandenen Einstellungen angezeigt und Sie können deren Werte bearbeiten. Wenn Sie mit der rechten Maustaste auf eine Einstellung oder eine Gruppe klicken, können Sie sie löschen (um eine Einstellung oder Gruppe hinzuzufügen, müssen Sie die Datei `QGIS3.ini` bearbeiten). Änderungen werden automatisch in der Datei `QGIS3.ini` gespeichert.

Warnung

Verwenden Sie die Erweiterte Einstellungsbearbeitung nicht leichtfertig

Bitte seien Sie vorsichtig beim Ändern von Elementen in diesem Dialog, da die Änderungen sofort übernommen werden. Änderungen an Werten, über deren Funktion Sie sich nicht ganz sicher sind, können Ihre QGIS-Installation unbrauchbar machen!

5.2 Arbeiten mit Benutzerprofilen

5.2.1 Das Konzept

Das Menü *Einstellungen* ► *Benutzerprofile* bietet Funktionen zum Einrichten und Aufrufen von verschiedenen Benutzerprofilen. Ein Benutzerprofil ist eine standardisierte Anwendungskonfiguration, mit der die folgenden Einstellungen zusammen in einem Ordner gespeichert werden können:

- Alle Einstellungen *Optionen*, einschließlich Gebietsschema, Projektionen, Authentifizierungseinstellungen, Farbpaletten, Tastaturkürzel…
- GUI-Konfigurationen und :ref:`Anpassungen <sec_customization> `
- Grid-Dateien und andere Projektions-Hilfsdateien, die für die Datentransformation installiert würden
- Installierte *Plugins* und ihre Konfigurationen

- Projektvorlagen und der Verlauf der gespeicherten Projekte mit Bildvorschau
- *Verarbeitungs-Einstellungen*, Log-ins, Skripte, Modelle

Standardmäßig gibt es in QGIS nur ein Benutzerprofil namens `default`. Sie können aber so viele Benutzerprofile anlegen, wie Sie möchten:

1. Klicken Sie auf *Einstellungen* ► *Benutzerprofile* ► *Neues Benutzerprofil…*.
2. Sie werden aufgefordert, einen Profilnamen anzugeben; dadurch wird auch ein gleichnamiger Ordner unter `~/<UserProfiles>/` angelegt, wobei
   - ~ für das **HOME**-Verzeichnis steht, das unter Windows standardmäßig `C:\Users\ <username>` ist;
   - und `<UserProfiles>` steht für den Hauptordner der Profile, d.h.:
     - `.local/share/QGIS/QGIS3/profiles/`
     - `%AppData%\Roaming\QGIS\QGIS3\profiles\`
     - `Library/Application Support/QGIS/QGIS3/profiles/`

   Der Benutzerprofilordner kann von QGIS aus mit dem *Open Active Profile Folder* geöffnet werden.
3. Eine neue Instanz von QGIS mit der Standard Start-Konfiguration wird gestartet. Sie können dann Ihre eigenen Konfigurationen festlegen.

Wenn Sie mindestens ein weiteres Profil angelegt haben, wird der Name des jeweils aktiven Profils in der Titelleiste von QGIS in eckigen Klammern angezeigt.

Da für jedes Benutzerprofil die Einstellungen, Plugins und der Verlauf getrennt gespeichert werden, eignen sich Benutzerprofile sehr gut für unterschiedliche Arbeitsabläufe, Demos, Benutzer desselben Rechners, zum Testen von Einstellungen, usw. Sie können von einem zum anderen wechseln, indem Sie im Menü *Einstellungen* ► *Benutzerprofile* das entsprechende auswählen. Sie können QGIS auch mit einem bestimmten Benutzerprofil über die *Befehlszeile* starten.

Tipp

Führen Sie QGIS unter einem neuen Benutzerprofil aus, um die Fehlerpersistenz zu überprüfen

Fehler, die bei Funktionen in QGIS auftreten können, können mit Konflikten bei den Einstellungen zusammenhängen. Wenn Sie QGIS unter einem anderen Benutzerprofil ausführen, können Sie diese Fehler möglicherweise beheben oder die Ursache des Problems überprüfen. Im Allgemeinen empfiehlt es sich, QGIS unter einem neuen Benutzerprofil zu starten, um eine sauberere Konfiguration zu erhalten, und die Befehle erneut auszuführen

Wenn der Fehler Sie daran hindert, ein neues Benutzerprofil über das Menü *Einstellungen* ► *Benutzerprofile* zu erstellen, haben Sie folgende Möglichkeiten:

- Benennen Sie im Datei-Explorer den „beschädigten” Benutzerprofilordner im Ordner `QGIS3/profiles` um und starten Sie QGIS neu. Ein neues Standardbenutzerprofil wird erstellt und ausgeführt.
- Starten Sie QGIS über die Befehlszeile mit dem *Profilname-Argument*:

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

5.2.2 Benutzerprofil einstellen

Standardmäßig öffnet QGIS eine neue Sitzung mit dem Profil der letzten geschlossenen Sitzung. Dies kann, neben anderen Einstellungen, unter *Einstellungen* ► *Optionen* ► *Benutzerprofile* angepasst werden:

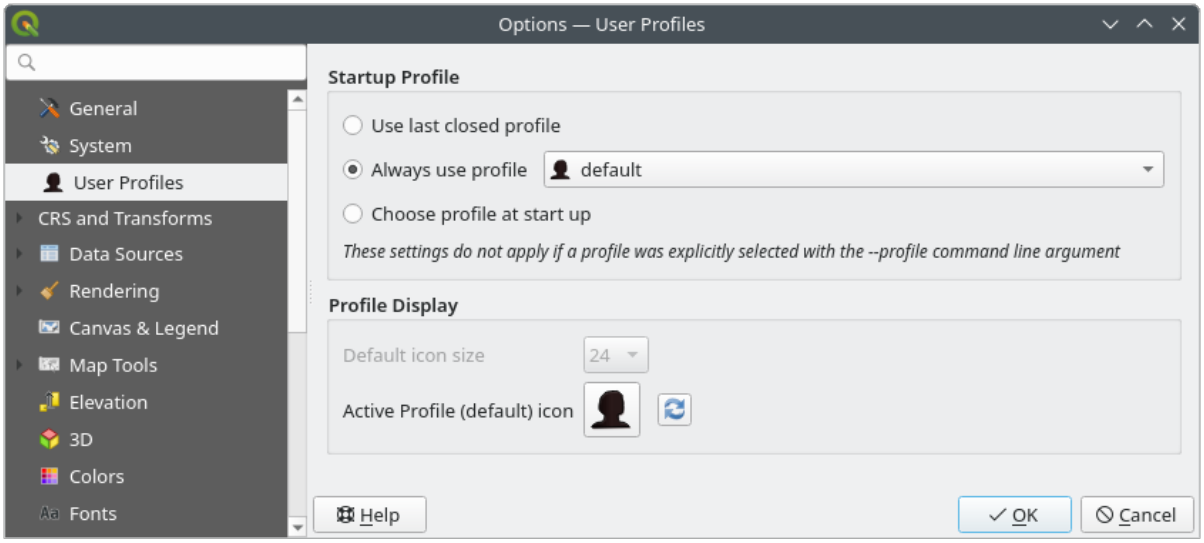


Abb. 5.31: Benutzerprofile-Reiter

- *Startprofil*: gibt das Benutzerprofil an, das beim Start einer QGIS-Sitzung verwendet werden soll. Die Optionen sind:
  - *Letztes gewähltes Profil verwenden*
  - ein bestimmtes Benutzerprofil, das aus dem Dropdown-Menü *Immer Profil verwenden* auszuwählen ist
  - *Profil beim Starten wählen*: Öffnet beim Start von QGIS das Dialogfeld *Benutzerprofilauswahl*, mit einer Liste der verfügbaren Benutzerprofile. Doppelklicken Sie auf einen Eintrag oder wählen Sie ein Profil und drücken Sie *OK*, um dieses Benutzerprofil für die Startsitzung zu verwenden. Mit *Neues Profil hinzufügen* können Sie ein Profil direkt neu erstellt und damit die Sitzung starten.
- Im Bereich *Profilanzeige* können Sie:
  - Die Größe der Symbole einstellen, die bei der Auswahl eines Profils im Dialogfeld *Benutzerprofilauswahl* angezeigt werden.
  - Ein Symbol auswählen, das neben dem jeweiligen Profil im Menü *Einstellungen* ► *Benutzerprofile* oder im Dialog *Benutzerprofilauswahl* angezeigt wird. Drücken Sie Icon zurücksetzen, um benutzerdefinierte Änderungen zu entfernen.

5.3 Projekteigenschaften

Im Eigenschaftsfenster des Projekts unter *Projekt* ► *Eigenschaften* können Sie projektspezifische Optionen einstellen. Die Einstellungen überschreiben ggf. ihre Entsprechung im Dialog *Optionen*.

5.3.1 Allgemein

Im Bereich *Allgemeine Einstellungen* des Reiters *Allgemein* haben Sie die folgenden Optionen:

- Den Speicherort der Projektdatei einsehen.
- Den Ordner für das *Projektheimatverzeichnis* festlegen. Der Pfad kann relativ zu dem Ordner der Projektdatei (geben Sie ihn ein) oder absolut sein. Das Projektverzeichnis kann zum Speichern von Daten und anderen für das Projekt nützlichen Inhalten verwendet werden. Dies ist praktisch, wenn Datensatz und Projektdateien nicht am selben Ort gespeichert sind. Wenn das Feld *Projektheimatverzeichnis* nicht ausgefüllt ist, wird standardmäßig der Ordner der Projektdatei verwendet.
- Einen Titel für das Projekt vergeben.
- Die *Auswahlfarbe* für markierte Objekte festlegen.
- Eine *Hintergrundfarbe* einstellen, die in der Kartenansicht verwendet wird.
- Sie können festlegen, ob der Pfad zu den Layern im Projekt als absoluter (vollständiger) oder als relativer Pfad zum Speicherort der Projektdatei gespeichert werden soll. Ein relativer Pfad ist z.B. dann sinnvoll, wenn Layer und/oder Projektdateien verschoben werden soll oder wenn der Zugriff auf das Projekt von verschiedenen Computern erfolgt.
- Sie können auswählen, dass versucht werden soll, Artefakte bei der Darstellung von Kartenkacheln zu vermeiden. Beachten Sie, dass die Aktivierung dieser Option zu Leistungseinbußen führen kann.
- *Attributtabellenfenster und Docks zwischen Sitzungen merken*: Wenn diese Option aktiviert ist, werden alle geöffneten Attributtabellen (als Fenster oder Dock) gespeichert und beim Laden des Projekts direkt wiederhergestellt. Dies kann Arbeitsabläufe erleichtern, wenn Sie in einem Projekt mit einer bestimmten Konfiguration der Attributtabellen arbeiten und das erneute Einrichten aufwendig wäre.

Die Berechnung von Flächen und Entfernungen ist eine häufige Aufgabe im GIS. Die ermittelten Werte sind aber abhängig von den zugrundeliegenden Einstellungen. Im Bereich *Messungen* können Sie diese Parameter steuern. Sie können auswählen:

- das *Ellipsoid*, auf dem alle Entfernungs-, Flächen- und Winkelberechnungen basieren; es kann sein:
  - **None/Planimetric**: die zurückgegebenen Werte sind in diesem Fall kartesische Maße. Diese Option kann als Standard für neue Projekte über das Menü *Einstellungen* ► *Optionen* ► *KBS Behandlung* gesetzt werden.
  - **Angepasst**: Sie müssen die Werte für die Hauptachse und die Nebenachse selbst festlegen.
  - oder aus einer Liste mit vordefinierten Ellipsoiden ausgewählt werden (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:
  - dem Aktualisierungsfeld in Attributtabellen
  - Feldrechner Berechnungen
  - *Objekte abfragen Werkzeug* : Werte für Länge, Umfang und Fläche
  - im *Messen Dialog*
  - bei *Dekoration mit einem Maßstab*

> **Bemerkung**
>
> Um sicherzustellen, dass die Maßeinheiten (für Entfernung und Fläche) immer die Einheiten des aktuellen KBS des Projekts widerspiegeln, wählen Sie für diese Optionen *Karteneinheiten*.

- 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.

    Bemerkung

    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

Bemerkung

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

Im Bereich *Koordinaten- und Kursanzeige* haben Sie die Möglichkeit, die Anzeige für die folgenden Werte anzupassen:

- Koordinaten, die in der QGIS-Statusleiste und im Bereich *Abgeleitet* in den Ergebnissen des Objekte abfragen Werkzeugs angezeigt werden,
- Peilungswerte, die in der Statusleiste oder bei Abfragen mit dem Kurs Messen Werkzeug angezeigt werden.

Folgende Parameter sind verfügbar:

- *Koordianten anzeigen in*:
  - `Karteneinheiten (Meter)`, basierend auf dem für das Projekt eingestellten KBS
  - „Kartengeografisch (Grad)“: basiert auf dem Projekt-KBS, wenn es sich um einen geografischen Typ handelt, andernfalls wird das zugehörige geografische KBS verwendet. Dies ist beispielsweise für nicht zur Erde gehörende Himmelskörper hilfreich.
  - `Benutzerprojektionseinheit`: ermöglicht es, jedes beliebige KBS für die Anzeige der Koordinaten zu verwenden
- Im Feld *Koordinaten KBS* wird, je nach oben gewähltem Modus, das KBS verwendete KBS angezeigt oder Sie können eins auswählen.

- Unter *Koordinatenformat* können Sie mit *Ändern…`einstellen, ob die Werte in ``Dezimalgrad`*, `Grad, Minuten` oder `Grad, Minuten, Sekunden` angezeigt werden sollen. Darüber hinaus haben sie folgende Optionen:
  - ☐ *Richtungsendung anzeigen*
  - ☐ *Führende Nullen für Minuten und Sekunden anzeigen*
  - ☐ *Führende Nullen bei Grad anzeigen*
  - ☐ *Nullen am Ende anzeigen*
- *Koordinatengenauigkeit*: die Anzahl der Dezimalstellen kann automatisch (abgeleitet vom Typ des KBS) oder manuell eingestellt werden
- *Koordinatenreihenfolge*: Sie können wählen, ob die Koordinaten in der nativen Reihenfolge des KBS (`Voreinstellung`) angezeigt werden sollen, in der Reihenfolge `Ostwert, Nordwert (Länge, Breite)` oder in der Reihenfolge `Nordwert, Ostwert (Breite, Länge)`.
- Unter *Kursformat* können Sie mit der Schaltfläche *Anpassen* einstellen, ob der Kurs von `0 bis 180° mit O/W-Zusatz`, von `-180 bis +180°` oder von `0 bis 360°` angezeigt wird. Darüber hinaus können sie Anzahl der *Dezimalstellen* festlegen und die Option *Nullen am Ende anzeigen* auswählen.

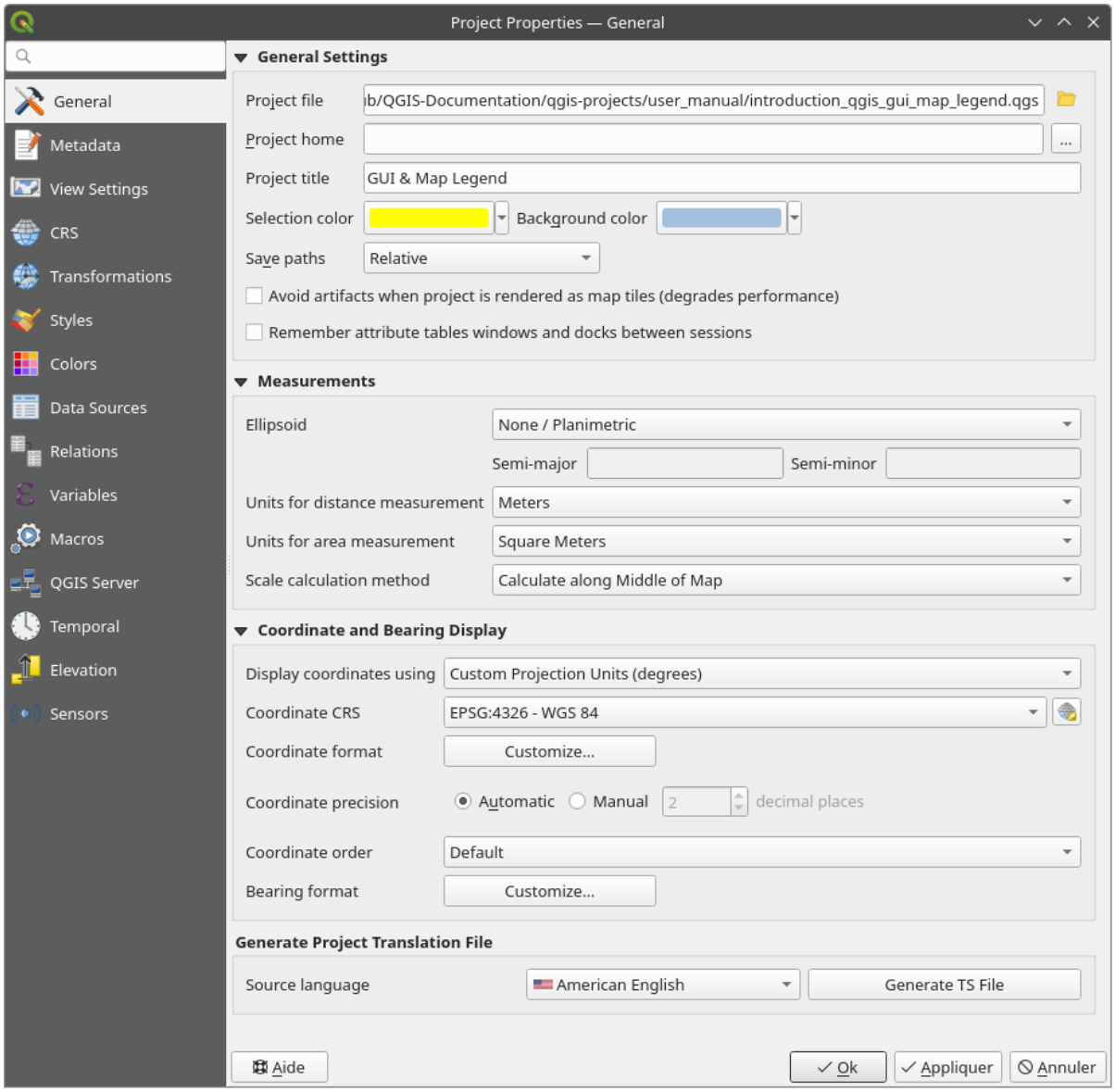


Abb. 5.32: Der Allgemein-Reiter der Projekteigenschaften

5.3.2 Metadaten

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

5.3.3 Ansichtseinstellungen

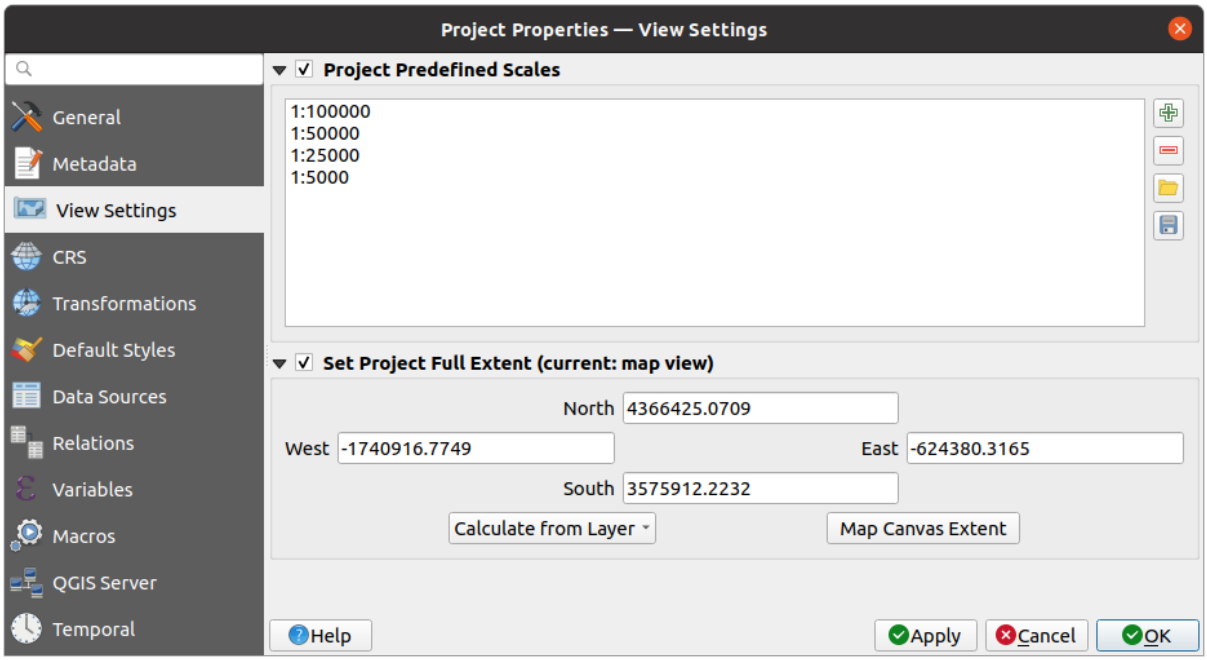


Abb. 5.33: Der Reiter Ansichtseinstellungen der Projekteigenschaften

Der Reiter *Ansichtseinstellungen* bietet Möglichkeiten zur Steuerung der Kartenansicht des Projekts. Sie haben die folgenden Optionen:

- *Vordefinierte Projektmaßstäbe*: eine Liste von Maßstäbe, die in Maßstabs-Drop-Down-Listen angezeigt werden sollen, wie z.B. im *Maßstab* der Statusleiste oder den Einstellungen einer zweiten Kartenansicht. Wenn hier *Vordefinierte Projektmaßstäbe* aktiviert ist, werden die globalen *vordefinierte Maßstäbe* überschrieben und nur die hier definierten angezeigt.
- *Volle Projektausmaße setzen*: diese Ausdehnung wird anstelle der Ausdehnung aller Layer beim Zoomen auf die volle Kartenausdehnung ( ) verwendet. Dies kann genutzt werden, wenn ein Projekt Web-Layer/nationale Layer/globale Layer enthält, das eigentliche Interessengebiet des Projekts jedoch ein kleineres geografisches Gebiet ist. Die Koordinaten für die volle Ausdehnung des Projekts können mit dem Widget extent selector 1 festgelegt werden.

5.3.4 KBS

> **Bemerkung**
>
> Für weitere Informationen darüber, wie QGIS mit Projektionen arbeitet, lesen Sie bitte den entsprechenden Abschnitt unter *Arbeiten mit Projektionen*.

Auf dem Reiter *KBS* können Sie das Koordinatenbezugssystem festlegen, das in diesem Projekt verwendet werden soll. Die Optionen sind:

- *Kein KBS (oder unbekannte/nicht-Erd-Projektion)*: Layer werden auf der Grundlage ihrer ursprünglichen, also nicht projizierten, Koordinaten gezeichnet
- Ein vorhandenes Koordinatenbezugssystem, das *geografisch*, *projiziert* oder *benutzerdefiniert* sein kann. Dem Projekt hinzugefügte Layer werden on-the-fly in dieses KBS übersetzt, um sie unabhängig von ihrem ursprünglichen KBS zu überlagern.

5.3.5 Transformationen

Auf dem Reiter „Transformationen“ können Sie die Einstellungen für die Reprojektion von Layern steuern, indem Sie die Datentransformationen festlegen, die für das aktuelle Projekt gelten sollen. Die Einstellungen hier haben Vorrang vor den entsprechenden globalen Einstellungen. Siehe *Datumstransformationen* für weitere Details.

5.3.6 Stile

Unter dem Reiter *Stile* können Sie Symbole und Farben konfigurieren, die spezifisch für das jeweilige Projekt genutzt werden.

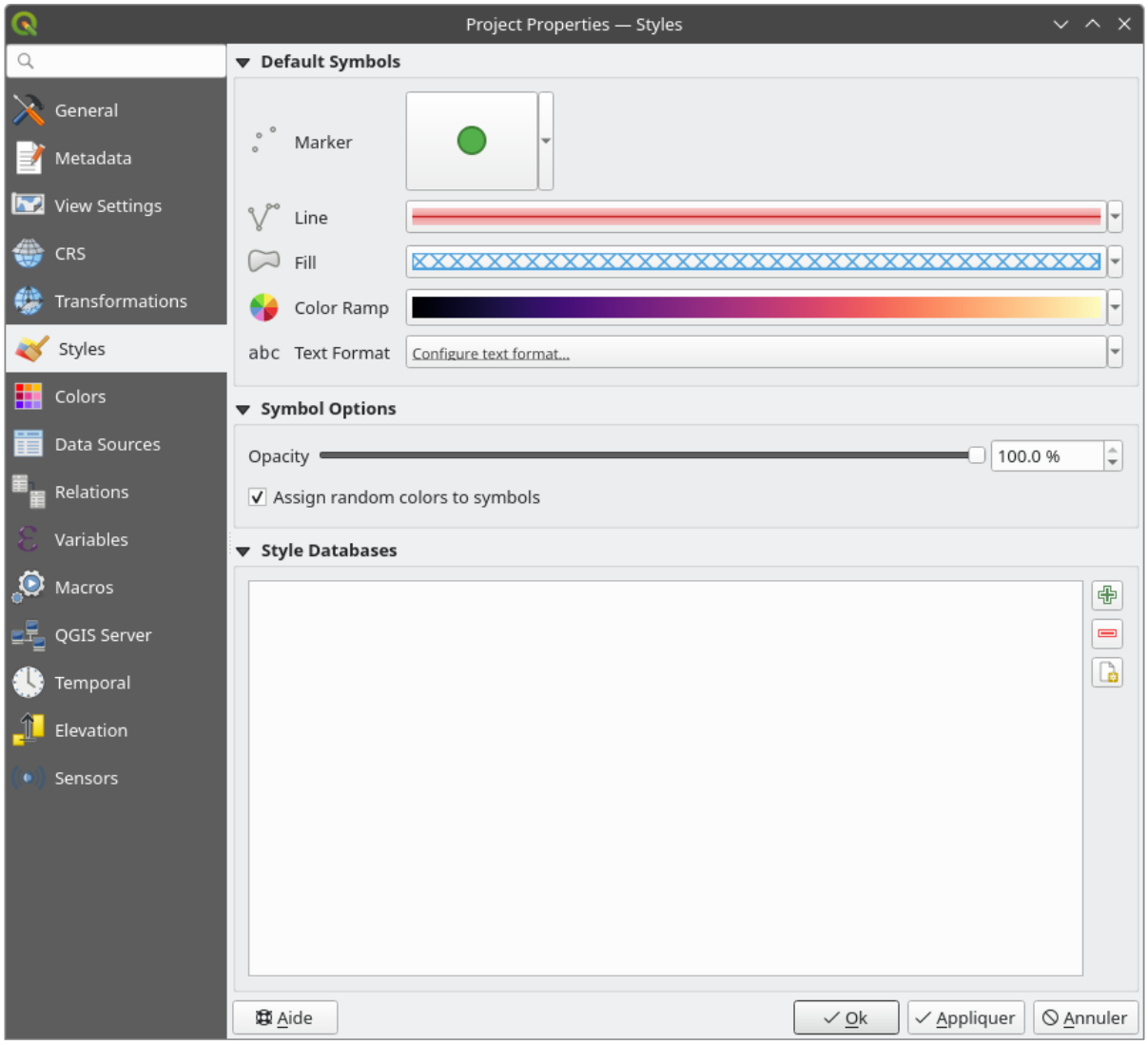


Abb. 5.34: Der Stile-Reiter

Vorlagesymbole

Mit der Gruppe *Vorlagesymbole* können Sie steuern, wie neue Layer im Projekt gezeichnet werden (wenn für sie kein Vorgabestil definiert ist). Sie können *Markierung*, *Linie* und *Füllung* je nach Geometrietyp der Ebene sowie die Standardwerte für *Farbverlauf* und *Textformat* (z. B. beim Aktivieren der Beschriftung) festlegen. Alle diese Elemente können über den Eintrag *Aktuelles Symbol leeren* im entsprechenden Dropdown-Widget zurückgesetzt werden.

Symboloptionen

In der Gruppe *Symboloptionen* können Sie:

- Die *Deckkraft* eines neu hinzugefügten Layers einstellen
- Das Kästchen *Den Symbolen zufällige Farben zuordnen* aktivieren, damit nicht alle Layer mit derselben Farbe dargestellt werden.

Stildatenbanken

Eine Stildatenbank in QGIS ist spezielle Datei zum Speichern von Symbolen, Textformaten und anderen Stilelementen. Sie dient als zentraler Ort, an dem Sie Ihre Symbologieressourcen effizient organisieren und verwalten können. Sie können z. B. eine eigene Stildatenbank für einen bestimmten Kunden erstellen, in der Symbole gespeichert werden, die auf die Bedürfnisse dieses Kunden zugeschnitten sind. Dies gewährleistet einen sauberen und organisierten Ansatz, ohne die Standard-Stil-Datenbank zu überladen. Oder in Mehrbenutzerumgebungen ist es möglich, projektspezifische Stile an einem gemeinsamen Ort zu speichern. Durch die Verknüpfung eines Projekts mit diesen Stilen erhalten alle Benutzer Zugriff auf eine gemeinsame Symbolik, wodurch der manuelle Import oder die Aktualisierung der einzelnen lokalen Stildatenbanken entfällt. Dieser Ansatz ermöglicht es, dass andere Benutzer, die an demselben Projekt arbeiten, sofort nach dem Laden des Projekts Zugang zu allen erforderlichen Symbolen haben.

Im Bereich *Stildatenbank* können Sie wählen zwischen *Hinzufügen* oder *Entfernen* einer Stil-Datenbank oder Sie können eine *Neue Stildatenbank erzeugen*. Wenn Sie in diesem Abschnitt eine Stildatenbank hinzufügen oder entfernen, werden die Änderungen automatisch in den *Symbology Properties* übernommen.

5.3.7 Farben

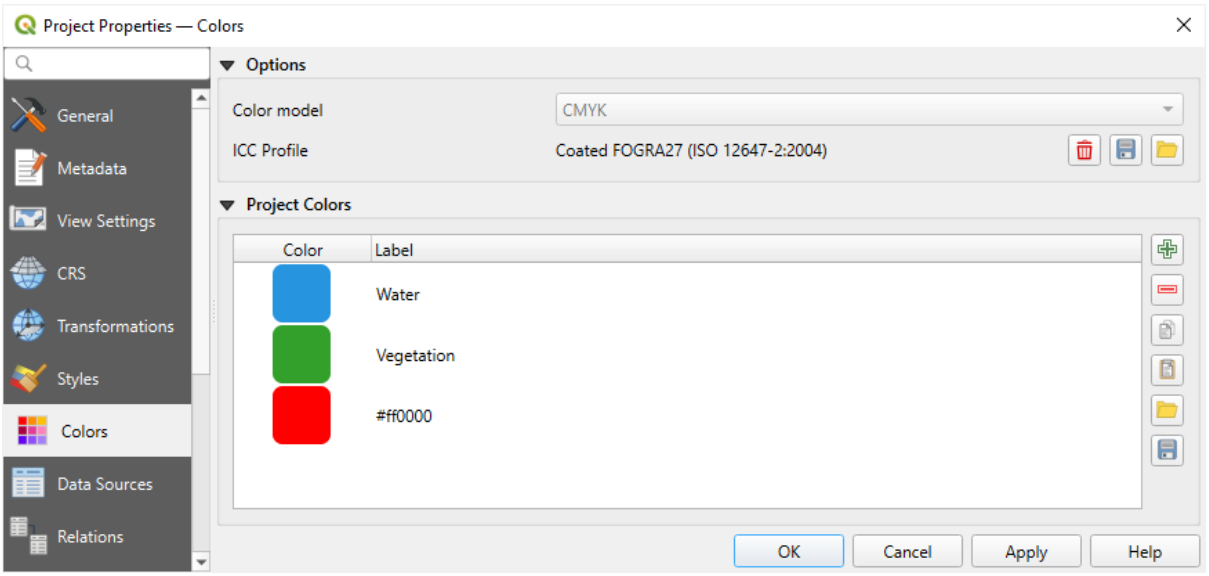


Abb. 5.35: Der Farben-Reiter

Auf dem Reiter *Farben* können Sie das *Farbmodell* auswählen, d. h. ob die Auswahl einer Farbe standardmäßig als `RGB` oder `CMYK` definiert werden soll. Alle Farben, die in einem anderen als dem hier angegebenen Farbmodell definiert sind, werden beim Exportieren eines Layouts in dieses Farbmodell konvertiert.

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.

Bemerkung

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

Es gibt auch einen zusätzlichen Abschnitt, in dem Sie spezifische Farben für das laufende Projekt definieren können. Wie bei den *globalen Einstellungen für Farben*, können Sie:

- Eine *Farbe hinzufügen* oder eine *Farbe entfernen*
- Eine *Farbe kopieren* oder eine *Farbe einfügen*
- Die ganze Farbpalette als `.gpl` Datei *Importieren* oder *Exportieren*.

Doppelklicken Sie auf eine Farbe in der Liste, um sie mit dem *Farbauswahl Dialog* zu ändern. Mit einem Doppelklick in die Spalte *Beschriftung* können Sie die Farbe auch benennen bzw. umbenennen.

Diese Liste wird als *Projektfarben* bezeichnet und bei der *Farbauswahl* als eigene Palette aufgeführt.

Tipp

Verwenden Sie Projektfarben, um Farbwidgets schnell zuzuweisen und zu aktualisieren

Projektfarben können über ihre Bezeichnung referenziert werden und die Farbwidgets, in denen sie verwendet werden, sind an sie gebunden. Das bedeutet, dass Sie nicht immer wieder dieselbe Farbe für viele Eigenschaften einstellen müssen, um eine umständliche Aktualisierung zu vermeiden:

1. Definieren Sie eine Farbe als Projektfarbe
2. Klicken Sie danach an einer beliebigen Stelle in QGIS, an der Sie diese Farbe verwenden wollen, auf das Feld *Datendefinierte Übersteuerung* am Ende der Farbauswahlzeile
3. Bewegen Sie dann den Mauszeiger über den Menüeintrag *Farbe* und wählen Sie Ihre definierte Projektfarbe aus. Der Eigenschaft wird dann der Ausdruck `project_color('Name_der_Farbe')` zugewiesen und die Farbauswahl zeigt die entsprechende Farbe an.
4. Wiederholen Sie die Schritte 2 und 3 für so viele Farben, wie Sie benötigen.
5. Wenn Sie jetzt die Projektfarbe in den Projekteigenschaften ändern, wird die Änderung an allen Stellen, an denen Sie die Farbe wie oben beschrieben festgelegt haben, automatisch übernommen.

5.3.8 Datenquellen

Auf dem Reiter *Datenquellen* stehen die folgenden Optionen zur Verfügung:

- *Transaktionsmodus*, definiert, wie Bearbeitungen an den Datenanbieter gesendet werden:
  - *Lokaler Bearbeitungspuffer*: Bearbeitungen werden lokal zwischengespeichert und an den Datenanbieter gesendet, wenn der Layer-Bearbeitungsmodus umgeschaltet oder auf *Layeränderungen speichern* geklickt wird.
  - *Automatische Transaktionsgruppen*: Bei unterstützten Datenquellen (Postgres- und GeoPackage-Datenbanken) wird der Bearbeitungsstatus aller Tabellen, die aus der gleichen Datenbank stammen, synchronisiert und in einer serverseitigen Transaktion ausgeführt. Anstatt Änderungen lokal zu puffern, werden sie direkt an eine Transaktion an die Datenbank gesendet und beim Umschalten des Layer-Bearbeitungsmodus oder beim Klicken auf *Layeränderungen speichern* nur noch bestätigt.
  - *Gepufferte Transaktionsgruppen*: alle bearbeitbaren Layer, egal von welchem Anbieter, werden synchron umgeschaltet und alle Bearbeitungen in einem lokalen Bearbeitungspuffer gespeichert. Das Speichern von Änderungen wird in einer einzigen Transaktion auf allen Layern (pro Anbieter) durchgeführt.

  Beachten Sie, dass Sie diese Option nur ändern können, wenn sich kein Layer im Projekt im Bearbeitungsmodus befindet.
- *Vorgabewerte auf Anbieterseite auswerten*: Beim Hinzufügen neuer Objekte in einer PostgreSQL-Tabelle werden Felder für die Vorgabewertebeschränkung bestehen („constraint“) beim Anlegen des Objekts ausgewertet und ausgefüllt, und nicht erst bei der Übergabe/Speicherung. Das bedeutet, dass z. B. im Attribut-Formular anstelle eines Ausdrucks wie `nextval('serial')` der zu erwartende Wert (z.B. `25`) anzeigt wird.
- *Bearbeitungsstatus zwischen Sitzungen merken*: Wenn diese Option aktiviert ist, werden alle Layer, die beim Schließen und Speichern des Projekts auf bearbeitbar gestellt waren, auch wieder in den Bearbeitungsmodus gesetzt, wenn das Projekt erneut geöffnet wird.
- Sie können die *Layereigenschaften* konfigurieren:
  - Legen Sie fest (oder deaktivieren Sie), welcher Layer „abfragbar“ ist, d. h. auf das *Identifizierungswerkzeug* reagiert. Standardmäßig sind Layer abfragbar eingestellt.
  - Legen Sie fest, ob ein Layer `schreibgeschützt` sein soll, was bedeutet, dass er vom Benutzer nicht bearbeitet werden kann, unabhängig von den Möglichkeiten des Datenanbieters. Auch wenn es sich hier um einen schwachen Schutz handelt, da er einfach wieder aufgehoben werden kann, bietet er doch eine schnelle und praktische Möglichkeit, um unbeabsichtigte Änderungen an den Daten zu verhindern.
  - Legen Sie fest, welcher Layer „durchsuchbar“ ist, d. h. mit dem *Suchen-Widget* abgefragt werden kann. Standardmäßig sind Layer durchsuchbar.
  - Legen Sie fest, welche Layer als `erforderlich` definiert sind. Markierte Layer in dieser Liste sind vor versehentlichem Entfernen aus dem Projekt geschützt.
  - Legen Sie fest, welcher Layer als „privat“ gekennzeichnet werden soll, d. h. im :guilabel:„Layer“-Bedienfeld ausgeblendet wird. Dies ist für Zusatzlayer gedacht (Grundkarte, Verknüpfungen, Nachschlagen von Wertbeziehungen, höchstwahrscheinlich nicht-räumliche Layer usw.), die Sie in einem Projekt weiterhin benötigen, die aber nicht die Legendenstruktur und andere Layerauswahl-Tools überladen sollen. Wenn sie als sichtbar eingestellt sind, werden sie weiterhin in der Kartenansicht angezeigt und in der Legende des Drucklayouts gerendert. Verwenden Sie die Option :menuselection:„Legende filtern –> Private Ebenen anzeigen“ in der oberen Symbolleiste des :guilabel:„Layer“-Bedienfelds, um sie vorübergehend für Interaktionen zu aktivieren

  Darüber hinaus bietet die Tabelle der *Layereigenschaften* einige nützliche Möglichkeiten:

  - Sie können mehrere Zellen (nicht Zeilen!) markieren, indem Sie einfach in den Bereich hinter das Kontrollkästchen klicken; wenn Sie jetzt unten auf *Auswahl umschalten* klicken, wird der Status aller ausgewählten Kontrollkästchen geändert;

  - Sie können sich in der Liste *Nur räumliche Layer anzeigen* lassen;
  - Mit der Funktion *Layer filtern…* können Sie schnell einen gewünschten Layer finden.

- Im Bereich *Erweiterte Einstellungen* können Sie die Option *Projekt vertrauen, wenn Datenquelle keine Metadaten hat* auswählen (standardmäßig nicht markiert), um das Laden von Projekten zu beschleunigen, indem Datenprüfungen übersprungen werden. Nützlich im QGIS Server Kontext oder in Projekten mit großen Datenbankansichten/materialisierten Ansichten. Die Ausdehnung der Layer wird aus der QGIS Projektdatei gelesen (statt aus den Datenquellen) und bei Verwendung des PostgreSQL Providers wird die Primärschlüssel-Unität für Views und Materialized Views nicht geprüft.

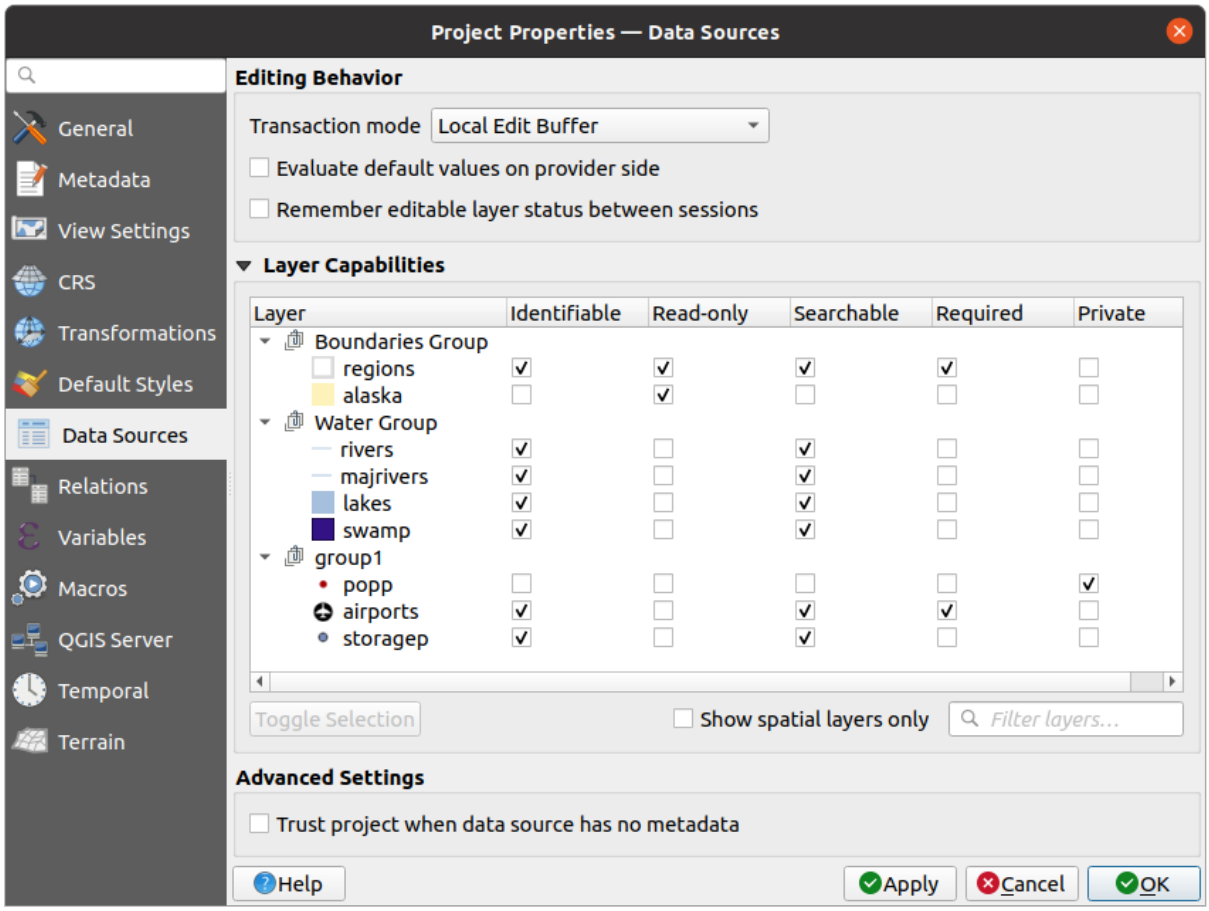


Abb. 5.36: Der Datenquellen Reiter

5.3.9 Beziehungen

Die Registerkarte *Beziehungen* dient zur Definition von Beziehungen zwischen Layern. Die Beziehungen können vom Typ 1:1, viele-zu-viele oder polymorph sein. Sie werden im Dialogfeld „Projekteigenschaften“ definiert. Sobald Beziehungen für eine Ebene existieren, listet ein neues Element der Benutzeroberfläche in der Formularansicht (z. B. beim Identifizieren eines Elements und Öffnen seines Formulars) die zugehörigen Entitäten auf. Dies bietet eine leistungsstarke Möglichkeit, um beispielsweise die Inspektionshistorie einer Rohrleitung oder eines Straßenabschnitts darzustellen. Weitere Informationen zur Unterstützung von Beziehungen finden Sie im Abschnitt *Festlegung von Beziehungen zwischen mehreren Layern*.

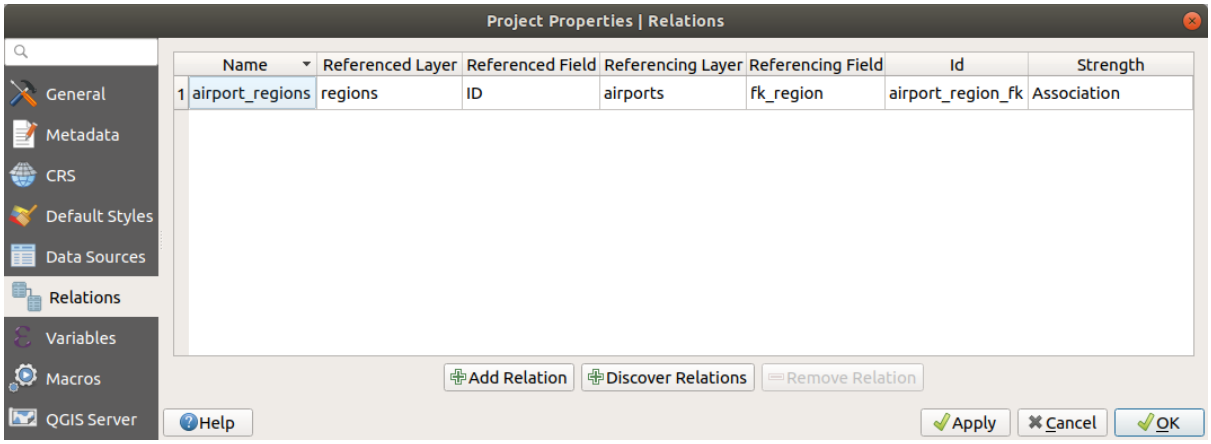


Abb. 5.37: Der Beziehungen-Reiter

5.3.10 Variablen

Der Reiter *Variablen* listet alle auf Projektebene verfügbaren Variablen auf (einschließlich aller globaler Variablen). Darüber hinaus können sie hier mit der Schaltfläche eine neue, benutzerdefinierte Variable hinzuzufügen. Wenn Sie eine benutzerdefinierte Variable aus der Liste auswählen, können Sie diese mit der Schaltfläche wieder entfernen. Weitere Informationen zur Verwendung von Variablen finden Sie im Kapitel Allgemeine Werkzeuge im Abschnitt *Speichern von Werten in Variablen*.

5.3.11 Makros

Der Reiter *Makros* dient der Bearbeitung von Python-Makros für das Projekt. Bei neuen Projekten sind standardmäßig drei Makros angelegt, die mit Inhalten gefüllt werden können: `openProject()`, `aveProject()` und `closeProject()`.

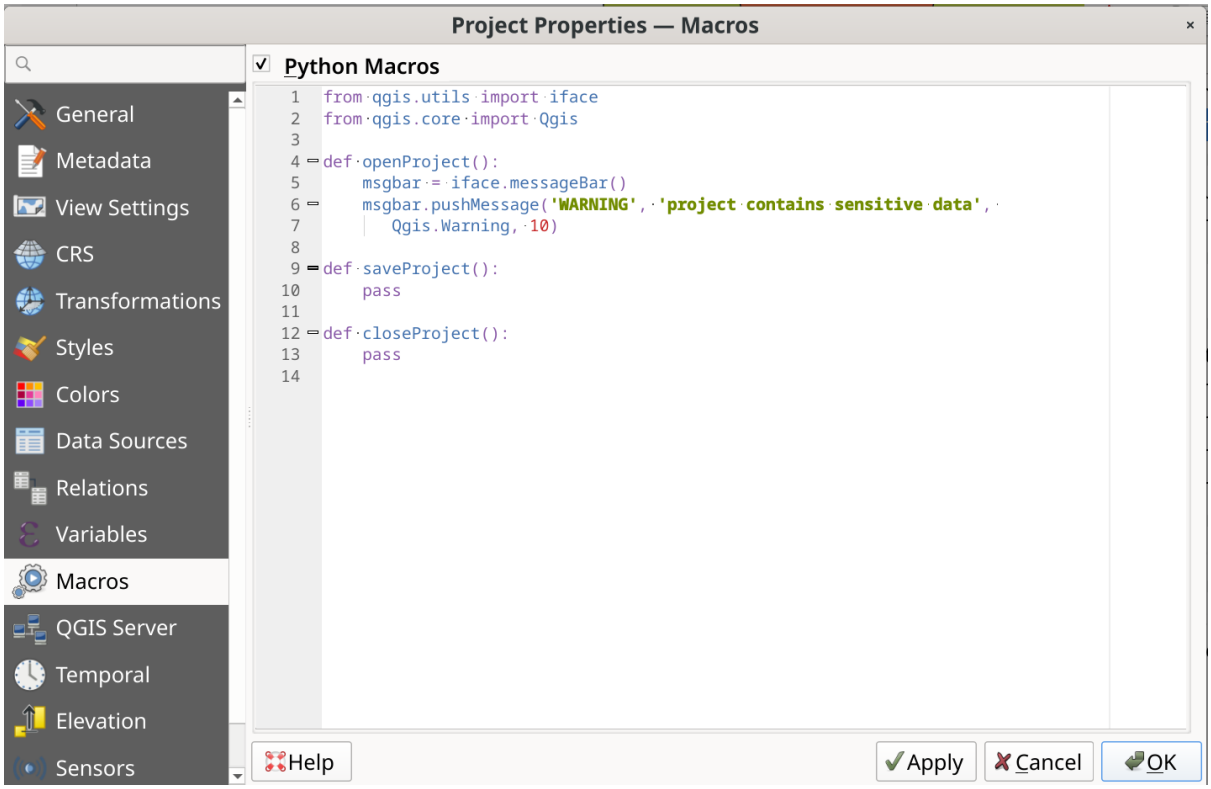


Abb. 5.38: Der Makro Reiter

5.3.12 QGIS Server

Der Reiter *QGIS Server* ermöglicht es Ihnen, Ihr Projekt zu konfigurieren, wenn es online veröffentlicht werden soll. Hier können Sie Informationen über die WMS- und WFS-Fähigkeiten des QGIS-Servers, den Umfang und die KBS-Einschränkungen festlegen. Weitere Informationen finden Sie im QGIS Server Handbuch im Abschnitt Erstellen von WMS aus dem Projekt und nachfolgende.

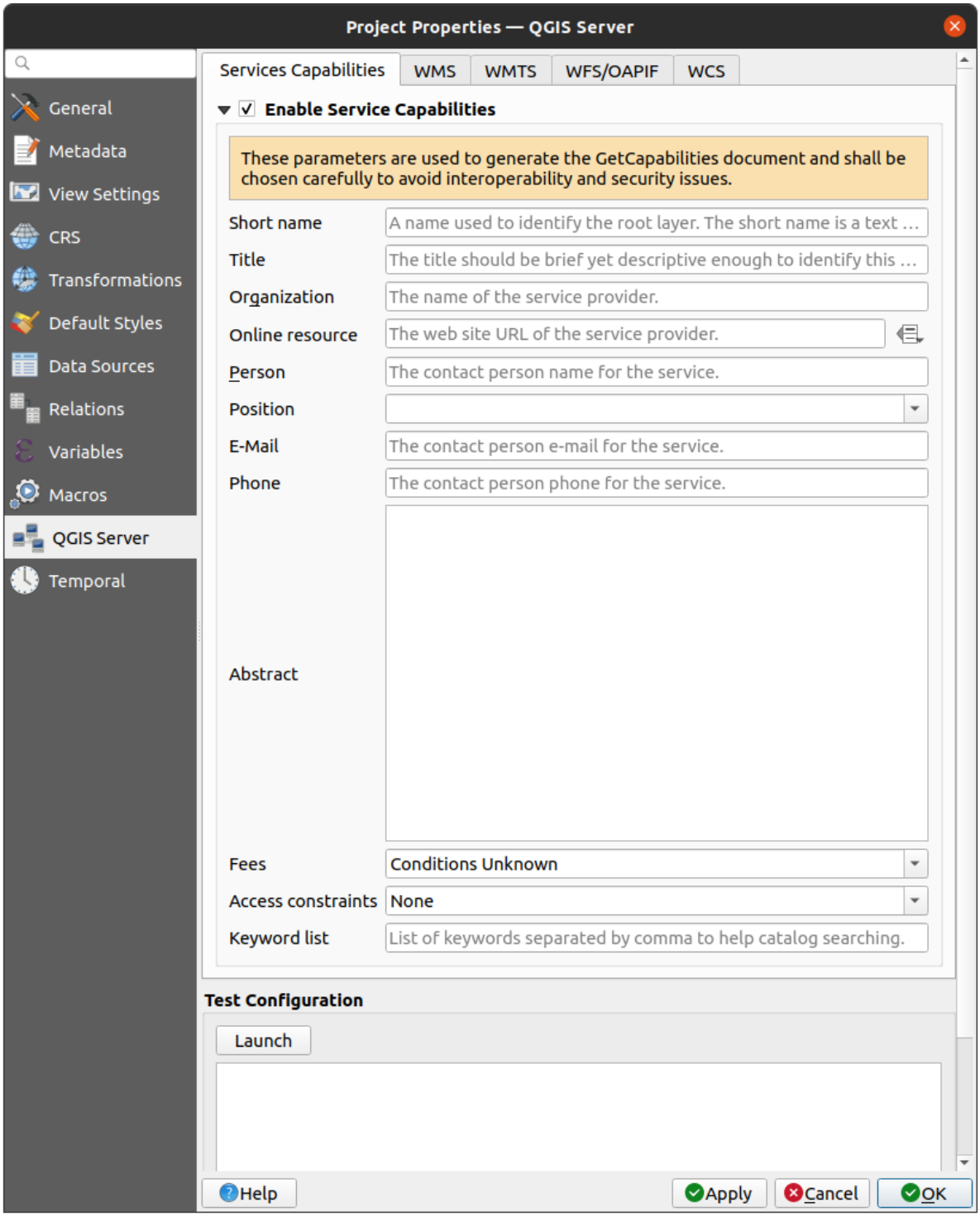


Abb. 5.39: Der QGIS Server Reiter

5.3.13 Zeitlich

Auf dem Reiter *Zeitlich* können Sie einen zeitlichen Bereich für Ihr Projekt festzulegen, entweder durch manuelle Eingabe von einem *Startdatum* und einem *Enddatum* oder durch Analysieren der zeitlichen Layer des Projekts mit *Aus Layern berechnen*. Der Zeitbereich des Projekts kann dann im Bedienfeld *Zeitsteuerung* verwendet werden, um die Kartenansicht zeitlich darzustellen; s. auch <maptimecontrol> .

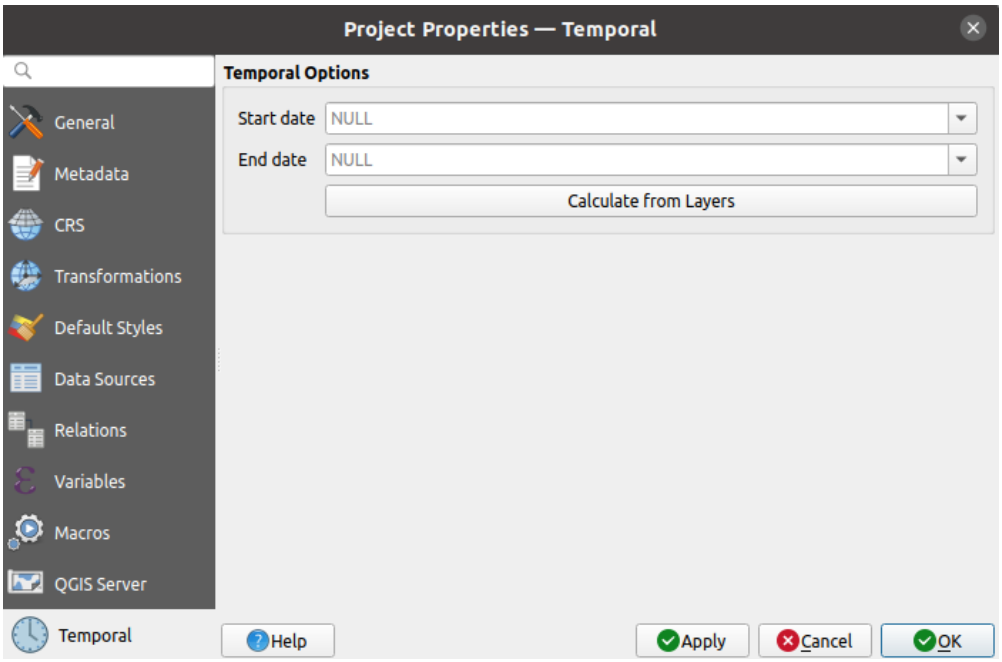


Abb. 5.40: Der Zeitlich-Reiter

5.3.14 Geländehöhe

Die Registerkarte *Geländehöhe* hilft Ihnen bei der Konfiguration der Standardeinstellungen für das Gelände und die Höhe. Wenn eine neue *3D-Karte* im Projekt erstellt wird, verwendet die Karte standardmäßig dieselben Geländeeinstellungen, die für das Projekt definiert sind. Die Höheneinstellungen des Projekts werden auch beim *Geländehöhenprofil* berücksichtigt.

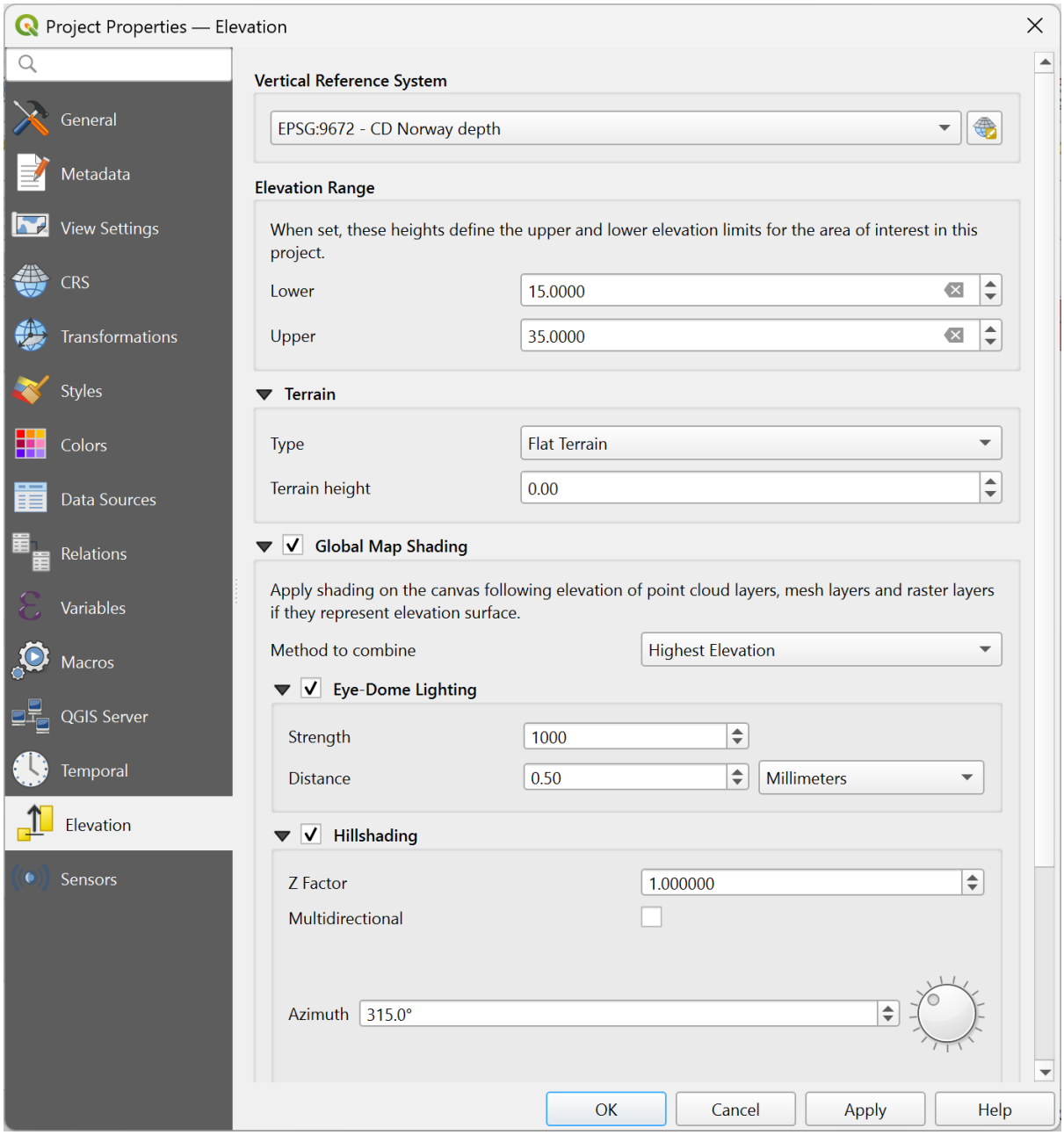


Abb. 5.41: Der Geländehöhen-Reiter

- *Vertikales Bezugssystem*: Wenn das KBS Ihres Projekts ein dreidimensionales ist, also auch eine Z-Komponente enthält, dann ist auch das für das Projekt verwendete vertikale KBS diese Z-Komponente. Wenn Ihr Projekt-KBS horizontal ist, können Sie ein bestimmtes vertikales KBS auswählen, indem Sie auf KBS wählen klicken oder eines aus der Favoritenliste auswählen.
- *Geländehöhenbereich*: Hilft Ihnen dabei, die *Untere* und *Obere* Grenze für die *Höhensteuerung* im Projekt festzulegen.
- Im Bereich *Gelände* können sie zwischen den folgenden Typen wählen:
  - *Flaches Gelände* mit der Einstellung für die *Geländehöhe*;
  - *DHM (Rasterlayer)* mit der Möglichkeit den entsprechenden *DHM-Layer* auszuwählen, einen *Vertikal Maßstab* für die Bandwerte und einen vertikalen *Versatz* anzugeben;
  - *Netz* mit der Möglichkeit, den entsprechenden *Netzlayer* auszuwählen, einen *Vertikalen Maßstab*, der auf den Z-Wert der Eckpunkte angewendet wird und einen vertikalen *Versatz* anzugeben.

Diese Einstellungen können mit dem *Konfigurieren…* Dialog einer 3D-Kartenansicht überschrieben werden.

- Mit der Einstellung *Globale Kartenschattierung* legen Sie für Ihr Projekt einen Schattierungseffekt fest, der auf der Höhe aller aktiven Layer der folgenden Typen basiert:
  - *Raster-Layer*
  - *Netz-Layer*
  - *Punktwolken-Layer*

  Methode zur Kombination: um den Höhenwert an einer bestimmten Position zu ermitteln, werden alle relevanten Layer berücksichtigt, wobei einer der folgenden Möglichkeiten gewählt werden kann:
  - *Höchste Erhebung*: der höchste Wert aller Layer wird verwendet;
  - *Basierend auf der Layerreihenfolge*: die Höhe auf dem obersten Layer im Layer-Baum wird verwendet.

  Unterstützte Schattierungsoptionen sind:
  - *Eye-dome-Lighting*: wendet Schattierungseffekte auf die Kartenansicht an, um eine bessere Tiefenwiedergabe zu erreichen. Die folgenden Parameter können gesteuert werden:
    - *Stärke*: erhöht den Kontrast und ermöglicht eine bessere Tiefenwahrnehmung
    - *Abstand*: stellt den Abstand der verwendeten Pixel vom mittleren Pixel dar und bewirkt, dass die Ränder dicker werden.
  - *Schummerung* generiert Reliefs auf der Karte mithilfe von Schattierungen (Graustufen):
    - *Z-Faktor*: Skalierungsfaktor für den Höhenwert
    - *Mehrgerichtet*: Sie können festlegen, ob eine multidirektionale Schattierung verwendet werden soll.
    - *Azimut*: Der Azimut der fiktiven Lichtquelle
    - *Höhenlage*: Der Neigungswinkel der fiktiven Lichtquelle

> **Bemerkung**
>
> Über das Bedienfeld *Layergestaltung* steht eine Verknüpfung zu den Eigenschaften von *Globale Kartenschattierung* zur Verfügung.

5.3.15 Sensoren

Der Reiter *Sensoren* dient dazu, Sensoren zu konfigurieren und ihren Verbindungsstatus umzuschalten. Wenn sie aktiv sind, sammeln die Sensoren passiv Daten im Hintergrund und stellen ihre neuesten Daten für Ausdrücke und Python-Skripte zur Verfügung.

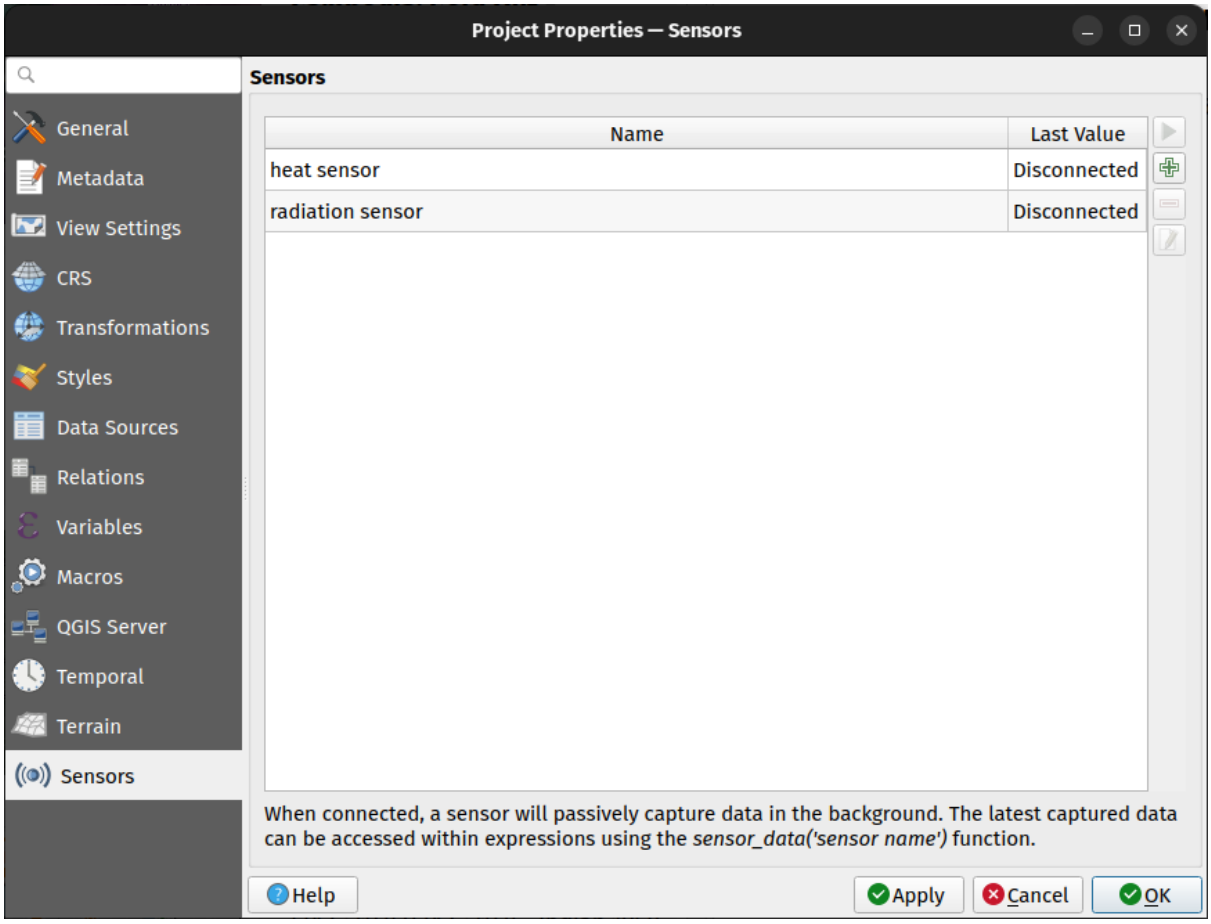


Abb. 5.42: Der Sensor Reiter

Um einen neuen Sensor hinzuzufügen, klicken Sie auf die Schaltfläche. Es öffnet sich ein Bedienfeld, in dem Sie die folgenden Einstellungen vornehmen können:

- *Name* des Sensors: wird verwendet, um Sensorwerte in Ausdrücken und Python-Skripten abzurufen;
- *Sensortyp*: TCP, UDP oder serielle Schnittstelle;
- zusätzliche typspezifische Details (z. B. Hostname und Port).

  Für Sensoren mit serieller Schnittstelle stehen Kombinationsfelder mit den verfügbaren seriellen Schnittstellen des Systems und der Verbindungsbaudrate zur Auswahl. Ein *Data frame delimiter* kann zum Parsen der zurückgegebenen Daten verwendet werden. Wenn im letzten vom seriellen Port übertragenen Datenblock mehrere Datenframes gefunden werden, werden alle Datenframes zurückgegeben. Sie können dann einen Ausdruck verwenden, um den zurückgegebenen letzten Datenwert des Sensors in einzelne Frames aufzuteilen (z. B. mit der Funktion *string_to_array*).

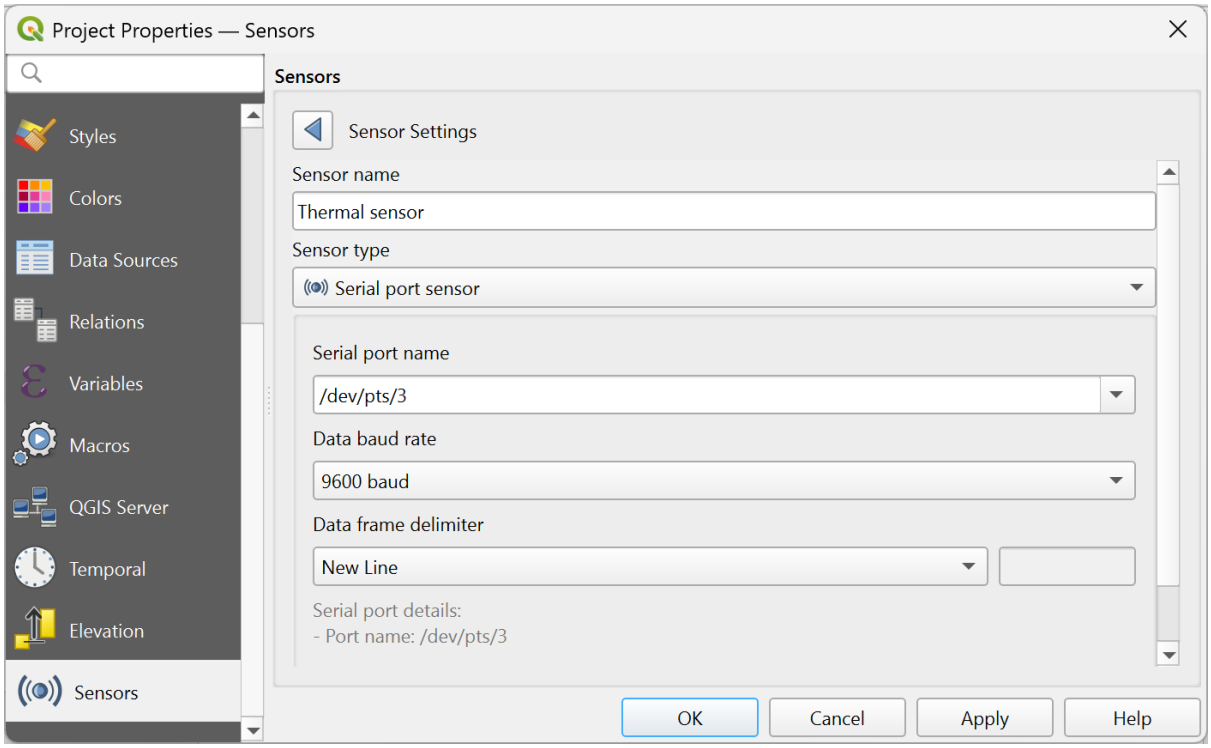


Abb. 5.43: Das Sensor-Einstellungen Fenster

Sobald ein Sensor konfiguriert ist, können Sie mit der Schaltfläche Sensorverbindungsstatus umschalten die Verbindung mit dem Sensor herstellen. Wenn die Verbindung aktiv ist, werden die zuletzt erfassten Daten in der Spalte *Letzter Wert* der Sensortabelle angezeigt.

5.4 Anpassung

Mit dem Dialog *Oberflächenanpassung* im Menü *Einstellungen* können Sie fast jedes Element der QGIS-Benutzeroberfläche (de)aktivieren. Dies kann sehr nützlich sein, wenn Sie Ihren Endbenutzern eine „Light“-Version von QGIS zur Verfügung stellen möchten, die nur die Symbole, Menüs oder Bedienfelder enthält, die sie benötigen.

Bemerkung

Bevor die Änderungen an der Benutzeroberfläche übernommen werden, müssen Sie QGIS neu starten.

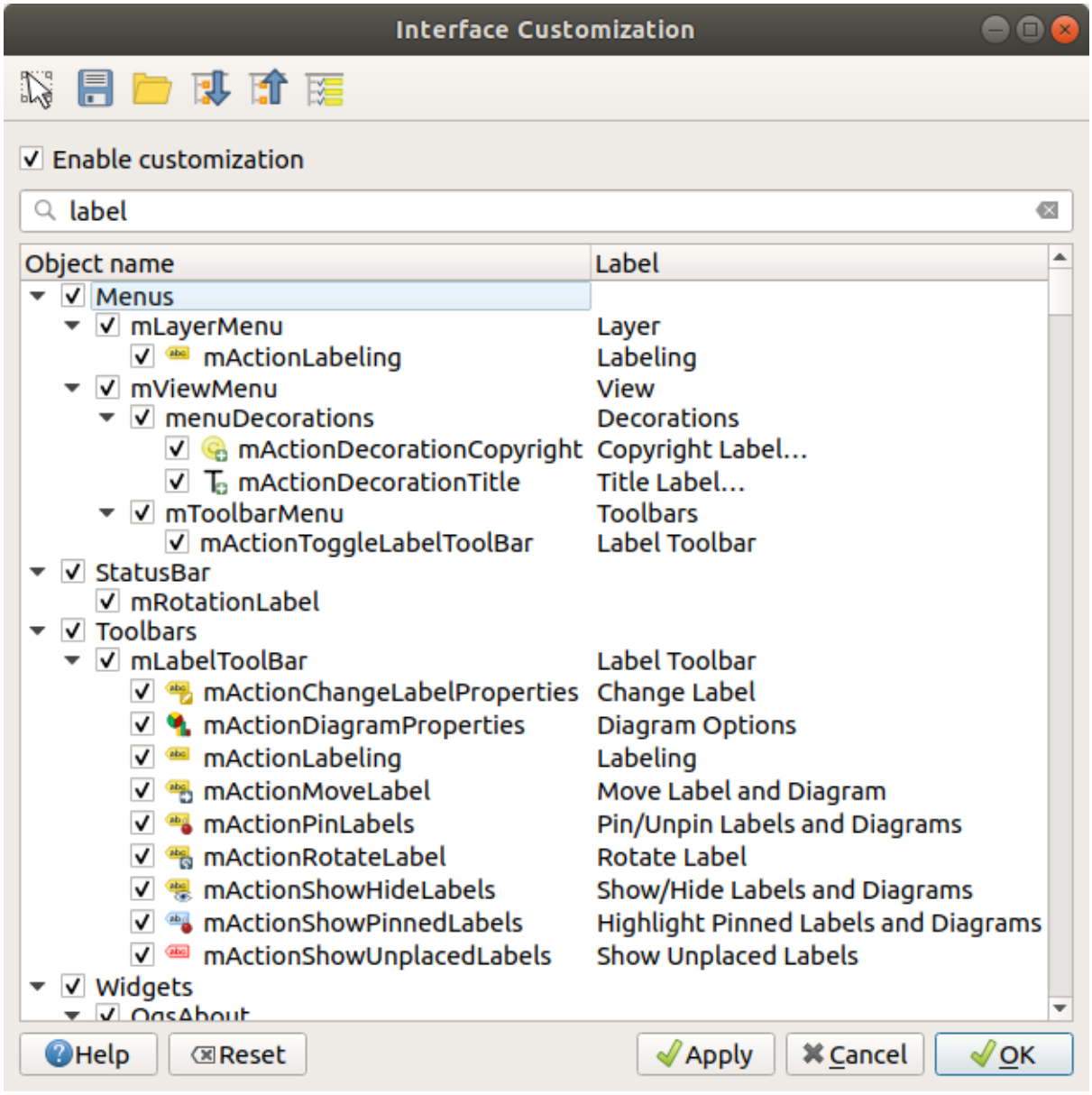


Abb. 5.44: Der Anpassung Dialog

Bevor Sie Änderungen an der Oberfläche vornehmen können, müssen Sie *Anpassung aktivieren* anklicken. Dies aktiviert die Werkzeugleiste und das Widget-Panel, hier können Sie einige QGIS Elemente unsichtbar schalten.

Die deaktivierbaren Elemente können sein:

- aus dem **Menu** ein kompletter Eintrag oder ein Untermenü (s. *Menüleiste*);
- ein komplettes **Panel** (s. *Bedienfelder und Werkzeugkästen*);
- die **Statusleiste** oder Teile davon (s. *Statusleiste*);
- im **Browser** angezeigte Dateitypen;
- ein beliebiges **Element** („Widget“) aus einem beliebigen QGIS Dialog: Beschriftung, Knöpfe, Komboboxen…

Mit Umschalten um Bedienelemente in der Hauptanwendung zu fangen können Sie auf ein Element in der QGIS-Oberfläche klicken, das ausgeblendet werden soll, und QGIS deaktiviert automatisch die Markierung des entsprechenden Eintrags im Anpassungsdialog. Sie können auch das Feld *Suchen* verwenden, um Elemente anhand ihres Namens oder ihrer Bezeichnung zu finden.

Wenn Sie Ihre Einstellungen vorgenommen haben, klicken Sie auf *Anwenden* oder *OK*, um Ihre Änderungen zu bestätigen. Die Einstellungen werden ab dem nächsten Start von QGIS standardmäßig verwendet.

Die Modifikationen können mit Als Datei speichern in einer `.ini` Datei gesichert werden. Mit dieser Option kann auch eine bestimmte Konfiguration für mehrere Benutzer eingerichtet werden. Mit Aus Datei laden kann die``.ini`` Datei auf anderen Rechnern importiert werden. Sie können auch verschiedene Einstellungen für verschiedene Anwendungsfälle speichern und diese dann beim Start von QGIS über die *Kommandozeile aufrufen*

Tipp

Standardeinstellungen einfach wiederherstellen

Mit einer der unten aufgeführten Methoden kann die standardmäßige QGIS GUI-Konfiguration wiederhergestellt werden:

- deaktivieren Sie *Anpassung aktivieren* oder klicken Sie auf Alle anhaken;
- drücken Sie im Menü *Einstellungen* ► *Optionen* auf dem Reiter *System* im Bereich **Einstellungen** auf die *Zurücksetzen* Schaltfläche;
- starten Sie QGIS über die Eingabeaufforderung mit dem Befehl `qgis --nocustomization`;
- setzen Sie im Menü *Einstellungen* ► *Optionen* auf dem Reiter *Erweitert* den Wert der Variablen *UI* ► *Customization* ► *Enabled* auf `false` (Beachten Sie die *Warnung*).

In den meisten Fällen müssen Sie QGIS neu starten, damit die Änderung übernommen werden.

5.5 Tastenkürzel

QGIS bietet Standard-Tastaturkürzel für viele Funktionen, Sie finden diese im Abschnitt *Menüleiste*. Über den Menüpunkt *Einstellungen* ► *Tastenkürzel…* können Sie die Standard-Tastaturkürzel ändern und neue hinzufügen.

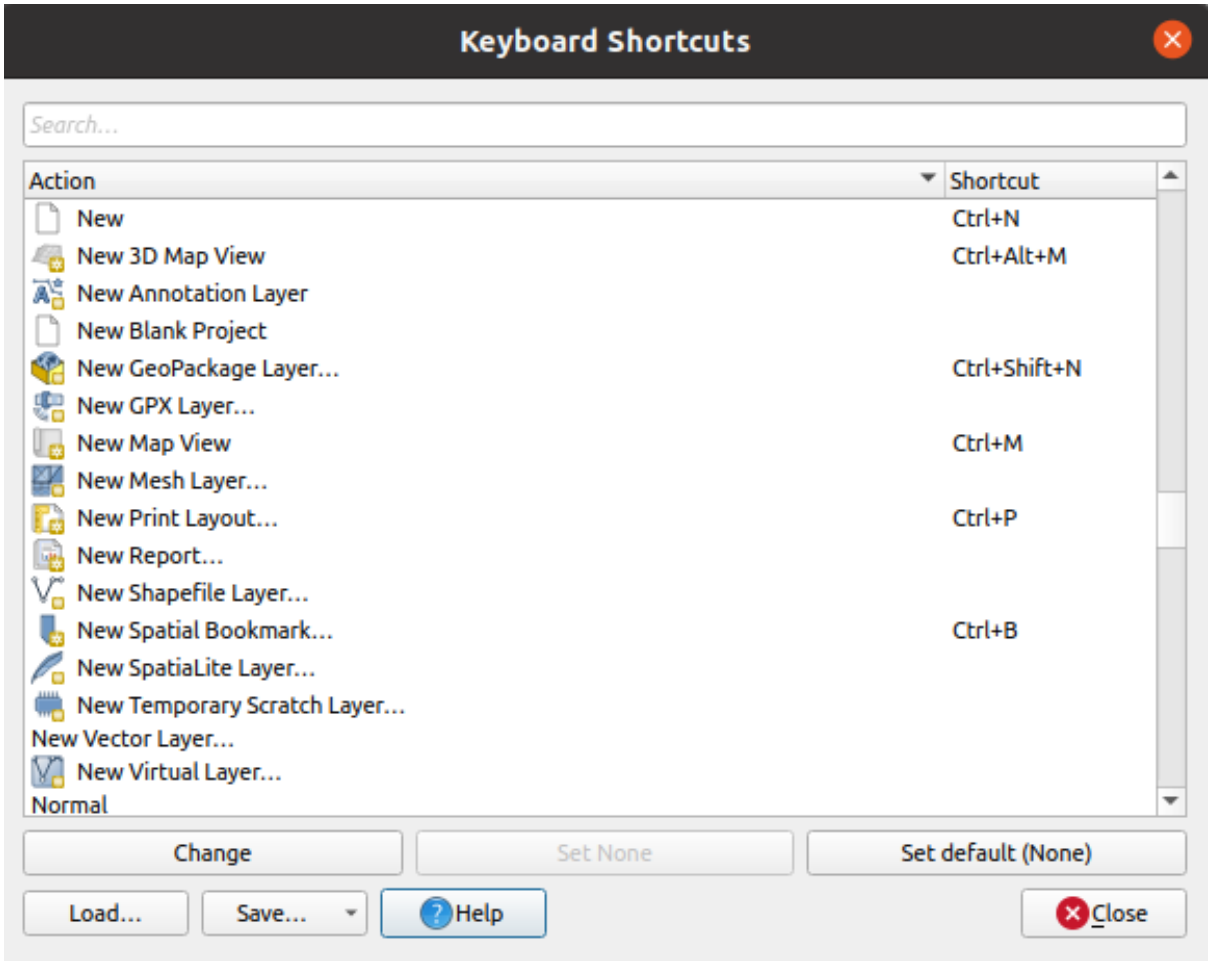


Abb. 5.45: Das Fenster der Tastenkürzel

Die Konfiguration ist sehr einfach. Verwenden Sie das Suchfeld oben im Dialogfeld, um eine bestimmte Aktion zu finden, wählen Sie diese in der Liste aus und klicken dann auf:

- *Ändern* und drücken Sie die neue Kombination, die Sie als neuen Shortcut zuweisen möchten;
- *Leer setzen* um eine zugewiesene Verknüpfung zu löschen;
- oder *Voreinstellung* um die Verknüpfung auf ihren ursprünglichen Standardwert zurückzusetzen.

Verfahren Sie wie oben für alle anderen Aktionen, die Sie anpassen möchten. Wenn Sie Ihre Konfiguration abgeschlossen haben, schließen Sie einfach den Dialog, um die Änderungen zu übernehmen. Sie können die Änderungen auch *Speichern…*, entweder als `.XML`-Datei nur mit den benutzerdefinierten Tastenkürzeln oder mit allen Tastenkürzeln oder als `.PDF`-Datei mit allen Tastenkürzeln. Ebenso können sie benutzerdefinierte Einstellungen aus einer Datei *Laden…*.

5.6 QGIS mit erweiterten Einstellungen ausführen

5.6.1 Eingabeaufforderung und Umgebungsvariablen

Wir haben gesehen, dass der *Start von QGIS* wie bei jeder anderen Anwendung auf Ihrem Betriebssystem funktioniert. QGIS bietet Befehlszeilenoptionen für fortgeschrittenere Anwendungsfälle (in einigen Fällen können Sie anstelle der Befehlszeilenoption eine Umgebungsvariable verwenden). Um eine Liste der Optionen zu erhalten, geben Sie `qgis --help` in die Befehlszeile ein, woraufhin Folgendes angezeigt wird:

```
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:
    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
```

Tipp

Beispiel für die Verwendung von Kommandozeilen-Argumenten

Sie können eine oder mehrere Dateien direkt beim Start von QGIS laden, indem Sie diese in der Kommandozeile angeben. Wenn Sie sich zum Beispiel im Verzeichnis `qgis_sample_data` befinden, können Sie QGIS mit einem Vektor-Layer und einer Rasterdatei starten, indem Sie den folgenden Befehl verwenden: `qgis ./raster/landcover.img ./gml/lakes.gml`

`--version`

Diese Option liefert QGIS Versionsinformationen.

`--snapshot`

Mit dieser Option können Sie ein Bildschirmfoto im PNG-Format von der aktuellen Ansicht erstellen. Dies ist z.B. praktisch, wenn Sie Bildschirmfotos desselben Projekts mit verschiedenen Daten erstellen müssen.

Ohne weitere Angaben wird eine PNG-Datei mit 800x600 Pixeln erzeugt. Die Größe kann mit den Argumenten `--width` und `--height` angepasst werden. Der Dateiname kann nach `--snapshot` hinzugefügt werden. Zum Beispiel:

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

`--width`

Diese Option gibt die Breite des auszugebenden Schnappschusses an (wird zusammen mit `-snapshot` verwendet`).

`--height`

Diese Option gibt die Höhe des auszugebenden Schnappschusses an (wird zusammen mit `-snapshot` verwendet).

`--lang`

Mit dieser Option kann für die QGIS Lokale (Sprache der Oberfläche, Zahlenformat, etc) eine von der ursprünglichen Installation abweichende Auswahl getroffen werden. Zum Beispiel startet `qgis --lang it` QGIS in der italienischen Lokalisierung.

`--project`

Es ist auch möglich, QGIS mit einer bestehenden Projektdatei zu starten. Fügen Sie einfach die Kommandozeilenoption `--project` gefolgt von Ihrem Projektnamen hinzu und QGIS wird mit dem entsprechenden Projekt gestartet.

--extent

Verwenden Sie diese Option, um mit einem bestimmten Kartenausschnitt zu beginnen. Sie müssen die Angaben für den Ausschnitt Ihres Bereichs in der folgenden Reihenfolge durch ein Komma getrennt hinzufügen:

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

Diese Möglichkeit kann gut mit der Option `--project` kombiniert werden, um ein bestimmtes Projekt in der gewünschten Größe zu öffnen.

--nologo

Diese Option blendet den Startbildschirm aus, wenn Sie QGIS starten.

--noversioncheck

Damit überspringen Sie die Suche nach einer neuen Version von QGIS beim Starten.

--noplugins

Wenn Sie beim Start Probleme mit Plugins haben, können Sie mit dieser Option das Laden der Plugins beim Start unterbinden. Sie sind danach immer noch über den Plugin-Manager verfügbar.

--nocustomization

Mit dieser Option werden alle vorgenommene *Oberflächen Anpassungen* beim Start ignoriert. Das bedeutet, dass QGIS mit allen Standardeinstellungen, wie bei einer Neuinstallation gestartet wird. Dies ist keine permanente Änderung, die Anpassungen werden wieder angewendet, wenn QGIS ohne diese Option gestartet wird.

Diese Option ist nützlich, um vorübergehend den Zugriff auf Werkzeuge zu ermöglichen, die durch Anpassung entfernt wurden.

--skipbadlayers

Mit dieser Option können Sie vermeiden, dass QGIS beim Start den Dialog *Nicht verfügbare Layer behandeln* anzeigt. Fehlende Layer werden dann beim Starten eines Projekts als nicht verfügbar gekennzeichnet. Mehr Details zu diesem Thema unter *Behandlung von fehlerhaften Dateipfaden*.

--customizationfile

Mit dieser Option können Sie eine UI-Anpassungsdatei definieren, die beim Start verwendet wird.

--globalsettingsfile

Die entsprechende Umgebungsvariable ist `QGIS_GLOBAL_SETTINGS_FILE`.

Mit dieser Option können Sie den Pfad zu einer `.ini`-Datei mit globalen Einstellungen (auch Standardeinstellungen genannt) angeben. Die Definitionen in der angegebenen Datei ersetzen die ursprünglichen Inline-Standardeinstellungen, wobei die Einstellungen des Benutzerprofils zusätzlich gesetzt werden.

QGIS sucht in der folgenden Reihenfolge nach der globalen Standardeinstellungsdatei, wobei die erste gefundene Datei verwendet wird:

- in dem Pfad, der durch diesen Kommandozeilenparameter angegeben wird

- in dem Pfad, der durch die Umgebungsvariable angegeben wird
- im AppDataLocation-Ordner, in dem persistente Anwendungsdaten gespeichert werden können; dieser wird vom Benutzer oder Systemadministrator verwaltet und wird vom Installationsprogramm nicht verändert und erfordert keine zusätzliche Einrichtung wie die Übergabe von Befehlszeilenparametern oder Umgebungsvariablen. Je nach Betriebssystem ist dieser Ordner:
  - `$HOME/.local/share/QGIS/QGIS3/`
  - `C:\Users\<username>\%AppData%\Roaming\QGIS\QGIS3\`
  - `$HOME/Library/Application Support/QGIS/QGIS3/`
- das Installationsverzeichnis, d. h.:file:*Ihr_QGIS_Paketpfad/resources/qgis_global_settings.ini*.

Es gibt derzeit keine Möglichkeit, die Einstellungen von QGIS aus in eine Datei zu schreiben; Sie können daher lediglich eine Kopie der ursprünglichen Einstellungsdatei erstellen, diese umbenennen und anpassen.

Das Festlegen des Dateipfads für die Datei `qgis_global_setting.ini` auf einen gemeinsam genutzten Netzlaufwerksordner ermöglicht es einem Systemadministrator, globale Einstellungen und Standardwerte für mehrere Rechner zu verwalten, indem er nur eine Datei bearbeitet.

--authdbdirectory

Diese Option ist ähnlich wie `--globalsettingsfile`, definiert aber den Pfad zu dem Verzeichnis, in dem die Authentifizierungsdatenbank gespeichert und geladen wird.

--code

Diese Option kann verwendet werden, um eine bestimmte Python-Datei direkt nach dem Start von QGIS auszuführen.

Wenn Sie zum Beispiel eine Python-Datei mit dem Namen `load_alaska.py` mit folgendem Inhalt haben:

```
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)
```

Angenommen, Sie befinden sich in dem Verzeichnis, in dem sich die Datei `load_alaska.py` befindet, können Sie QGIS starten, die Rasterdatei `landcover.img` laden und dem Layer mit dem folgenden Befehl den Namen „Alaska" geben:

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

--py-args

Diese Option ermöglicht es, Argumente an das Skript über den Parameter *–code* und allgemeiner an jede Python-Ausführung zu übergeben. Nach `--py-args` können keine anderen Parameter als – an QGIS übergeben werden. Alle Argumente nach `--py-args` bis `--` werden an den Python-Interpreter übergeben und von QGIS ignoriert.

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

Im obigen Code hat `test.py` folgenden Inhalt in `sys.argv`: [‚/home/user/test.py', ‚–specialScriptArguments', ‚a text arg', ‚and another arg']. `layer1` und `layer2` werden von QGIS normalerweise als zu ladende Layer behandelt.

`--defaultui`

Setzt beim Laden **permanent** die Benutzeroberfläche (UI) auf die Standardeinstellungen zurück. Diese Option stellt die Sichtbarkeit, Position und Größe der Bedienfelder und Werkzeugleisten wieder her. Die Standardeinstellungen der Benutzeroberfläche werden in den folgenden Sitzungen verwendet, sofern sie nicht erneut geändert werden.

Beachten Sie, dass diese Option keine Auswirkungen auf *GUI-Anpassung* hat. Durch die GUI-Anpassung ausgeblendete Elemente (z. B. die Statusleiste) bleiben auch bei Verwendung der Option `--defaultui` ausgeblendet. Siehe auch die Option `--nocustomization`.

`--hide-browser`

Blendet beim Laden das Bedienfeld *Browser* aus der Benutzeroberfläche aus. Das Bedienfeld kann durch einen Rechtsklick auf ein Feld in der Werkzeugleiste oder über die *Ansicht* ► *Bedienfelder* (*Einstellungen*► *Bedienfelder* in Linux KDE) aktiviert werden.

Wenn es nicht wieder aktiviert wird, bleibt das Bedienfeld „Browser“ in den folgenden Sitzungen ausgeblendet.

`--dxf-*`

Diese Angabe kann verwendet werden, um ein QGIS-Projekt in eine DXF-Datei zu exportieren. Es sind mehrere Optionen verfügbar:

- *–dxf-export*: gefolgt von dem DXF-Dateinamen, in den die Layer exportiert werden sollen;
- *–dxf-extent*: gefolgt von dem Ausschnitt der exportierten DXF-Datei;
- *–dxf-symbology-mode*: hier können verschiedenen Angaben folgen: `none` (keine Symbologie), `symbollayer` (Symbollayer Symbologie), `feature` (Objekt Symbologie);
- *–dxf-scale-denom*: der Maßstabsnenner der Symbologie;
- *–dxf-encoding*: die Datei-Kodierung;
- *–dxf-map-theme*: Wählen Sie ein vorhandenes *Kartenthema* aus dem Layer-Bedienfeld, das für den Export verwendet werden soll.

`--take-screenshots`

Nimmt Screenshots für die Benutzerdokumentation auf. Kann zusammen mit `--screenshots-categories` verwendet werden, um zu filtern, welche Kategorien/Abschnitte der Dokumentations-Screenshots erstellt werden sollen (siehe QgsAppScreenShots::Categories).

`--profile`

Lädt QGIS mit einem bestimmten Profil aus dem Profilordner des Benutzers. Wenn das angegebene Profil nicht existiert, wird es erstellt. Diese Option hat Vorrang vor *Benutzerprofil-Starteinstellung*.

--profiles-path

Mit dieser Option können Sie einen Pfad zum Laden und Speichern der Benutzerprofile wählen. Die Profile werden im Ordner `{Pfad}\profiles` erstellt, welche Einstellungen, installierte Plugins, Verarbeitungsmodelle und Skripte usw. enthalten.

Das bietet Ihnen die Möglichkeit z. B. alle Ihre Plugins und Einstellungen auf einem externen Laufwerk/USB-Stick mit sich zuführen oder die Einstellungen über einen Dateifreigabedienst zwischen verschiedenen Computern auszutauschen.

Die entsprechende Umgebungsvariable ist `QGIS_CUSTOM_CONFIG_PATH`.

--version-migration

Wenn Einstellungen aus einer älteren Version gefunden werden (z.B., der `.qgis2` Ordner von QGIS 2.18), ist es mit dieser Option möglich, diese Einstellungen in das Standard QGIS Profil zu importieren.

--openclprogramfolder

Mit dieser Option können Sie einen alternativen Pfad für Ihre OpenCL-Programme angeben. Dies ist für Entwickler nützlich, wenn sie neue Versionen der Programme testen wollen, ohne dabei die vorhandenen Programme ersetzen zu müssen.

Die entsprechende Umgebungsvariable ist `QGIS_OPENCL_PROGRAM_FOLDER`.

--

Mit dieser Option können die folgenden Argumente als Dateien zum Öffnen in QGIS übergeben werden. Dabei kann es sich um unterstützte Raster- oder Vektordateien, QGIS- Layerdefinitionsdateien oder Projektdateien handeln.

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

Der obige Code startet QGIS, öffnet die Datei `project` und lädt `layer1` darin.

5.6.2 Bereitstellen von QGIS innerhalb von Organisationen

Wenn Sie QGIS innerhalb einer Organisation (Unternehmen, Verwaltung, …) mit einer eigenen, benutzerdefinierten Konfigurationsdatei einsetzen möchten, müssen Sie zunächst den Inhalt der Standard-Konfigurationsdatei kopieren/einfügen, die sich in `Ihr_QGIS_Installationspfad/resources/qgis_global_settings.ini` befindet. Diese Datei enthält bereits einige Blöcke zu verschiedenen Bereichen, die jeweils durch einen entsprechenden Begriff in eckigen Klammern gekennzeichnet sind, z.B. `[digitizing]`. Wir empfehlen, dass Sie diese Standardwerte beibehalten und Ihre eigenen Abschnitte am Ende der Datei hinzufügen. Wenn ein Abschnitt in der Datei doppelt vorhanden ist, wird QGIS den letzten Abschnitt von oben nach unten übernehmen.

Sie können hier z.B. auch `allowVersionCheck=false` ändern, um die QGIS Versionsprüfung zu deaktivieren.

Wenn Sie das Migrationsfenster nach einer Neuinstallation nicht anzeigen möchten, müssen Sie den folgenden Abschnitt hinzufügen:

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

Wenn Sie eine benutzerdefinierte Variable hinzufügen möchten, müssen Sie folgendes eingeben:

```
[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
```

Nicht alle Optionen der QGIS GUI sind einstellbar, um die Möglichkeiten der `.ini`-Datei zu entdecken, schlagen wir vor, dass Sie die gewünschte Konfiguration in QGIS einstellen und dann in Ihrer `.ini`-Datei nach der entsprechenden Einstellung suchen.

Schließlich müssen Sie noch die Umgebungsvariable `QGIS_GLOBAL_SETTINGS_FILE` auf den Pfad Ihrer angepassten Datei setzen.

Darüber hinaus können Sie auch Dateien wie Python-Makros, Farbpaletten, Layoutvorlagen, Projektvorlagen, … entweder im QGIS-Systemverzeichnis oder im QGIS-Benutzerprofil bereitstellen.

- Layout-Vorlagen müssen im Verzeichnis `composer_templates` bereitgestellt werden.
- Projektvorlagen müssen im Verzeichnis `project_templates` bereitgestellt werden.
- Benutzerdefinierte Python-Makros müssen im Verzeichnis `python` bereitgestellt werden.

KAPITEL 6

Arbeiten mit Projektionen

Ein Koordinatenbezugssystem, kurz KBS, ist ein Verfahren zur Zuordnung von numerischen Koordinaten zu einer Position auf der Erdoberfläche. QGIS unterstützt ca. 7.000 Standard-KBS mit jeweils unterschiedlichen Anwendungsfällen, Vor- und Nachteilen. Die Auswahl eines geeigneten KBS für Ihre QGIS-Projekte und -Daten kann eine komplexe Aufgabe sein, aber QGIS hilft Ihnen dabei, diese Auswahl zu treffen und macht die Arbeit mit verschiedenen KBS so transparent und genau wie möglich.

6.1 Überblick zur Projektionsunterstützung

QGIS unterstützt etwa 7.000 bekannte KBS. Diese Standard-KBS basieren auf den von der European Petroleum Search Group (EPSG) und dem Institut Geographique National de France (IGNF) definierten KBS und werden in QGIS durch die zugrunde liegende Projektionsbibliothek „Proj“ zur Verfügung gestellt. Üblicherweise werden diese Standardprojektionen durch eine Kombination aus Institution und Code identifiziert, wobei die Institution ein Organisationsname wie „EPSG“ oder „IGNF“ und der Code eine eindeutige Nummer ist, die mit einem bestimmten KBS verbunden ist. Zum Beispiel wird das verbreitete WGS 84 Breiten-/Längen-KBS unter der Kennung `EPSG:4326` geführt, das Web Mapping Standard-KBS ist `EPSG:3857`.

Benutzerdefinierte KBS werden in einer Benutzer-KBS-Datenbank gespeichert. Für Informationen zur Verwaltung Ihrer benutzerdefinierten Koordinatenbezugssysteme siehe Abschnitt *Ein eigenes Koordinatenbezugssystem definieren*.

6.2 Koordinatenbezugssysteme für Layer

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`.

Immer wenn ein Layer in QGIS geladen wird, versucht QGIS, automatisch das korrekte KBS für diesen Layer zu ermitteln. In manchen Fällen ist dies nicht möglich, z.B. wenn ein Layer ohne diese Information bereitgestellt wurde. Sie können einstellen, wie QGIS verfahren soll, wenn es nicht automatisch das richtige KBS für einen Layer ermitteln kann:

1. Öffnen Sie *Einstellungen* ► *Optionen…* ► *KBS-Behandlung*

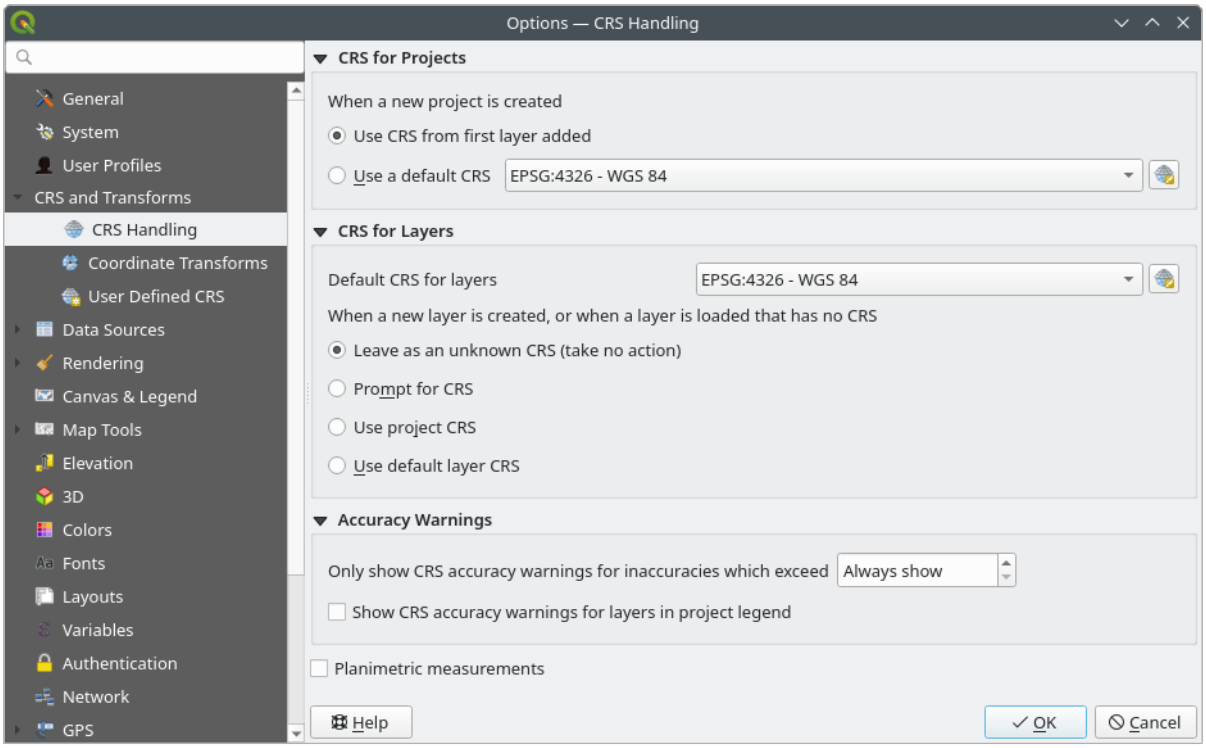


Abb. 6.1: Der Reiter KBS-Behandlung im Optionen-Dialog

2. Legen Sie im Bereich *KBS für Layer* die Aktion *Wenn ein neuer Layer erzeugt wird oder wenn ein Layer geladen wird, der kein KBS hat* fest:

   - *bei unbekanntem KBS belassen (nichts tun)*: es wird keine Aufforderung zur Auswahl eines KBS angezeigt, die Wahl des KBS kann zu einem späteren Zeitpunkt erfolgen. Dies kann z.B. nützlich sein, wenn viele Layer auf einmal geladen werden. Ein Layer ohne KBS wird im *Layer* Bedienfeld mit dem Symbol gekennzeichnet. Diese Layer werden nicht referenziert, wobei die Koordinaten des Layers als rein numerische, nicht erdgebundene Werte behandelt werden. Ebenso wird mit allen Layern verfahren, wenn das Projekt so eingestellt ist, :ref:` dass es kein KBS hat <project_crs>`.
   - *KBS abfragen*: Sie werden aufgefordert, das KBS manuell auszuwählen. Die Auswahl des richtigen KBS ist von entscheidender Bedeutung, da ein falsches Ihren Layer an der falschen Position auf der Erdoberfläche platziert! Manchmal ist das richtige KBS für einen Layer in den begleitenden Metadaten beschrieben, in anderen Fällen müssen Sie den ursprünglichen Autor der Daten kontaktieren, um das richtige KBS zu erfahren.
   - *Projekt KBS verwenden*
   - *KBS-Vorgabe verwenden*, wie es in der obigen Combobox *Vorgabe-KBS für Layer* eingestellt ist.

Tipp

Um mehreren Layern, die kein oder ein falsches KBS haben, in einem Arbeitsgang dasselbe KBS zuzuweisen, können Sie wie folgt vorgehen:

1. Wählen Sie die entsprechenden Layer im *Layer* Bedienfeld aus
2. Drücken Sie `Strg+Shift+C`; Sie können auch mit der rechten Maustaste auf einen der ausgewählten Layer klicken oder *Layer* ► *KBS von Layer(n) setzen* aufrufen.

3. Wählen Sie das richtige KBS aus
4. Drücken Sie *OK*; Sie können die korrekte Einstellung im Reiter *Quelle* des Eigenschaftsdialogs der Layer überprüfen.

Beachten Sie, dass das Ändern des KBS in dieser Einstellung die zugrundeliegende Datenquelle in keiner Weise verändert, sondern nur die Art und Weise, wie QGIS die Rohkoordinaten des Layers im aktuellen QGIS-Projekt interpretiert.

6.3 Koordinatenbezugssysteme für Projekte

Jedes Projekt in QGIS hat ein zugehöriges Koordinatenbezugssystem (KBS). Das KBS des Projekts bestimmt, wie die Daten von den zugrunde liegenden Rohkoordinaten auf die Karte projiziert werden, die in der Kartenansicht von QGIS dargestellt wird.

QGIS unterstützt die „on the fly“ KBS-Transformation sowohl für Raster- als auch für Vektordaten. Das bedeutet, dass unabhängig vom zugrundeliegenden KBS bestimmter Layer in Ihrem Projekt, diese immer automatisch in das für Ihr Projekt definierte KBS transformiert werden. Im Hintergrund projiziert QGIS alle in Ihrem Projekt enthaltenen Layer in das projekteigene KBS, sodass sie alle in der richtigen Position zueinander gerendert werden.

Es ist wichtig, das korrekte KBS für Ihre QGIS-Projekte zu wählen. Die Wahl eines ungeeigneten KBS kann dazu führen, dass Ihre Karten verzerrt aussehen und die realen Größen und Positionen von Merkmalen nur unzureichend wiedergeben. Wenn Sie in kleineren geografischen Gebieten arbeiten, gibt es normalerweise eine Reihe von Standard-KBS, die in einem bestimmten Land oder Verwaltungsgebiet verwendet werden. Es ist wichtig zu recherchieren, welche KBS für das Gebiet, das Sie kartieren, geeignet sind und sicherzustellen, dass Ihr QGIS-Projekt diesen Standards folgt.

Standardmäßig startet QGIS jedes neue Projekt mit der Standardprojektion``EPSG:4326`` („WGS 84“), welches ein globales Breiten-/Längen-basiertes Referenzsystem ist. Dieses Standard-KBS kann über die Einstellung *KBS für neue Projekte* im Reiter *KBS-Behandlung* unter *Einstellungen* ► *Optionen…* geändert werden (siehe Abb. 6.1). Es besteht auch die Möglichkeit, das KBS für neue Projekt von dem ersten Layer, der hinzugefügt wird, übernehmen zu lassen. Diese Auswahl wird für die Verwendung in nachfolgenden QGIS-Sitzungen gespeichert.

Das KBS des Projekts wird auch unten rechts in der QGIS-Statusleiste angezeigt. Es kann ebenfalls über den Reiter *KBS* im Dialog *Projekt* ► *Eigenschaften…* eingestellt werden.

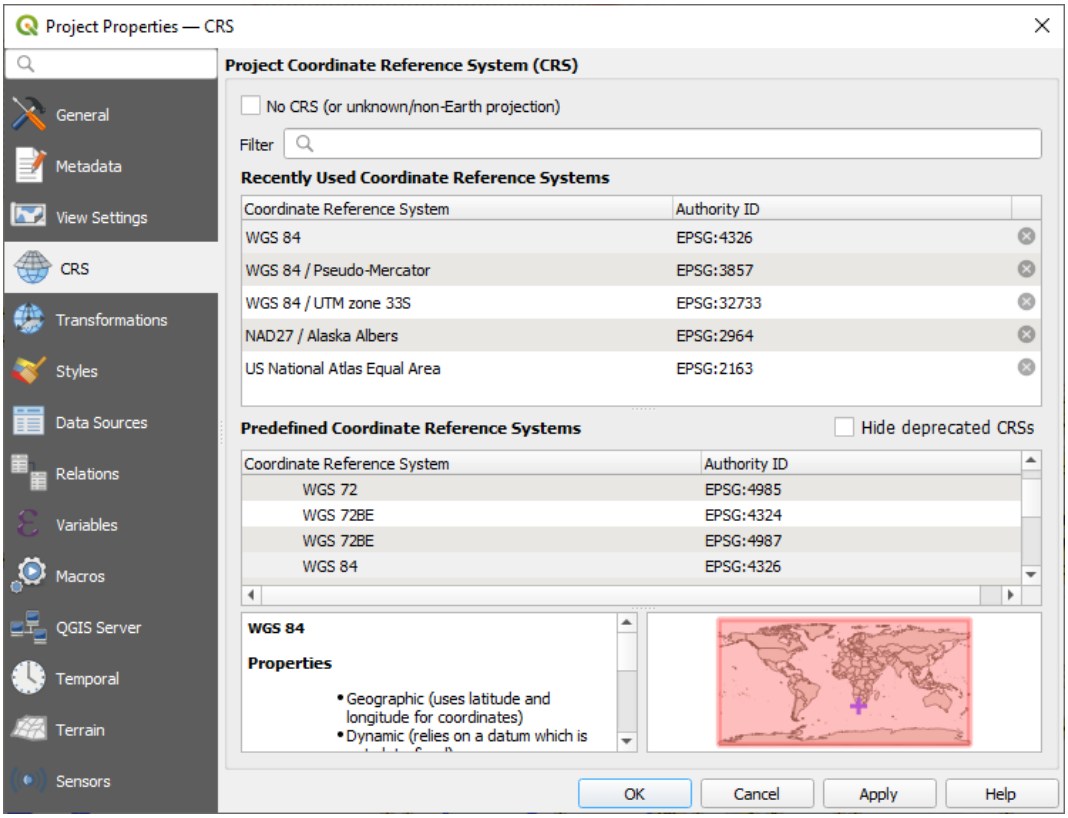


Abb. 6.2: Der Reiter KBS in den Projekteigenschaften

Folgende Optionen sind verfügbar:

- *Kein KBS (oder unbekannte/nicht-irdische Projektion)*: Wenn Sie diese Einstellung aktivieren, werden alle Layer und Kartenkoordinaten als einfache kartesische 2D-Koordinaten behandelt, ohne Bezug zu Positionen auf der Erdoberfläche. Diese Option kann verwendet werden, um die KBS eines Layers abzuschätzen (basierend auf seinen Rohkoordinaten) oder wenn QGIS für nicht-irdische Anwendungen wie Rollenspielkarten, Gebäudekartierungen oder mikroskopische Dinge verwendet wird. Bei dieser Auswahl gilt:
  - Beim Rendern der Layer wird keine Reprojektion durchgeführt: Die Features werden einfach mit ihren Rohkoordinaten gezeichnet.
  - Der Ellipsoid wird nicht berücksichtigt und wird auf `Nicht/Planimetrisch` gesetzt.
  - Die Entfernungs- und Flächeneinheiten sowie die Koordinatenanzeige sind deaktiviert und auf „unbekannte Einheiten“ eingestellt; alle Messungen erfolgen in unbekannten Karteneinheiten, und eine Umrechnung ist nicht möglich.
- Ein vorhandenes Koordinatenbezugssystem auswählen, das *geografisch*, *projiziert* oder *benutzerdefiniert* sein kann. Eine Vorschau der KBS-Ausdehnung auf der Erde wird angezeigt, um Ihnen die Auswahl des geeigneten Systems zu erleichtern. Dem Projekt hinzugefügte Layer werden on-the-fly in dieses KBS übersetzt, um sie unabhängig von ihrem ursprünglichen KBS korrekt zu überlagern. Die Verwendung von Einheiten und Ellipsoideinstellungen ist möglich und sinnvoll, und Sie können entsprechende Berechnungen durchführen.

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.

Tipp

Einstellung des Projekt-KBS von einem Layer

Sie können dem Projekt ein KBS zuweisen, indem Sie das KBS eines Layers verwenden:

1. Klicken Sie im *Layer* Bedienfeld mit der rechten Maustaste auf den Layer, von dem Sie das KBS übernehmen möchten.
2. Wählen Sie *Layer-KBS* und dann *Layer-KBS dem Projekt zuweisen* aus.

Das Projekt-KBS wird unter Verwendung des Layer-KBS neu festgelegt. Die Kartenansicht und die Koordinatenanzeige werden entsprechend aktualisiert und alle Layer im Projekt werden on-the-fly in das neue KBS des Projekts übersetzt.

6.4 Auswahl des Koordinatenbezugssystems

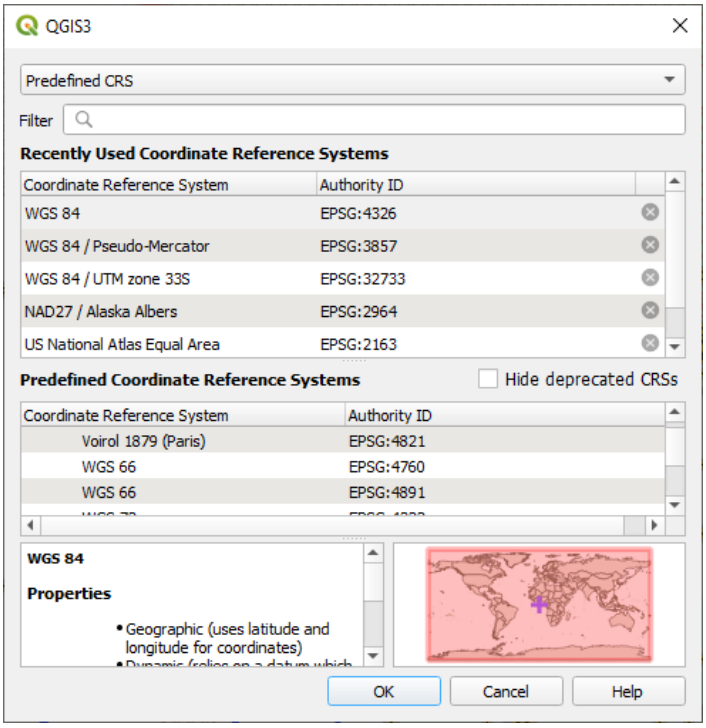


Abb. 6.3: Das KBS-Auswahl-Fenster

Dieser Dialog hilft Ihnen, einem Projekt oder einem Layer ein Koordinatenbezugssystem zuzuweisen. Die Elemente im Dialog sind:

- **Filter**: Wenn Sie den EPSG-Code, die Kennung oder den Namen eines Koordinatenbezugssystems kennen, können Sie die Filterfunktion verwenden, um es zu suchen. Sobald Sie im Filter eine Eingabe machen, werden nur noch die Übereinstimmungen angezeigt.
- **Kürzlich benutzte Koordinatenbezugssysteme**: Wenn Sie bestimmte KBS einmal verwendet haben, wird es in dieser Liste angezeigt. Klicken Sie auf einen Eintrag in dieser Liste, um das zugehörige KBS auszuwählen.
- **Vordefinierte Koordinatenbezugssysteme**: Dies ist eine Liste aller in QGIS verfügbaren KBS, einschließlich geografischer, projizierter und benutzerdefinierter Koordinatenbezugssysteme. Um ein KBS festzulegen, wählen Sie den entsprechenden Eintrag aus der Liste aus (ggf. muss zunächst der enthaltende Ast zuerst ausgeklappt werden). Das aktive KBS ist vorausgewählt.
- **PROJ-Text**: Im Bereich unten links werden allgemeine Informationen sowie die Definitionen, die für die Projektion verwendet werden, angezeigt.

Wenn Sie ein KBS aus der Liste „Kürzlich benutzte Koordinatensysteme“ entfernen möchten, können Sie dies auf verschiedene Weise tun:

- Drücken Sie die Taste Aus kürzlich verwendeten KBS entfernen auf der rechten Seite.
- Drücken Sie `Entfernen`, nachdem Sie das KBS ausgewählt haben, das Sie entfernen möchten.
- Klicken Sie mit der rechten Maustaste auf ein verwendetes KBS und wählen Sie *Gewähltes KBS aus kürzlich verwendeten KBS entfernen*, um das ausgewählte KBS zu entfernen oder *Alle kürzlich verwendeten KBS löschen*, um alle zu entfernen.

Der KBS-Selektor zeigt auch eine ungefähre Vorschau auf das geografische Gebiet, für das ein ausgewähltes KBS gültig ist. Viele KBS sind nur für die Verwendung in kleineren geografischen Gebieten vorgesehen, und Sie sollten ein KBS nicht außerhalb dieses Gebiets einsetzen. Die Vorschaukarte markiert ein ungefähres Einsatzgebiet, wenn ein KBS aus der Liste ausgewählt wird (farbig markierte Fläche). Darüber hinaus zeigt diese Vorschaukarte auch die aktuelle Ausdehnung der Kartenansicht an (violettes Kreuz). Es ist möglich, mit dem Mausrad in der Vorschaukarte zu zoomen.

6.5 Ein eigenes Koordinatenbezugssystem definieren

Wenn QGIS das benötigte Koordinatenbezugssystem nicht bereitstellt, können Sie ein eigenes KBS definieren. Um ein KBS zu definieren, wählen Sie aus dem Menü *Einstellungen* **|den den Punkt customProjection|** *Benutzerprojektionen…* aus. Benutzerdefinierte KBS werden in Ihrer QGIS Benutzerdatenbank gespeichert. Zusätzlich zu Ihren benutzerdefinierten KBS enthält diese Datenbank auch Ihre räumlichen Lesezeichen und andere benutzerdefinierte Daten.

Die Definition eines benutzerdefinierten KBS in QGIS erfordert ein gutes Verständnis der PROJ-Projektionsbibliothek. Für die Grundlagen, können Sie „Cartographic Projection Procedures for the UNIX Environment - A User's Manual" von Gerald I. Evenden, U.S. Geological Survey Open-File Report 90-284, 1990 lesen (verfügbar unter https://pubs.usgs.gov/of/1990/of90-284/ofr90-284.pdf).

Dieses Handbuch beschreibt die Verwendung von `proj` und verwandten Kommandozeilenprogrammen. Die kartographischen Parameter, die mit `proj` verwendet werden, werden im Benutzerhandbuch beschrieben und sind die gleichen, die auch von QGIS verwendet werden.

Der *Definition*-Dialog für ein benutzerdefiniertes KBS benötigt zunächst zwei Angaben, um eine eigene Projektion anlegen zu können:

1. einen Name,
2. die Auswahl des Formats, in dem die kartografischen Parameter gespeichert werden sollen (PROJ- oder WKT-Format).

Um eine neues KBS anzulegen müssen Sie:

1. auf die Schaltfläche Neues KBS hinzufügen klicken;
2. einen Namen eingeben;
3. ein Format auswählen, welches *Proj Zeichenkette* or *WKT* sein kann;
4. die *Parameter* eingeben.

> **Bemerkung**
>
> **Speichern Sie die KBS-Definition im WKT-Format**
>
> Obwohl beide Formate, `Proj String` und `WKT`, unterstützt werden, ist es dringend empfohlen, Projektionsdefinitionen im WKT-Format zu speichern. Wenn Ihnen die Definitionen im proj Format vorliegt, können Sie dieses Format auswählen, die Parameter eingeben und dann zum WKT Format wechseln. QGIS wird die Definition automatisch in das WKT-Format konvertieren, das Sie dann speichern können.

5. Klicken Sie auf *Validieren*, um zu prüfen, ob Ihre Angaben einer gültigen Projektionsdefinition entsprechen.

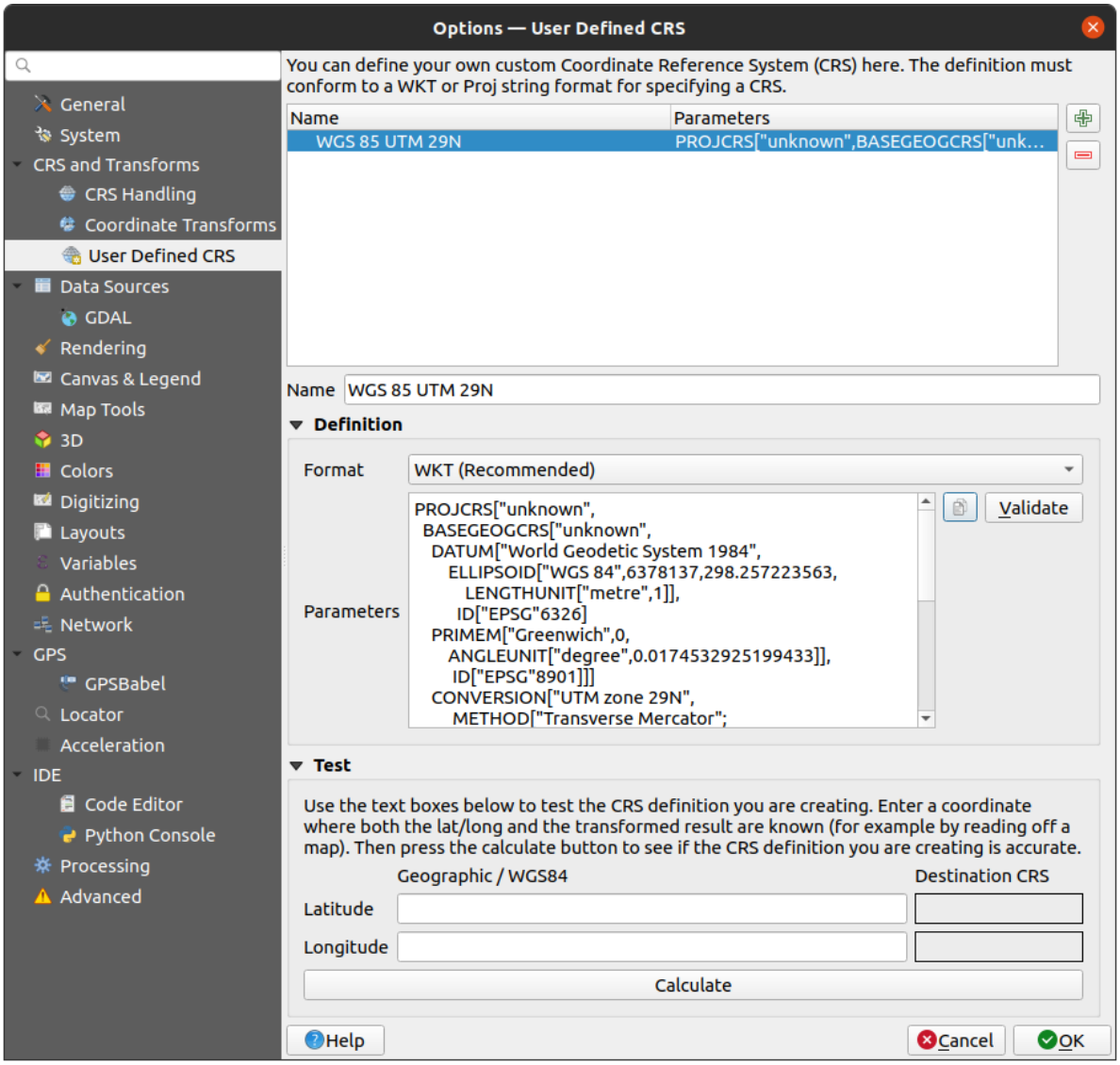


Abb. 6.4: Der Benutzerkoordinatensystem Dialog

Wenn Sie eine bekannte Zuordnung von WGS 84 zu ihrem Koordinatensystem haben, können Sie Ihre KBS Angaben testen, um zu sehen, ob sie korrekte Ergebnisse liefern. Geben Sie dazu WGS 84-Breiten- und Längenwerte in die Felder *Breitengrad* bzw. *Längengrad* ein. Klicken Sie auf *Berechnen* und vergleichen Sie das Ergebnis mit den bekannten Werten in Ihrem Koordinatenbezugssystem.

6.5.1 Einbinden einer NTv2-Transformation in QGIS

Um eine NTv2-Transformationsdatei in QGIS zu integrieren, ist ein weiterer Schritt notwendig:

1. Legen Sie die NTv2-Datei (.gsb) in den KBS/Proj-Ordner, den QGIS verwendet (z. B. `C:\OSGeo4W64share\proj` für Windows-Benutzer);
2. Fügen Sie **nadgrids** (`+nadgrids=NamederDatei.gsb`) zur Proj-Definition im Feld *Parameter* hinzu.

Abb. 6.5: Hinzufügen einer NTv2-Transformation

6.6 Datumstransformationen

In QGIS ist die „on-the-fly“ KBS-Transformation standardmäßig aktiviert, d.h. wann immer Sie Layer mit unterschiedlichen Koordinatensystemen verwenden, projiziert QGIS diese automatisch auf das Projekt-KBS. Für einige KBS gibt es eine Reihe von verschiedenen Transformationen, die für die Reprojektion in das Projekt-KBS zur Verfügung stehen.

Standardmäßig wird QGIS versuchen, die genaueste verfügbare Transformation zu verwenden. In einigen Fällen kann dies jedoch nicht möglich sein, z.B. wenn zusätzliche Unterstützungsdateien für die Verwendung einer Transformation erforderlich sind. Wenn eine genauere Transformation zur Verfügung steht, aber derzeit nicht verwendet werden kann, zeigt QGIS eine Warnmeldung an, die Sie auf die genauere Transformation hinweist und wie Sie diese auf Ihrem System aktivieren können. Normalerweise erfordert dies den Download eines externen Pakets von Transformations-Unterstützungsdateien und das Extrahieren dieser Dateien in den Ordner `proj` unter Ihrem QGIS Ordner *Benutzerprofil*.

Wenn gewünscht, kann QGIS Sie auch darauf hinweisen, wenn mehrere mögliche Transformationen zwischen zwei KBS möglich sind, und Ihnen die Möglichkeit geben, eine fundierte Auswahl zu treffen, welche Transformation für Ihre Daten am besten geeignet ist.

Diese Anpassung erfolgt im Reiter *Einstellungen* ► *Optionen* ► *Koordinatentransformationen*

- Sie können *Datumstransformation erfragen, wenn mehrere verfügbar* auswählen: wenn mehr als eine geeignete Datentransformation für eine KBS-Kombination vorhanden ist, wird automatisch ein Dialog geöffnet, der den Benutzer auffordert, die Datentransformation auszuwählen, die für das Projekt verwendet werden soll. Wenn das Kontrollkästchen *Als Voreinstellung übernehmen* bei der Auswahl einer Transformation aus diesem Dialog angekreuzt wird, dann wird die Auswahl gespeichert und automatisch auf alle neu erstellten QGIS-Projekte angewendet.
- Sie können eine Liste mit geeigneten Datentransformationen erstellen, die als Standardwerte beim Laden eines Layers in ein Projekt oder bei der Neuprojektion eines Layers verwendet werden:

  Verwenden Sie die Schaltfläche , um den Dialog *Datumstransformation* zu öffnen. Dann:

  1. Wählen Sie das *Quell-KBS* des Layers mithilfe des Dropdown-Menüs oder der KBS wählen Schaltfläche.
  2. Geben Sie das *Ziel-KBS* auf die gleiche Weise an.
  3. Eine Liste der verfügbaren Transformationen von der Quelle zum Ziel wird in der Tabelle angezeigt. Wenn Sie auf eine Zeile klicken, werden Details zu den verwendeten Einstellungen sowie die entsprechende Genauigkeit und der Anwendungsbereich der Transformation angezeigt.

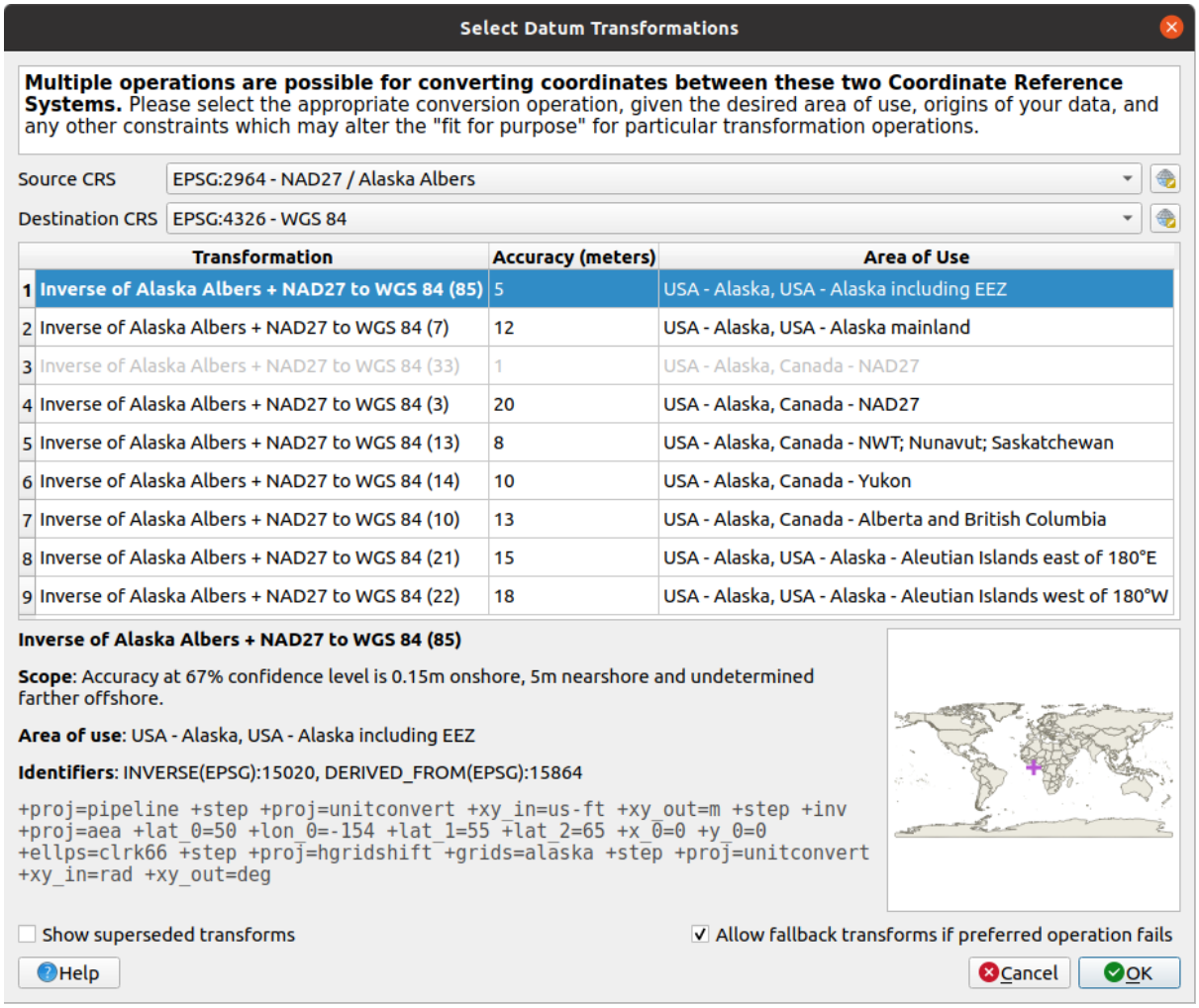


Abb. 6.6: Auswahl einer bevorzugten Standard-Datentransformation

In einigen Fällen kann es vorkommen, dass eine Transformation auf Ihrem System nicht zur Verfügung steht. In diesem Fall wird die Transformation zwar in dieser Liste angezeigt (grau), kann aber nicht ausgewählt werden, bis Sie das erforderliche Paket zur Unterstützung der Transformation installiert haben. Normalerweise gibt es eine Schaltfläche zum Herunterladen und Installieren des entsprechenden Pakets, das dann unter dem Ordner `proj` im aktiven Verzeichnis *Benutzerprofil* gespeichert wird.

4. Suchen Sie die gewünschte Transformation und wählen Sie sie aus.
5. Sie können die Option *Rückgriffstransformation erlauben, wenn bevorzugte Operation scheitert* festlegen.
6. Klicken Sie *OK*.

   Der Tabelle wird unter *Datumtransformationsvorgabe* eine Zeile mit Informationen über das *Quell-KBS*, das *Ziel-KBS*, die *Operation*, die für die Transformation angewendet wird, und ob *Rückgriffstransformation erlaubt* aktiviert ist, hinzugefügt.

Von nun an verwendet QGIS automatisch die ausgewählten Datentransformationen für weitere Transformationen zwischen diesen beiden KBS, bis Sie diese aus der Liste entfernen ( ) oder den Eintrag in der Liste ändern ( ).

Datumstransformationen, die im Reiter *Einstellungen* ► |Optionen| *Optionen* ► *Koordinatentransformationen* eingestellt werden, werden an alle neuen QGIS-Projekte weitergegeben, die auf dem System erstellt werden. Zusätzlich kann in einem bestimmten Projekt über den Reiter *Transformationen* des Dialogs *Projekteigenschaften*

(*Projekt* ► *Eigenschaften…*) ein eigener Satz von Transformationen festlegen werden. Diese Einstellungen gelten nur für das jeweilige Projekt.

KAPITEL 7

Karten visualisieren

Das Erstellen von Karten ist einer der hauptsächlichen „Geschäftsbereiche“ von QGIS. Kartenansichten sind Abbildungen von im Projekt geladenen Layern, auf die bestimmte Darstellungseinstellungen angewendet werden. Sie können vom Typ *2D* oder *3D* sein oder entlang einer *Profillinie* gezeichnet werden, unterschiedliche Maßstäbe haben oder verschiedene Ausschnitte haben.

7.1 2D Kartenansichten

Die 2D-Kartenansicht, meist nur **Kartenansicht** genannt, ist der zentrale Ort, an dem Karten dargestellt werden. QGIS öffnet standardmäßig mit einer einzigen Kartenansicht (`Hauptkarte`), die Layer in 2D zeigt und eng mit dem *Layer* Bedienfeld verbunden ist. In diesem Bedienfeld können die Einstellungen gemacht werden (Symbologie, Beschriftung, Sichtbarkeit …), die Sie auf die geladenen Layer anwenden möchten.

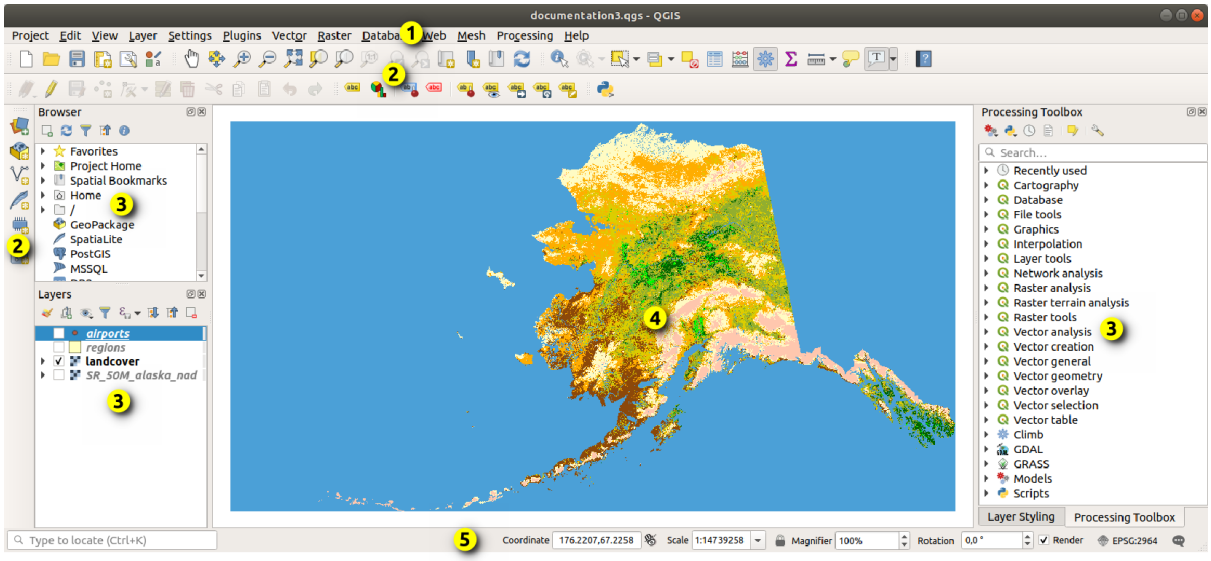


Abb. 7.1: Die QGIS Oberfläche

7.1.1 Funktionen des Kartenfensters

Wenn Sie einen Layer hinzufügen (s.a. *Öffnen von Daten*), sucht QGIS automatisch nach dem Koordinatenbezugssystem (KBS) des Layers. Wenn für das Projekt ein anderes KBS eingestellt ist, dann wird die Darstellung des Layers „on-the-fly“ in dieses KBS überführt (siehe *Koordinatenbezugssysteme für Projekte*). Wenn Sie mit einem leeren QGIS-Projekt beginnenund wird die Kartenansicht auf die Ausdehnung des/der zuerst hinzugefügten Layer gezoomt. Wenn bereits Layer im Projekt vorhanden sind, wird keine Größenänderung der Kartenansicht durchgeführt, so dass nur Features sichtbar sind, die innerhalb der aktuellen Kartenansicht liegen.

Wenn Sie in die Kartenansicht klicken, können Sie mit dieser interagieren, indem Sie den Ausschnitt der Karte verschieben oder hinein- oder heraus-zoomen. Spezielle Werkzeuge sind in der *Navigations-Werkzeugleiste* und im Menü *Ansicht* zu finden, für viele davon existieren Tastenkombinationen.

Tab. 7.1: Navigieren in der Kartenansicht

Werkzeug	Funktion
Karte verschieben	• Einfacher Linksklick: die Karte wird auf den angeklickten Punkt zentriert, der Maßstab bleibt unverändert • Halten Sie die linke Maustaste gedrückt und ziehen Sie die Kartenansicht
Hineinzoomen	• Einfacher Linksklick: die Karte wird auf den angeklickten Punkt zentriert, wobei der Maßstab verdoppelt wird • Ziehen Sie mit der linken Maustaste in der Kartenansicht ein Rechteck auf, um diesen Bereich zu vergrößern • Halten Sie die `Alt` Taste gedrückt, um zum Werkzeug Herauszoomen zu wechseln
Herauszoomen	• Einfacher Linksklick: die Karte wird auf den angeklickten Punkt zentriert, wobei der Maßstab halbiert wird • Ziehen Sie mit der linken Maustaste ein Rechteck in der Kartenansicht auf, um aus einem Bereich herauszuzoomen • Halten Sie die `Alt` Taste gedrückt, um zum Werkzeug Hereinzoomen zu wechseln
Karte zu gewählten Objekten ver	Verschiebt die Karte auf die ausgewählten Objekte aller ausgewählten Layer im *Layer* Bedienfeld
Zu ausgewählten Objekten zoom	Zoom auf die ausgewählten Objekte aller ausgewählten Layer im *Layer* Bedienfeld (*Ist auch im Layer-Kontextmenü verfügbar*)
Auf Layer zoomen	Zoomt auf die Ausdehnung aller ausgewählten Layer im *Layer* Bedienfeld (*Ist auch im Layer-Kontextmenü verfügbar*)
Volle Ausdehnung	Zoom auf den Umfang aller Layer im Projekt oder auf die *gesetzte Projekt-Ausdehnung*
Zoom zurück	Zoomt die Karte auf die vorherige Ausdehnung
Zoom vor	Zoomt die Karte auf die nächste Ausdehnung
Auf eigene Auflösung zoomen	Zoomt die Karte auf einen Maßstab, bei der ein Pixel des aktiven Raster-Layers einem Bildschirmpixel entspricht (*Ist auch im Layer-Kontextmenü verfügbar*)

Fortsetzung auf der nächsten Seite

Tab. 7.1 – Fortsetzung der vorherigen Seite

Werkzeug	Funktion
Maus	• Karte verschieben: Halten Sie das Mausrad gedrückt und bewegen Sie die Maus • Zoomen: Drehen Sie das Mausrad zum Vergrößern oder Verkleinern. Wenn Sie die Taste `Strg` gedrückt halten, während Sie das Mausrad drehen, erhalten Sie einen feineren Zoom • Drücken Sie die „Vor“- oder „Zurück“-Taste der Maus, um den Zoomverlauf der Kartenansicht zu durchlaufen
Tastatur	• Karte verschieben: Halten Sie die `Leertaste` gedrückt und bewegen Sie die Maus oder drücken Sie die Pfeiltasten, um zu verschieben • Herein zoomen: Drücken sie `Bild auf` oder `Strg + +` • Heraus zoomen: Drücken Sie `Bild ab` oder `Strg + -` • Auf Bereich zoomen: Wenn bestimmte Kartenwerkzeuge aktiv sind (Identifizieren, Messen…), können Sie bei gedrückter `Umschalt` Taste ein Rechteck auf der Karte aufziehen, um auf diesen Bereich zu zoomen. Nicht kompatibel mit aktiven Auswahl- oder Bearbeitungswerkzeugen.

Mit einem rechten Mausklick in die Karte können Sie die Koordinaten des angeklickten Punktes kopieren (*Koordinaten kopieren*), entweder im KBS der Karte, in WGS84-Koordinaten oder Sie können ein anderes KBS auswählen. Die kopierten Informationen stehen dann in der Zwischenablage zur Verfügung und können über „Einfügen“ in einer anderen Anwendung verwendet werden.

7.1.2 Steuerung der Kartendarstellung

Standardmäßig werden von QGIS alle sichtbaren Layer dargestellt, wenn die Kartenansicht aktualisiert wird. Die folgenden Ereignissen führen zu einer Aktualisierung der Kartenansicht:

- Ändern der Sichtbarkeit eines Layers
- Ändern der Symbologie eines sichtbaren Layers
- Einen Layer hinzufügen
- Kartenausschnitt verschieben oder Zoomen
- Ändern der Größe des QGIS Fensters

QGIS ermöglicht es Ihnen, die Darstellung auf verschiedene Weise zu steuern:

- Auf *Projekt-Ebene*
- Auf Layer-Ebene, z. B. unter Verwendung der *Maßstabsabhängigen Sichtbarkeit*
- Mit speziellen Werkzeugen in der grafischen Benutzeroberfläche

Um das Zeichnen der Karte zu stoppen, drücken Sie die `Esc` Taste. Dadurch wird die Aktualisierung der Kartenansicht angehalten und die Karte bleibt so gezeichnet, wie sie es zum Zeitpunkt des Abbruchs war. Nach dem Drücken der `Esc` Taste kann es jedoch einige Zeit dauern, bis der Prozess tatsächlich abgebrochen wird.

Um eine Aktualisierung der Kartenansicht zu unterdrücken, können Sie das Kontrollkästchen *Zeichnen* im rechten Bereich der Statusleiste unten deaktivieren. Die Kartenansicht wird dann nicht aktualisiert, wenn eins der oben genannten Ereignisse eintritt. Dies können Sie z.B. für eine der folgenden Situationen nutzen:

- Hinzufügen von mehreren Layern und dabei die Darstellung anpassen, bevor sie das erste Mal gezeichnet werden
- Hinzufügen eines oder mehrerer großer Layer und Einstellen der Skalierungsabhängigkeit
- Hinzufügen eines oder mehrerer großer Layer und Zoomen auf einen bestimmten Ausschnitt

- eine beliebige Kombination der oben genannten Punkte

Wenn Sie das *Zeichnen* Kontrollkästchen wieder aktivieren, findet unmittelbar eine Aktualisierung der Kartenansicht statt.

7.1.3 Zeitabhängige Darstellung in der Kartenansicht

In QGIS können Layer zeitlich gesteuert dargestellt werden, d.h. die Kartenansicht wird in Abhängigkeit von der Zeit verändert. Um dies durchführen zu können, benötigen Sie:

1. Einen oder mehrere Layer, für den zeitlich-dynamische Eigenschaften festgelegt wurden. QGIS unterstützt die zeitliche Steuerung für verschiedene Layer-Typen mit unterschiedlichen Einstellungen. Diese unterscheiden sich in erster Linie bei der Einstellung des Zeitintervalls, in dem der Layer angezeigt werden kann:
   - *Raster Layer*: es wird nur festgelegt, ob der Layer angezeigt werden soll oder nicht
   - *WMTS Layer*: es wird festgelegt, ob die Daten auf der Grundlage eines statischen Zeitbereichs oder nach einem dynamischen Zeitbereich dargestellt werden sollen
   - *Vektor-Layer*: die Objekte können über Zeitangaben gesteuert werden, die in der Attributtabelle enthalten sind
   - *Netz-Layer*: die Werte der aktiven Datensatzgruppen können dynamisch anzeigt

   Wenn zeitlich-dynamische Eigenschaften für einen Layer aktiviert sind, ist neben dem Layer im *Layer* Bedienfeld ein Symbol sichtbar, um anzuzeigen, dass der Layer zeitgesteuert ist. Klicken Sie auf das Symbol, um die zeitlichen Einstellungen zu ändern.
2. Aktivierung der zeitlichen Steuerung der Kartenansicht mit dem *Zeitsteuerung* Bedienfeld. Das Bedienfeld selber kann angezeigt werden über
   - das Symbol Zeitsteuerungsfenster in der *Kartennavigationswerkzeugleiste*
   - über das Menü *Ansicht* ► *Bedienfelder* ► *Zeitsteuerung*
   - über das Menü *Ansicht* ► *Datenfiltrierung* ► *Zeitsteuerung*

Das Zeitsteuerung-Bedienfeld

Das *Zeitsteuerung* Bedienfeld bietet die folgenden Einstellungsmöglichkeiten:

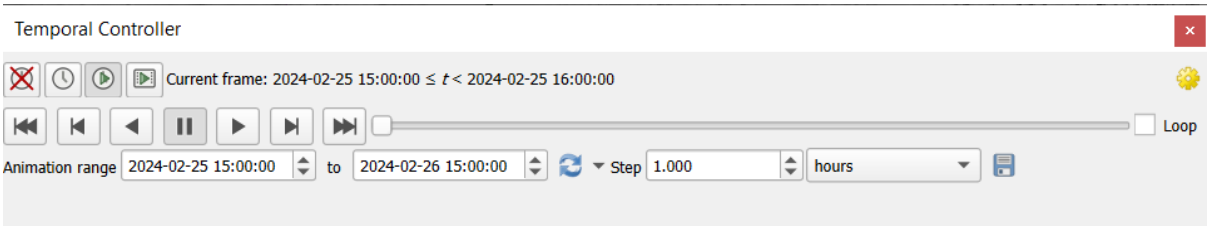


Abb. 7.2: Zeitsteuerung-Bedienfeld im Navigationsmodus

- Zeitliche Navigation abschalten: alle zeitlichen Einstellungen sind deaktiviert und die aktiven Layer werden wie gewohnt angezeigt
- Zeit-Navigation in festem Bereich: ein Zeitbereich wird festgelegt und nur Layer oder Objekte, deren Zeitbereich sich mit diesem überschneidet, werden auf der Karte angezeigt
- Animierte Zeitnavigation: ein Zeitbereich und eine Schrittlänge innerhalb dieses Zeitbereichs werden festgelegt und es werden nur Layer oder Objekte angezeigt, die sich innerhalb des jeweiligen Schrittes befinden

- Animierter Film: Für die Animation wird eine Gesamtzahl von Bildern festgelegt, wobei die Karteneinstellungen bei jedem Schritt weitergeschaltet werden und der Fortschritt Bild für Bild ohne zeitbasierte Filterung der Daten angezeigt wird.
- Einstellungen für die Steuerung der Animation:
  - *Bildrate*: Anzahl der Bilder, die pro Sekunde angezeigt werden
  - *Kumulativer Bereich*: alle Animationsbilder haben die gleiche Startzeit, aber unterschiedliche Enddaten und -zeiten. Dies ist nützlich, wenn Sie Daten in Ihrer zeitlichen Visualisierung akkumulieren möchten, anstatt ein „bewegliches Zeitfenster“ über die Daten laufen zu lassen.

Animieren einer zeitlichen Navigation

Eine zeitliche Animation besteht aus einer Reihe von Bildern aller zeitlich aktiven Layer zu einem bestimmten Zeitpunkten. So erstellen Sie eine zeitliche Animation:

1. Schalten Sie auf Animierte Zeitnavigation um, sodass die Bedienelemente des Players anzeigt werden
2. Geben Sie den *Animationsbereich* ein, der berücksichtigt werden soll . Mit der Schaltfläche kann dieser wie folgt automatisch definiert werden:
   - *Auf ganzen Bereich setzen* legt den Animationsbereich so fest, dass die Zeiträume aller zeitlich aktivierten Layer berücksichtigt werden
   - *Auf Projektbereich voreinstellen* legt den Animationsbereich so fest, wie er in den *Projekteigenschaften* definiert ist
   - *Bereich eines einzelnen Layers setzen* legt den Bereich auf die Spanne der Werte eines einzelnen Layers fest
3. Mit der Angabe unter *Schritt* wird die Zeitspanne in einzelne Intervalle unterteilt. Es werden verschiedene Einheiten unterstützt z. B. `Sekunden` oder `Jahrhunderte`. Als Einheit ist auch die Option `Quellzeitstempel` möglich: Wenn dies ausgewählt ist, bewirkt es, dass die zeitliche Navigation zwischen unterschiedlichen Zeitbereichen der Layer wechselt. Dies ist nützlich, wenn Layer verwendet werden, deren Daten nicht in regelmäßigen Abständen vorliegen. Dies kann z. B. bei einem WMS-T-Dienst der Falls sein, der Bilder bereitstellt, die unregelmäßige Zeitpunkte wiedergeben. Diese Option ermöglicht es ihnen, mit einem Schritt einfach in den Zeitbereich zu wechseln, in dem das nächste Bild verfügbar ist.
4. Klicken Sie auf die Schaltfläche , um eine Vorschau der Animation zu erhalten. QGIS erzeugt eine Serie von Bildern mit den Layern und/oder Objekten, die der eingestellten Zeit entsprechen. Die Anzeige der Layer oder Objekte hängt dabei davon ab, ob deren Zeitstempel innerhalb des jeweiligen Zeitintervalls liegen.

   Die Animation kann auch durch Verschieben des Zeitreglers in der Vorschau angezeigt werden. Wenn Sie das Kontrollkästchen *Schleife* markieren, wird die Animation wiederholt ausgeführt, während ein Klick auf eine laufende Animation stoppt. Ein vollständiger Satz der üblichen Player-Schaltflächen ist verfügbar.

   Durch horizontales Scrollen mit dem Mausrad (sofern von Ihrer Maus unterstützt), während sich der Cursor innerhalb der Kartenansicht befindet, können Sie auch den Schieberegler für die zeitliche Navigation vorwärts und rückwärts verschieben (“scrubben”).
5. Klicken Sie auf die Schaltfläche Animation exportieren, wenn Sie die Serie von Bildern exportieren wollen. Diese kann dann z. B. in einem Videoeditor zusammengefügt werden:

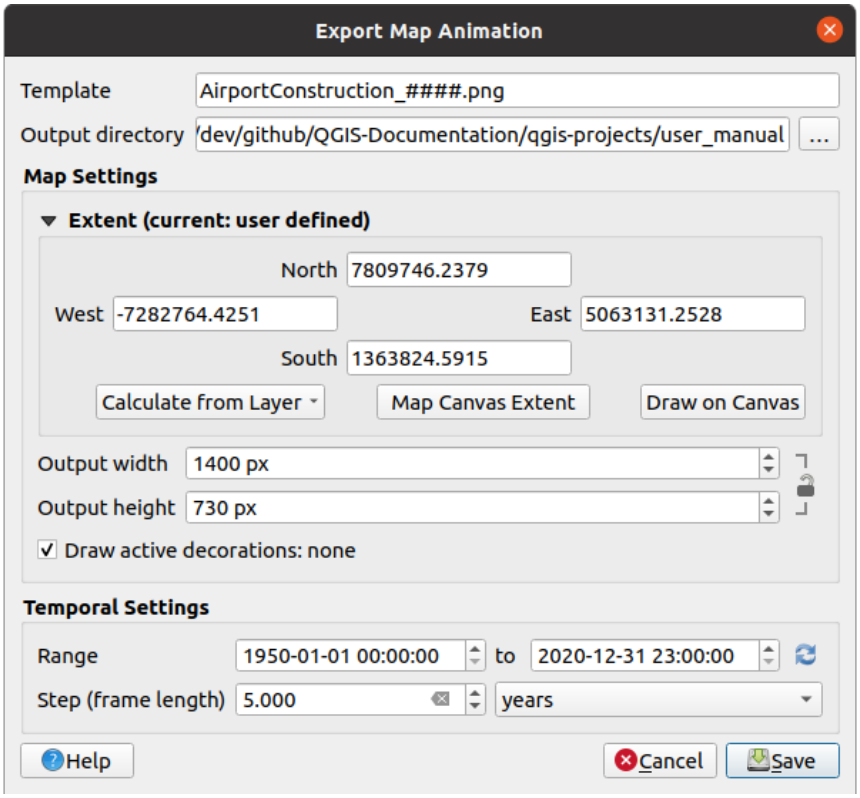


Abb. 7.3: Fenster zum Exportieren von Bildern einer Animation

- in der *Vorlage* für den Dateinamen werden die Platzhalter `####` durch fortlaufende Bildnummern ersetzt
- das *Ausgabeverzeichnis* muss angegeben werden
- unter *Karteneinstellungen* können Sie:
  - die *räumliche Ausdehnung* festlegen
  - die Auflösung der Bilder mit *Ausgabebreite* und *Ausgabehöhe* festlegen
  - *Aktive Dekorationen anzeigen* aktivieren/deaktivieren: damit legen Sie fest, ob *Dekorationen* in den Ausgabe-Bildern dargestellt sein sollen
- unter *Zeiteinstellungen* können Sie folgende Parameter noch verändern:
  - den *Bereich* für die Zeitspanne der Animation
  - die *Schrittweite (Einzelbildlänge)* in der gewünschten Einheit

7.1.4 Das Bedienelement für die Höhensteuerung

Mit dem *Höhensteuerung*-Bedienelement können Sie Höhenwerte (Z-Werte) in 2D-Karten verarbeiten. Diese Option unterstützt derzeit Punktwolken-Layer und Raster-DEMs.

Aktivieren Sie das *Höhensteuerung*-Bedienelement in der Kartenansicht, indem Sie im Menü *Ansicht* auf *Datenfiltrierung* klicken und dann *Höhensteuerung* auswählen. Diese wird darauf hin als Schieberegler auf der linken Seite der Kartenansicht angezeigt. Oberhalb des Schiebereglers befindet sich das Einstellungen-Menü, in dem Sie folgende Optionen haben:

- Setzen Sie eine Wert für *Fester Größenbereich*, um den Höhenbereich auf eine bestimmte Größe festzulegen. Diese Einstellung wird im Projekt gespeichert, sodass die Konsistenz Ihres Höhenfilters gewährleistet ist. Verwenden Sie *Richtung umkehren*, um den Schieberegler der Höhensteuerung umzukehren, sodass er sich von hohen Werten unten zu niedrigen Werten oben bewegt.

- Mit *Höhenbereich setzen* gelangen Sie zu den Projekt-Eigenschaften, wo Sie auf dem Reiter *Geländehöhe* spezielle Unter- und Obergrenzen für den Höhenbereich festlegen können.
- Mit *Höhenfilter abschalten* können Sie den Schieberegler aus der Kartenansicht entfernen

7.1.5 Räumliche Lesezeichen

Räumliche Lesezeichen ermöglichen es Ihnen, einen bestimmten Ausschnitt der Karte zu speichern und später dorthin zurückzukehren. Standardmäßig werden Lesezeichen im Benutzerprofil gespeichert (als *Benutzer-Lesezeichen*), d.h. sie sind in jedem Projekt, das der Benutzer öffnet, verfügbar. Sie können aber auch nur für ein bestimmtes Projekt gespeichert werden (als *Projekt-Lesezeichen*), wobei sie dann mit in der Projektdatei abgelegt werden. Das kann z. B. hilfreich sein, wenn das Projekt mit anderen Benutzern geteilt wird.

Ein Lesezeichen erstellen

Um ein Lesezeichen zu erstellen, gehen Sie wie folgt vor:

1. Wählen Sie mittels zoomen und schwenken in der Kartenansicht den Ausschnitt aus, für den Sie ein Lesezeichen setzen wollen
2. Wählen Sie entweder im Menü *Ansicht* ► den Punkt *Neues räumliches Lesezeichen…* aus oder klicken Sie im *Browser* Bedienfeld mit der rechten Maustaste auf den Eintrag *Räumliche Lesezeichen* und wählen Sie *Neues räumliches Lesezeichen…* oder drücken Sie `Strg+B`; jeweils wird das Dialog-Fenster *Lesezeichenbearbeitung* geöffnet

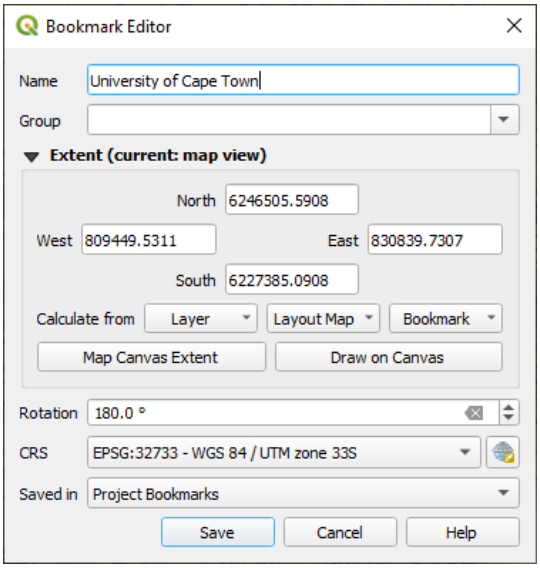


Abb. 7.4: Der Lesezeichenbearbeitung-Dialog

3. Geben Sie einen Namen für das Lesezeichen ein
4. Optional: Sie können einen Gruppennamen eingeben oder eine bestehende Gruppe auswählen, in der verwandte Lesezeichen gespeichert werden sollen
5. Optional: Sie können den Ausschnitt der Karte, den Sie speichern möchten, mit dem *Ausdehnung-Werkzeug* noch anpassen; vorausgewählt ist der aktuelle Ausschnitt der Kartenansicht
6. Optional: Geben Sie eine *Drehung* der Karte in Grad an
7. Optional: Geben Sie ein anderes *KBS*, als das des Projekts an, das für den Ausschnitt verwendet werden soll
8. Wählen Sie aus, ob das Lesezeichen als *Benutzer-Lesezeichen* (vorausgewählt) oder als *Projekt-Lesezeichen* gespeichert werden soll

9. Drücken Sie *Speichern*, um das Lesezeichen der Liste hinzuzufügen

Es ist möglich, mehrere Lesezeichen mit demselben Namen zu erstellen.

Arbeiten mit Lesezeichen

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.

Sie können die in der nachfolgenden Tabelle aufgeführten Aktionen ausführen.

Tab. 7.2: Verwalten von Lesezeichen-Aktionen

Aufgabe	Räumliche Lesezeichenverwaltung	Browser
Auf ein Lesezeichen zoomen	Doppelklicken Sie auf das Lesezeichen oder wählen Sie es aus und drücken Sie die Schaltfläche *Zum Lesezeichen zoomen*	Doppelklicken Sie auf das Lesezeichen oder ziehen Sie es auf die Kartenansicht oder klicken Sie mit der rechten Maustaste darauf und wählen Sie *Zum Lesezeichen zoomen*
Lesezeichen löschen	Wählen Sie das Lesezeichen aus und klicken Sie auf die Schaltfläche *Lesezeichen löschen*	Klicken Sie mit der rechten Maustaste auf das Lesezeichen und wählen Sie *Räumliches Lesezeichen löschen*
Lesezeichen nach XML exportieren	Klicken Sie auf die Schaltfläche *Lesezeichen importieren/exportieren* und wählen Sie *Export*. Alle Lesezeichen (Benutzer- und Projekt-) werden in einer xml-Datei gespeichert	Wählen Sie einen oder mehrere Ordner (Benutzer oder Projekt) oder Unterordner (Gruppen) aus, klicken Sie dann mit der rechten Maustaste und wählen Sie *Räumliche Lesezeichen exportieren…*. Die Lesezeichen-Auswahl wird gespeichert.
Lesezeichen aus XML importieren	Klicken Sie auf die Schaltfläche *Lesezeichen importieren/exportieren* und wählen Sie *Importieren*. Alle Lesezeichen in der XML-Datei werden als Benutzer-Lesezeichen importiert	Klicken Sie mit der rechten Maustaste auf den Eintrag *Räumliche Lesezeichen* oder einen Unterordner (Benutzer oder Projekt) oder eine Gruppe, um festzulegen, wohin die Lesezeichen importiert werden sollen, und wählen Sie dann *Räumliche Lesezeichen importieren…*. Wenn Sie den Eintrag *Räumliche Lesezeichen* auswählen, werden die Lesezeichen zu *Benutzer Lesezeichen* hinzugefügt.
Lesezeichen bearbeiten	Sie können ein Lesezeichen ändern, indem Sie die Werte in der Tabelle ändern. Sie können den Namen, die Gruppe, die Ausdehnung und ob es im Projekt gespeichert ist oder nicht, bearbeiten.	Klicken Sie mit der rechten Maustaste auf das gewünschte Lesezeichen und wählen Sie *Räumliches Lesezeichen bearbeiten…*, wodurch der *Lesezeichenbearbeitung* Dialog geöffnet wird. Hier können Sie jeden Aspekt des Lesezeichens neu definieren, als ob Sie es zum ersten Mal erstellen würden. Sie können die Lesezeichen auch mittels Drag-and-drop zwischen Ordnern (Benutzer und Projekt) und Unterordnern (Gruppen) verschieben.

Sie können den *Lesezeichenbearbeitung* Dialog auch öffnen, indem Sie im Bedienfeld *Räumliche Lesezeichenverwaltung* mit der rechten Maustaste auf das gewünschte Lesezeichen klicken. Darüber hinaus ist es möglich, zu einem Lesezeichen zu zoomen, indem Sie den Namen des Lesezeichens in das *Suchen* Feld in der Statusleiste eingeben.

7.1.6 Dekorationen

Unter den Begriff Dekorationen fallen Gitter, Titelbeschriftung, Copyright-Beschriftung, Bild, Nordpfeil, Maßstabsbalken und Layoutausdehnung. Sie werden z.B. verwendet, um den Informationsgehalt einer Karte zu erhöhen oder um sie lesbarer zu machen.

Gitter

Die Dekoration *Gitter* ermöglicht das Hinzufügen eines Koordinatengitters und von Koordinatenbeschriftungen zur Kartenansicht.

1. Wählen Sie den Menüpunkt *Ansicht* ► *Dekorationen* ► *Gitter…* um den Dialog zu öffnen.

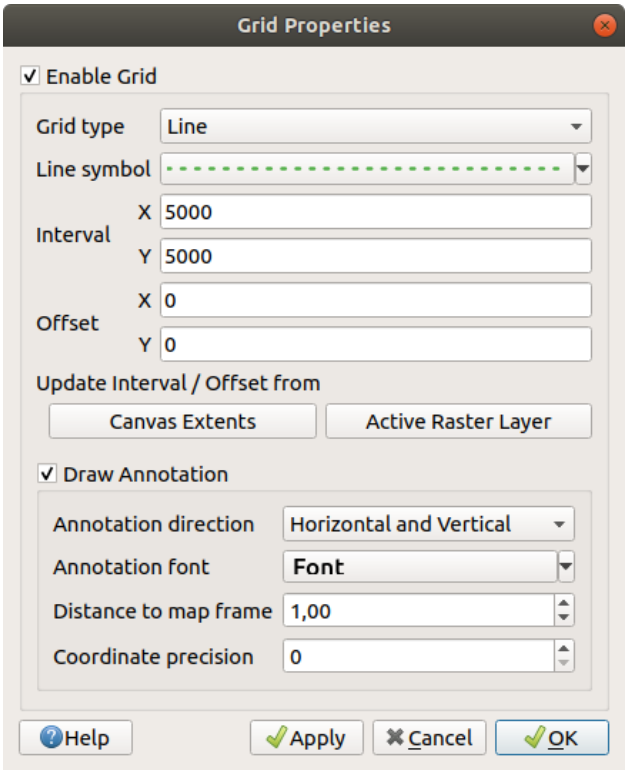


Abb. 7.5: Der Gitter Dialog

2. Markieren Sie das Kontrollkästchen *Gitter aktivieren* und setzen Sie die Gitterdefinitionen entsprechend Ihren Bedürfnissen:
   - *Gittertyp*: die Optionen sind *Linie* oder *Markierung*
   - Das zum Gittertyp gehörende *Liniensymbol* oder *Markierungssymbol*, das zur Darstellung der Gittermarken verwendet werden soll
   - Einen *Intervall* die *X*- und die *Intervall Y*-Achse für den Abstand zwischen den Gittermarkierungen (in Karteneinheiten)
   - Einen *Versatz* für *X* und *Y* für den Abstand der Gittermarkierungen von der linken, unteren Ecke der Kartenansicht (in Karteneinheiten)

- Sie können die Intervall- und Versatz-Parameter festlegen, indem Sie auf die darunter befindlichen Schaltflächen klicken:
  - *Anzeigeausdehnung*: erzeugt ein Raster mit einem Intervall, das ungefähr 1/5 der Kartenansicht ausmacht
  - *Aktiven Rasterlayer* verwendet die Auflösung des aktiven Rasterlayers

3. Aktivieren Sie *Beschriftung zeichnen*, um die Koordinaten der Gittermarkierungen anzeigen zu lassen, wobei Sie folgende Einstellungs-Optionen haben:
   - *Beschriftungsausrichtung*: legt fest, wie die Beschriftungen relativ zu ihrer Gitterlinie platziert werden sollen. Die Möglichkeiten sind:
     - *Horizontal* oder *Vertikal*, jeweils für alle Beschriftungen
     - *Horizontal und Vertikal*, d.h. jede Beschriftung ist parallel zu der Gittermarkierung, auf die sie sich bezieht
     - *Grenzlinienrichtung*, d.h. die Beschriftungen folgen der Begrenzung der Kartenansicht und stehen senkrecht zu der jeweiligen Gittermarkierung, auf die sie sich beziehen
   - *Beschriftungsschriftart*: Textformatierung, Puffer, Schatten, … unter Verwendung des *Schriftauswahl Werkzeugs*
   - *Abstand zum Kartenrahmen*: Abstand zwischen Beschriftung und den Grenzen der Kartenansicht; hilfreich beim *Exportieren der Kartenansicht* z.B. in ein Bildformat oder ein PDF, und um zu vermeiden, dass Anmerkungen auf den „Papier“-Grenzen liegen
   - *Koordinatengenauigkeit*
4. Klicken Sie auf *Anwenden*, um zu überprüfen, ob es wie erwartet aussieht, oder auf *OK*, wenn Sie zufrieden sind.

Titelbeschriftung

Die *Titelbeschriftung* ermöglicht es Ihnen, Ihre Karte mit einem **Titel** zu versehen.

Um eine Titelbeschriftung einzufügen, gehen Sie wie folgt vor:

1. Wählen Sie den Menüpunkt *Ansicht* ► *Dekorationen* ► *Titelbeschriftung…* um den Dialog zu öffnen.

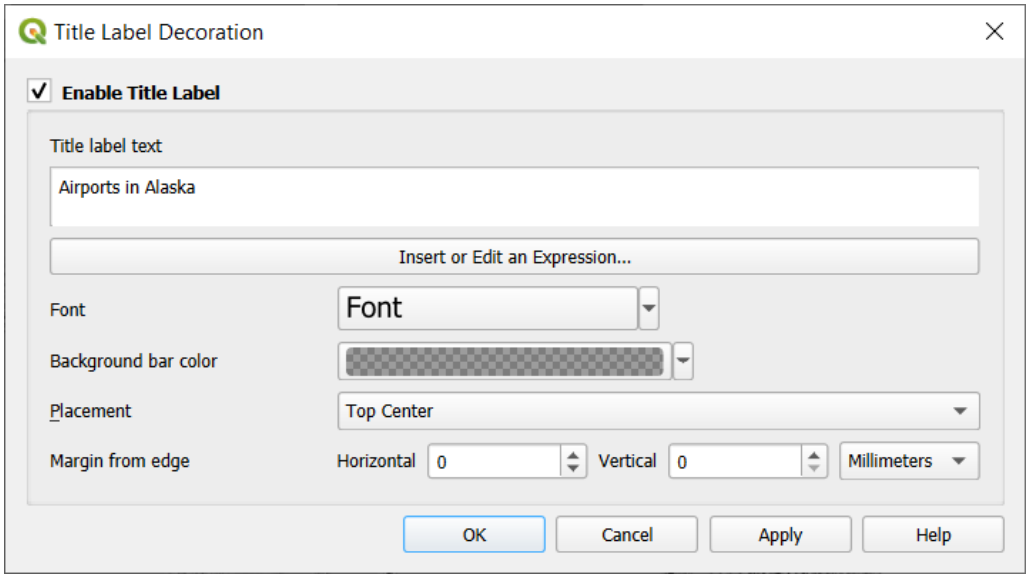


Abb. 7.6: Der Titelbeschriftung Dialog

2. Aktivieren Sie das Kontrollkästchen *Titelbeschriftung*.
3. Geben Sie den Text ein, den Sie auf der Karte platzieren möchten. Über die Schaltfläche *Einen Ausdruck einfügen oder bearbeiten …* können Sie den Text dynamisch gestalten.
4. Wählen Sie die *Schriftart* für den Titel mit dem *Schriftauswahl-Werkzeug* mit allen Funktionen der QGIS *Textformatierung*.
5. Optional können Sie eine *Farbe* für einen *Hintergrundbalkenfarbe* auswählen.
6. Wählen Sie die *Platzierung* der Beschriftung in der Kartenansicht; die Optionen sind: *Oben links*, *Oben Mitte* (Standard), *Oben rechts*, *Unten links*, *Unten Mitte* und *Unten rechts*.
7. Optional können Sie einen horizontalen und/oder vertikalen *Rand von Kante* festlegen; diese Werte können in **Millimetern**, **Pixeln** oder als **Prozentsatz** der Breite oder Höhe der Kartenansicht angegeben werden.
8. Klicken Sie auf *Anwenden*, um zu überprüfen, ob es wie erwartet aussieht, oder auf *OK*, wenn Sie zufrieden sind.

Urheberrechtshinweis

Der *Urheberrechtshinweis* kann verwendet werden, um Ihre Karte mit einer urheberrechtlichen Kennzeichnung zu versehen.

Um diesen Hinweis hinzuzufügen, gehen Sie wie folgt vor:

1. Wählen Sie den Menüpunkt *Ansicht* ► *Dekorationen* ► *Urheberrechtshinweis…*, um den Dialog zu öffnen.

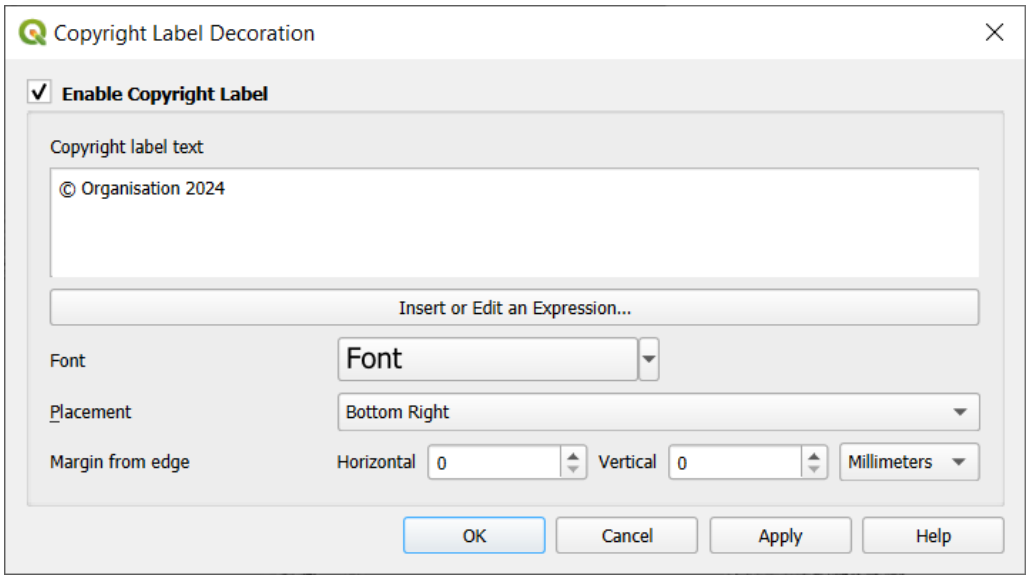


Abb. 7.7: Der Urheberrechtshinweis Dialog

2. Aktivieren Sie das Kontrollkästchen *Urheberrechtshinweis*.
3. Geben Sie den Copyright-Text ein, den Sie auf der Karte platzieren möchten. Über die Schaltfläche *Einen Ausdruck einfügen oder bearbeiten…* können Sie ihn dynamisch gestalten.
4. Wählen Sie die *Schriftart* für den Titel mit dem *Schriftauswahl-Werkzeug* mit allen Funktionen der QGIS *Textformatierung*.
5. Wählen Sie die *Platzierung* der Beschriftung in der Kartenansicht; die Optionen sind: *Oben links*, *Oben Mitte*, *Oben rechts*, *Unten links*, *Unten Mitte* und *Unten rechts* (Standard).
6. Optional können Sie einen horizontalen und/oder vertikalen *Rand von Kante* festlegen; diese Werte können in **Millimetern**, **Pixeln** oder als **Prozentsatz** der Breite oder Höhe der Kartenansicht angegeben werden.

7. Klicken Sie auf *Anwenden*, um zu überprüfen, ob es wie erwartet aussieht, oder auf *OK*, wenn Sie zufrieden sind.

Bild Dekoration

Die *Bild…* Dekoration ermöglicht es Ihnen, ein Logo, eine Legende oder eine andere Abbildung auf der Kartenansicht hinzuzufügen.

Um ein Bild hinzuzufügen, gehen Sie wie folgt vor:

1. Wählen Sie den Menüpunkt *Ansicht*► *Dekorationen* ► *Bild …* , um den Dialog zu öffnen.

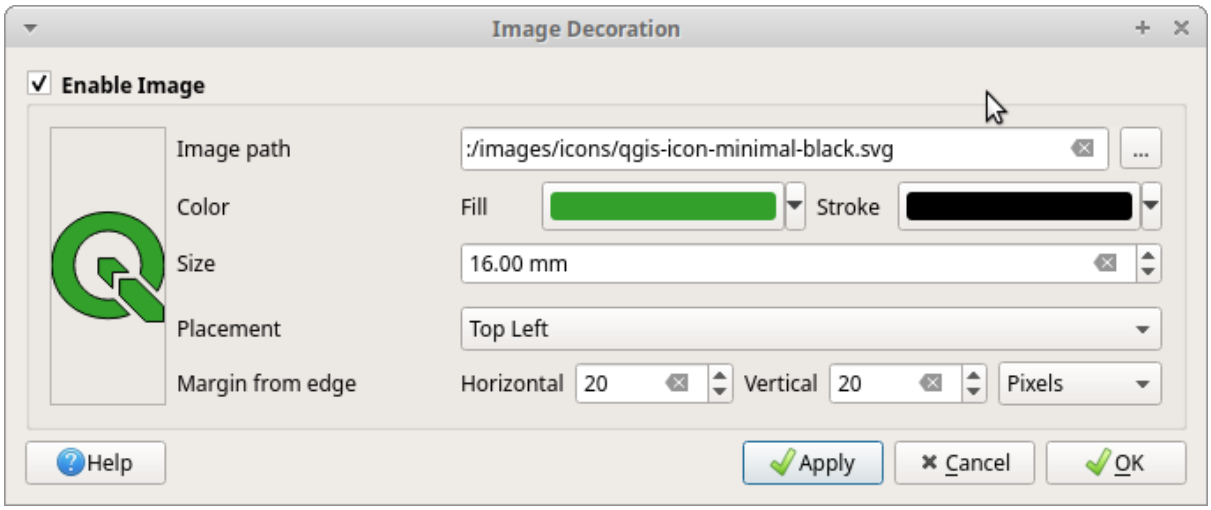


Abb. 7.8: Der Bild Dekoration Dialog

2. Aktivieren Sie das Kontrollkästchen *Bild aktivieren*.
3. Wählen Sie ein Bitmap- (z.B. png oder jpg) oder SVG-Bild mit der Schaltfläche … Durchsuchen aus.
4. Wenn Sie eine SVG-Datei gewählt haben, können Sie auch eine Farbe für die *Füllung* oder den *Strich* (Umriss) festlegen; für Bitmap-Bilder sind die Farbeinstellungen deaktiviert.
5. Mit der Einstellung *Größe* können Sie die Breite des ausgewählten Bildes in mm festlegen.
6. Legen Sie mit dem Auswahlmenü *Platzierung* fest, wo Sie das Bild auf der Kartenansicht platzieren möchten; die Standardposition ist *Unten links*.
7. Legen Sie die Werte für den *Rand von der Kante* fest, jeweils getrennt für *Horizontal* und *Vertikal*. Diese Werte können in **Millimetern**, **Pixeln** oder als **Prozentsatz** der Breite oder Höhe der Kartenansicht angegeben werden.
8. Klicken Sie auf *Anwenden*, um zu überprüfen, ob es wie erwartet aussieht und auf *OK*, wenn Sie zufrieden sind.

Nordpfeil

Die *Nordpfeil* Dekoration ermöglicht es Ihnen, einen Nordpfeil auf der Kartenansicht hinzuzufügen.

Um einen Nordpfeil hinzuzufügen, gehen wie folgt vor:

1. Wählen Sie den Menüpunkt *Ansicht* ► *Dekorationen* ► *Nordpfeil…* um den Dialog zu öffnen.

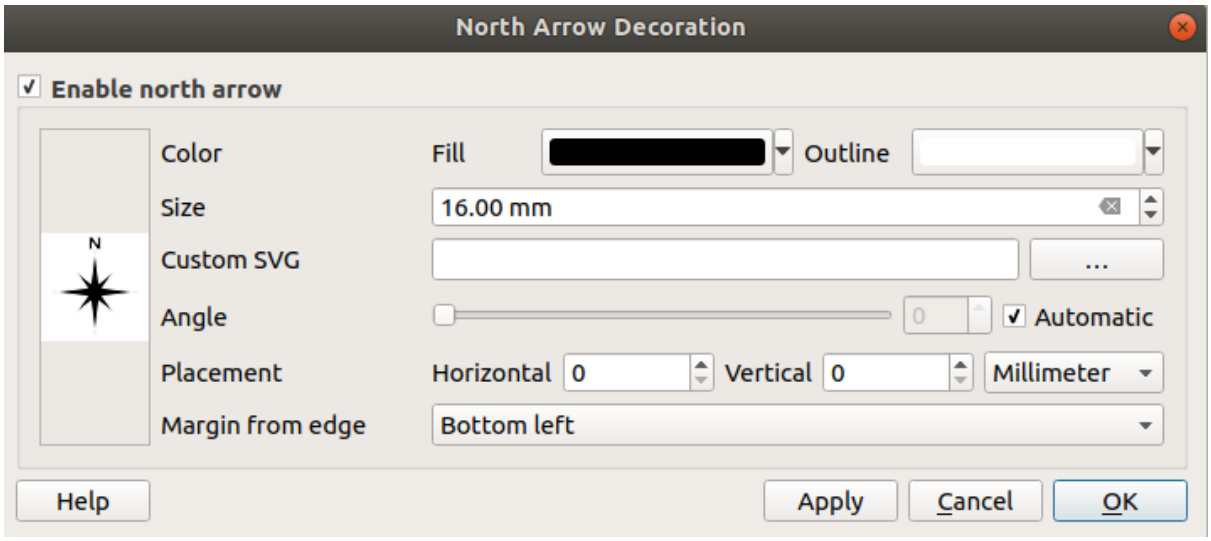


Abb. 7.9: Der Nordpfeil Dialog

2. Aktivieren Sie das Kontrollkästchen *Nordpfeil aktivieren*.
3. Optional können Sie die Farbe und Größe ändern oder eine benutzerdefinierte SVG für den Nordpfeil auswählen.
4. Optional können Sie den Winkel ändern oder *Automatisch* wählen, um die Richtung von QGIS steuern zu lassen.
5. Wählen Sie die *Platzierung* mit dem entsprechenden Auswahlmenü.; Standardwert ist „Unten links“.
6. Optional können Sie einen horizontalen und/oder vertikalen *Rand von Kante* festlegen; diese Werte können in **Millimetern**, **Pixeln** oder als **Prozentsatz** der Breite oder Höhe der Kartenansicht angegeben werden.
7. Klicken Sie auf *Anwenden*, um zu überprüfen, ob es wie erwartet aussieht und auf *OK*, wenn Sie zufrieden sind.

Maßstab

Mit der *Maßstab…* Dekoration können Sie einen einfachen Maßstabsbalken zur Kartenansicht hinzufügen. Sie können den Stil und die Platzierung, sowie die Beschriftung des Balkens bestimmen. Der Maßstabsbalken berücksichtigt die aktive *Entfernungseinheit des Projekts*, wie sie in *Projekteigenschaften* ► *Allgemein* ► *Einheiten für Entfernungsmessung* eingestellt sind.

Um einen Maßstabsbalken hinzuzufügen, gehen sie wie folgt vor:

1. Wählen Sie den Menüpunkt *Ansicht* ► *Dekorationen* ► *Maßstab…* um den Dialog zu öffnen.

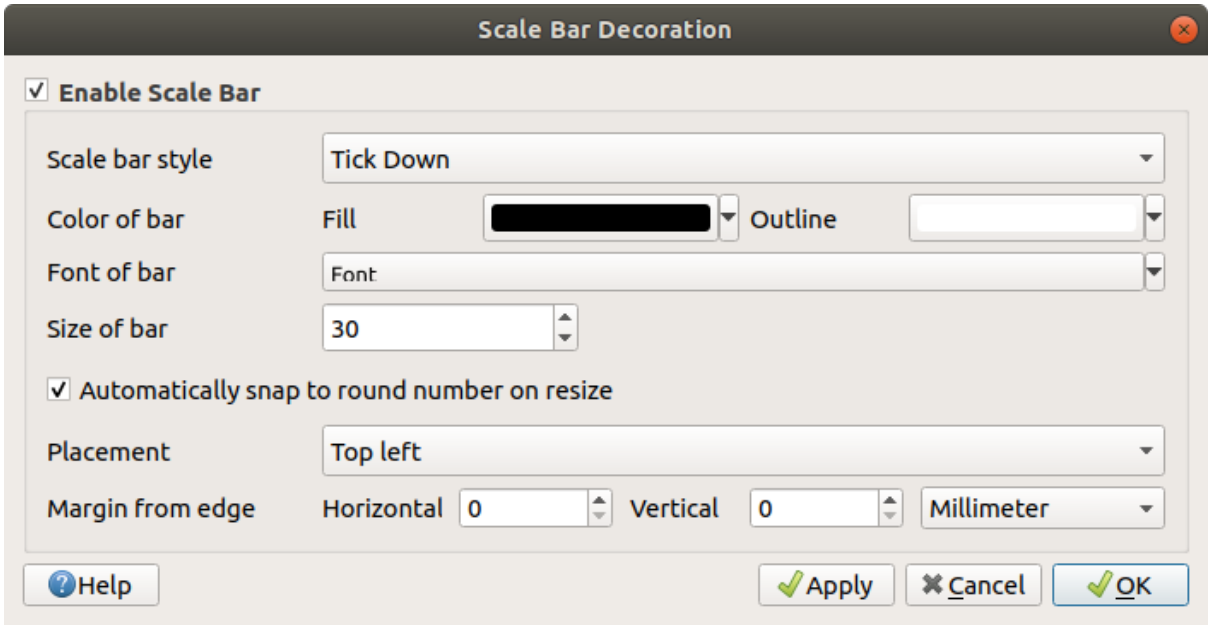


Abb. 7.10: Der Maßstab Dialog

2. Aktivieren Sie das Kontrollkästchen *Aktiviere Maßstab*.
3. Wählen Sie einen Stil aus dem Auswahlmenü *Maßstabsstil* .
4. Wählen Sie die *Balkenfarbe* , indem Sie eine Füllfarbe (Standard: schwarz) und eine Umrissfarbe (Standard: weiß) auswählen; die Füllung und der Umriss des Maßstabsbalkens können mit dem Auswahlmenü, jeweils rechts neben der Farbeingabe, auch transparent gemacht werden.
5. Wählen Sie die Schriftart für den Maßstabsbalken aus dem Kombinationsfeld *Balkenschriftart* aus.
6. Stellen Sie die *Größe des Maßstabs* ein.
7. Standardmäßig ist die Option *Bei Größenänderung automatisch runde Zahlen einstellen* aktiviert, um leicht lesbare Werte anzuzeigen.
8. Legen Sie die *Platzierung* mit den entsprechenden Auswahlmenü fest; Standardwert ist „Oben links“.
9. Optional können Sie einen horizontalen und/oder vertikalen *Rand von Kante* festlegen; diese Werte können in **Millimetern**, **Pixeln** oder als **Prozentsatz** der Breite oder Höhe der Kartenansicht angegeben werden.
10. Klicken Sie auf *Anwenden*, um zu überprüfen, ob es wie erwartet aussieht, oder auf *OK*, wenn Sie zufrieden sind.

Layoutausdehnung

Die *Layoutausdehnung…* fügt die Ausdehnungen von *Kartenelement(en)* aus Drucklayouts zur Kartenansicht hinzu. Wenn diese Funktion aktiviert ist, werden die Ausdehnungen aller Kartenelemente von allen Drucklayouts mit einem leicht gepunkteten Rahmen angezeigt, der mit dem Namen des Drucklayouts und des Kartenelements beschriftet ist. Sie können den Stil und die Beschriftung der angezeigten Ausdehnungen steuern. Diese Dekoration ist z. B. nützlich, wenn Sie die Positionierung von Kartenelementen wie Beschriftungen optimieren und den tatsächlich sichtbaren Bereich von Drucklayouts kennen müssen.

Abb. 7.11: Beispiel für die Anzeige von Layoutausdehnungen in einem QGIS-Projekt mit zwei Drucklayouts. Das Drucklayout mit dem Namen „Sights“ enthält zwei Kartenelemente, während das andere Drucklayout ein Kartenelement enthält.

Um die Layerausdehnung(en) hinzuzufügen, gehen Sie wie folgt vor:

1. Wählen Sie *Ansicht* ► *Dekorationen* ► *Layoutausdehnung*, um den Dialog zu öffnen.

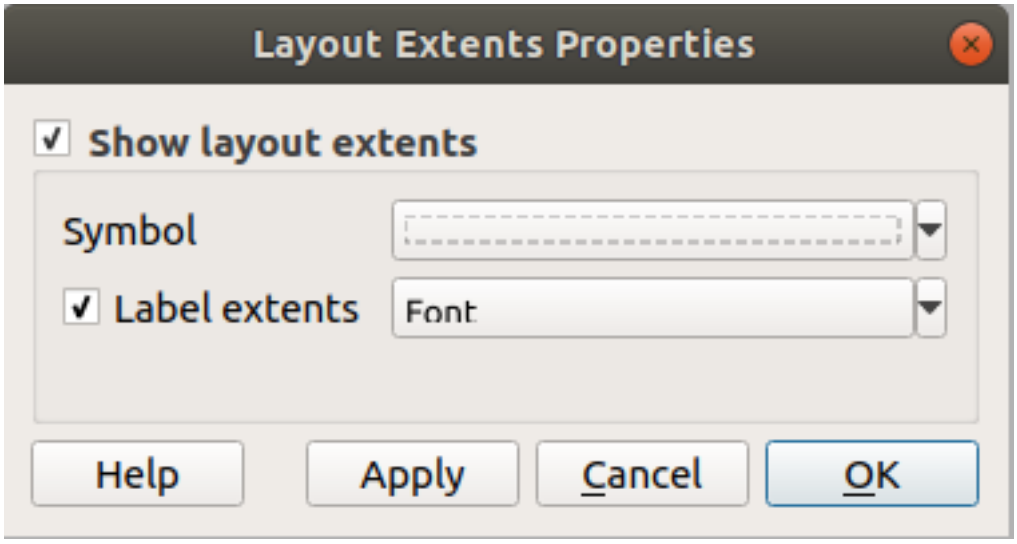


Abb. 7.12: Der Layerausdehnung Dialog

2. Aktivieren Sie das Kontrollkästchen *Layoutausdehnung anzeigen*.

3. Optional können Sie das Symbol und die Beschriftung der Ausdehnung ändern.
4. Klicken Sie auf *Anwenden*, um zu überprüfen, ob es wie erwartet aussieht und auf *OK*, wenn Sie zufrieden sind.

> **Tipp**
>
> **Dekorationseinstellungen**
>
> Wenn Sie eine QGIS-Projektdatei speichern, werden alle Änderungen, die Sie an Gitter, Nordpfeil, Maßstabsleiste, Copyright und Layout- Ausdehnung vorgenommen haben, im Projekt gespeichert und beim nächsten Laden des Projekts wiederhergestellt.

7.1.7 Anmerkungen hinzufügen

Anmerkungen stellen eine Möglichkeit dar, in der Kartenansicht Informationen bereitzustellen, die nicht von den Layern geliefert werden können. Im Gegensatz zu den *Beschriftungen, die aus Attributen des Layers erstellt werden*, sind Anmerkungen unabhängige Elemente, die in der Projektdatei selbst gespeichert sind.

In QGIS gibt es zwei Kategorien von Anmerkungen:

- **Feature-Anmerkungen**: Hierbei handelt es sich um georeferenzierte Elemente vom Typ Text, Marker, Linie oder Polygon, die in einem speziellen Layer („Beschriftungslayer“) gespeichert werden. Sie sind an eine bestimmte, geografische Position gebunden, d. h., wenn Sie Ihre Karte verschieben, den Maßstab oder die Projektion ändern, werden Ihre Anmerkungen an der Stelle bleiben, an der Sie sie gezeichnet haben.
- **Sprechblasen-Anmerkungen**: Dies sind Anmerkungen in Form von HTML oder Formularen innerhalb einer Sprechblase. Die Sichtbarkeit dieser Anmerkungen kann mit einem Layer verknüpft werden und wird dann mit diesem sichtbar/unsichtbar gemacht. Die Größe hängt vom Maßstab der Kartenansicht ab und ihre Position kann verankert werden.

> **Tipp**
>
> **Anmerkungen mit der Karte exportieren**
>
> Sie können die Anmerkungen zu Ihrer Karte in verschiedenen Formaten drucken/exportieren:
>
> - Werkzeuge für den Export der Kartenansichten finden Sie im Menü *Projekt*
> - Beim *Drucken von Layouts* müssen Sie die Option *Kartenelemente zeichnen* in den Elementeigenschaften des Kartenelements aktivieren sein

Die *Anmerkungswerkzeugleiste* bietet eine Reihe von Werkzeugen zum Erstellen und Bearbeiten von beiden Arten von Annotationen.

Tab. 7.3: Die Funktionen der Anmerkungswerkzeugleiste

Werkzeug	Funktion	Geltungsbereich
Neuer Beschriftungslayer	Erstellt einen neuen Layer zum Speichern von Anmerkungen	*Feature annotations*
Eigenschaften des Hauptbeschriftungslayers	Einstellungen für den Hauptbeschriftungslayer	
Beschriftungen modifizieren	Auswählen, Verschieben, Ändern der Größe und Ändern der Symbologieeigenschaften von Anmerkungen	
Erzeuge Polygon-Beschriftung	Erstellt Anmerkungen als Polygon-Feature	
Erzeuge Linien-Beschriftung	Erstellt Anmerkungen als Linien-Feature	

Fortsetzung auf der nächsten Seite

Tab. 7.3 – Fortsetzung der vorherigen Seite

Werkzeug	Funktion	Geltungsbereich
Erzeuge Markierungsbeschriftung	Erstellt Anmerkungen als Punkt-Feature	*Balloon annotations*
Erzeuge Textbeschriftung an Punkt	Create an annotation as a text label anchored at a point	
Erzeuge Textbeschriftung entlang einer Linie	Erstellt Anmerkungen als geschwungenen Text entlang eines Linien-Objekts	
Create Text Annotation in Rectangle	Create an annotation as a text label in a rectangular text box	
Erzeuge Bildbeschriftung	Create an annotation representing an `.SVG` or raster image	
HTML-Beschriftung	Select and create annotation with a `HTML` formatted text	
Formular-Beschriftung	Auswählen und Erstellen von Anmerkungen, die die Attribute eines Vektor-Layers in einer benutzerdefinierten Formulardatei anzeigen	

Bemerkung

Ab QGIS 3.40 ist es nicht mehr möglich, Ballon-Anmerkungen vom Typ SVG oder Text zu erstellen. Beim Laden alter Projekte werden alle diese Anmerkungen automatisch in die neuen Elementtypen „Bild“ und „Rechteckige Textanmerkung“ konvertiert.

Feature-Anmerkungen

Feature-Anmerkungen werden in **Beschriftungslayern** gespeichert. Im Gegensatz zu herkömmlichen Layern ist ein Beschriftungslayer nur im aktuellen Projekt verfügbar und kann Features verschiedener Typen (Text, Marker, Linie, Polygon) enthalten. Dem Layer sind keine Attribute und keine Symbolik zugeordnet, stattdessen kann jedes einzelne Feature über das Bedienfeld *Layergestaltung* symbolisiert werden.

In QGIS gibt es zwei Arten von Beschriftungslayern:

- Ein gewöhnlicher *Beschriftungslayer*: Sie können ihn mit dem Werkzeug Neuer Beschriftungslayer erstellen. Er wird wie jeder andere Layer im *Layer* Bedienfeld aufgelistet, sodass Sie z. B. die Transparenz festlegen können oder bestimmen können über und unter welchen Layern er liegt. Mit einem Doppelklick auf den Layer können Sie die Eigenschaften aufrufen.
- Der *Hauptbeschriftungslayer*: Standardmäßig werden hier die Anmerkungen gespeichert, wenn kein anderer Beschriftungslayer im Projekt vorhanden ist oder beim Erstellen der Anmerkungen ausgewählt war. Dieser Layer wird immer ganz oben auf der Karte gezeichnet und wird nicht im *Layer* Bedienfeld angezeigt. Mit dem Eintrag *Eigenschaften des Hauptbeschriftungslayers* in der *Anmerkungswerkzeugleiste* können Sie den Eigenschaften-Dialog öffnen.

Beschriftungslayer-Eigenschaften

Der Eigenschaftsdialog eines Beschriftungslayers bietet die folgenden Reiter:

- *Information*: ein schreibgeschützter Dialog, der einen nützlichen Überblick über die Informationen und Metadaten des aktuellen Layers bietet. Dazu gehören die Ausdehnung des Layers, die Anzahl der Elemente pro Beschriftungstyp, die Gesamtzahl, KBS-Details, …
- *Quelle*: legt allgemeine Einstellungen für den Beschriftungslayer fest. Sie können hier:
  - Den *Layernamen* festlegen, der benutzt wird, um den Layer innerhalb des Projekts zu identifizieren (z. B. im *Layer* Bedienfeld, bei Aussrücken, …).

  - *Zugewiesenes Koordinatenbezugssystems (KBS)*: Sie können das KBS des Layers ändern, indem Sie ein kürzlich verwendetes in der Dropdown-Liste auswählen oder auf die Schaltfläche KBS auswählen klicken (siehe *Auswahl des Koordinatenbezugssystems*).

- *Darstellung*:
  - Mit den Einstellungen *Minimum (exklusiv)* und *Maximum (inklusive)* können Sie einen Bereich festlegen, innerhalb dem die Beschriftungen angezeigt werden, außerhalb dieses Bereichs werden sie ausgeblendet. Mit der Schaltfläche Auf aktuellen Kartenmaßstab setzen können Sie den aktuellen Maßstab der Kartenansicht übernehmen. Siehe *Einstellen des Sichtbarkeitsmaßstabs* für weitere Informationen.
  - *Verknüpfter Layer*: Ermöglicht es Ihnen, optional einen verknüpften Sichtbarkeitslayer für den Annotationslayer festzulegen. Wenn diese Option aktiviert ist, werden die Annotationen nur dann gezeichnet, wenn der verknüpfte Layer in der Karte sichtbar ist.
  - *Deckkraft*: Mit diesem Werkzeug können Sie den darunter liegenden Layer in der Kartenansicht sichtbar machen. Verwenden Sie den Schieberegler, um die Sichtbarkeit Ihres Beschriftungslayers an Ihre Bedürfnisse anzupassen. Sie können den Prozentsatz der Sichtbarkeit auch im Menü neben dem Schieberegler genau festlegen.
  - *Mischmodus*: Mit diesen Werkzeugen können Sie spezielle Rendering-Effekte erzielen: die Pixel von sich überlagernden Layern werden durch die verwendete Einstellungen gemischt.
  - Mit dem *Zeicheneffekte* Knopf können Sie *Zeicheneffekte* auf alle Layerobjekte anwenden.

Einige dieser Optionen sind über die *Symbologie* Eigenschaften der Beschriftungselemente erreichbar.

Interaktion mit Objekten

Die Feature-Anmerkungen verfügen je nach Typ über spezielle Tools zur Erstellung:

Tab. 7.4: Erstellen von Feature-Anmerkungen

Familie / basierend auf	Anmerkungs-Werkzeug	Vorgehen
Geometrie	Erzeuge Polygon-Beschriftung	Dies entspricht der Digitalisierung eines Punkt-, Linien- oder Polygon-Vektorobjekts. Beim Erstellen von Anmerkungs-Elementen gelten alle üblichen Tastenkombinationen zum Erstellen von Objekten. Eine Linien- oder Polygon-Anmerkung wird durch Klicken mit der linken Maustaste für jeden Stützpunkt und anschließendes Klicken mit der rechten Maustaste zum Abschließen der Form gezeichnet. Während des Zeichnens kann das Einrasten aktiviert werden. Mit der *Erweiterte Digitalisierung*-Werkzeugleiste können Sie Stützpunkt präzise platzieren und es können auch die *Geometrie-Zeichenwerkzeuge* (z.B. dei „Laufende Digitalisierung“) verwendet werden.
	Erzeuge Linien-Beschriftung	
	Erzeuge Markierungsbeschriftung	
Text	Erzeuge Textbeschriftung an Punkt	Klicken Sie mit der linken Maustaste auf die Karte.
	Erzeuge Textbeschriftung entlang einer Linie	Klicken Sie mit der linken Maustaste, um der Textgrundlinie Stützpunkte hinzuzufügen, und klicken Sie mit der rechten Maustaste, um den Vorgang zu beenden. Wie oben beschrieben, stehen auch Funktionen zur Digitalisierung von Objekten zur Verfügung.

Fortsetzung auf der nächsten Seite

Tab. 7.4 – Fortsetzung der vorherigen Seite

Familie / basierend auf	Anmerkungs-Werkzeug	Vorgehen
	Erzeuge Textbeschriftung in einem Rechteck	Zeichnen Sie ein Rechteck für den Text, indem Sie zunächst mit der linken Maustaste auf die erste Ecke klicken und dann ein zweites Mal mit der linken Maustaste auf die diagonal gegenüberliegende Ecke klicken.
Bild	Erzeuge Bildbeschriftung	Zeichnen Sie ein Rechteckfeld für das Bild, indem Sie zunächst mit der linken Maustaste auf die erste Ecke klicken und dann ein zweites Mal mit der linken Maustaste auf die diagonal gegenüberliegende Ecke klicken. Wählen Sie dann ein Bild oder eine SVG-Datei aus dem Datei-Explorer aus.

Im Gegensatz zu normalen Layern muss ein Beschriftungslayer nicht aktiv sein, bevor Sie ein Element auswählen können. Verwenden Sie einfach das Beschriftung modifizieren Werkzeug, um mit den einzelnen Feature-Anmerkungen zu interagieren:

- **Auswahl**: Klicken Sie mit der linken Maustaste auf die Beschriftung. **Achtung**: Standardmäßig werden Beschriftungen in der Reihenfolge angezeigt, in der sie erstellt wurden, d. h. neuere Anmerkungen werden über älteren platziert. Möglicherweise müssen Sie den *Z-Index* in den *Symbol*-Eigenschaft von Beschriftungen (über das Bedienfeld *Layergestaltung*) ändern, um die gewünschte Beschriftung auswählen zu können.
- **Verschieben**: Klicken Sie mit der linken Maustaste auf ein ausgewähltes Beschriftungselement, um es zu verschieben. Ein Rechtsklick oder das Drücken der Taste `Esc` bricht die Verschiebung ab, während ein zweiter Linksklick die Verschiebung bestätigt. Die Verschiebung kann auch durch Drücken der Pfeiltasten gesteuert werden:
  - `Umschalt+Pfeiltaste` für große Bewegung
  - `Alt+Pfeiltaste` für eine `1 px` Bewegung
- **Geometrie ändern**: Bewegen Sie den Mauszeiger über eine Anmerkung und violette Quadrate werden an den Knotenpunkten der zugrunde liegenden Geometrie angezeigt. Klicken Sie mit der linken Maustaste auf das Quadrat, verschieben Sie es und klicken Sie erneut. Bei linearen oder polygonalen Anmerkungen wird durch Doppelklicken auf ein Segment ein neuer Stützpunkt hinzugefügt.
- **Löschen**: Wenn Sie die `Entf` Taste oder die `Rücktaste` drücken, wenn eine Anmerkung ausgewählt ist, wird diese Anmerkung gelöscht.
- *Änderung der Symbologie*

Symbologie der Anmerkungen

Die *Symbologie*-Eigenschaften einer ausgewählten Anmerkung werden im Bedienfeld *Layergestaltung* angezeigt. Die folgenden Optionen stehen zur Verfügung:

- Bei Polygon-, Polylinien- und Markierungs-Anmerkungen können Sie das Erscheinungsbild mithilfe aller Funktionen der *Symbol-Eigenschaften* ändern
- Bei textbasierten Anmerkungen hilft Ihnen ein Bereich mit Textbearbeitungswerkzeugen beim Erstellen der anzuzeigenden Zeichenfolge. Diese kann mehrzeilig sein, HTML-Formatierungen enthalten und auf QGIS-Ausdrucksfunktionen zurückgreifen. Darüber hinaus können Sie das Erscheinungsbild mithilfe der vollständigen Funktionen der *Textformatierungs-Eigenschaften* ändern.

  Je nach Art der Anmerkung stehen zusätzliche Optionen zur Verfügung:

  - Für Textanmerkungen an einem Punkt haben Sie die folgenden Möglichkeiten:
    - die *Ausrichtung* (links, mittig oder rechts vom Ankerpunkt) festlegen

        * konfigurieren, ob der Text *Kartenrotation ignorieren* oder *Mit Karte drehen* soll. In beiden Fällen kann ein benutzerdefinierter *Winkel* für die Ausrichtung der Elemente festgelegt werden.
    - Für Textanmerkungen entlang einer Linie können Sie einen *Abstand von der Linie* konfigurieren, wobei verschiedenen Einheiten ausgewählt werden können.
    - Für Textanmerkungen in einem Rechteck haben Sie folgende Möglichkeiten:
        * Legen Sie die *Horizontale Ausrichtung* (links, zentriert, rechts oder Blocksatz) und *Vertikale Ausrichtung* (oben, vertikal zentriert oder unten) des Textes im Rechteckfeld fest.
        * Konfigurieren Sie für das Rechteck den Abstand der *Ränder* in den Einheiten Ihrer Wahl sowie die Farben für den *Rahmen* und den *Hintergrund*.

- Mit einer Bild-Anmerkung können Sie ein *Rasterbild* oder eine *SVG-Grafik* mithilfe des Werkzeugs *Entfernte oder eingebettete Dateiauswahl* einfügen. Die Platzierung und Größe des Bildes kann wie folgt festgelegt werden:
    - *Maßstabsabhängige Größe*: Das Bild wird mit der Karte skaliert und erscheint beim Vergrößern größer und beim Verkleinern kleiner. Die Größe wird durch eine Referenzgröße in Karteneinheiten definiert (z. B. Meter oder Grad, je nach KBS des Layers). Mit *Seitenverhältnis sperren* können Sie die Größe der Anmerkung proportional zur Größe des eingebetteten Bildes beibehalten oder das Bild so strecken, dass es den Umfang der Anmerkung ausfüllt.
    - *Feste Größe*: Die Bild-Anmerkung behält unabhängig vom Kartenmaßstab eine konstante Größe in Bildschirm- oder Ausgabeeinheiten (z. B. Pixel oder Millimeter) bei. Die *Breite* und *Höhe* der Anmerkung können separat eingegeben oder proportional zur Größe des eingebetteten Bildes angepasst werden.
    - *Relativ zur Karte*: Die Bild-Anmerkung wird unabhängig vom Kartenmaßstab immer an derselben Position relativ zu den Grenzen der Kartenfläche gerendert. Die *Breite* und *Höhe* der Anmerkung können separat ausgefüllt oder proportional zur Größe des eingebetteten Bildes angepasst werden.

  Darüber hinaus können Sie die Anzeige von *Rahmen* und *Hintergrund* des Rechteckfeldes aktivieren und diese mithilfe der Eigenschaften von *Füllungen* konfigurieren.

- Für Text an einem Punkt, Text innerhalb eines Rechtecks und Bild-Anmerkungen können Sie *Führungslinie anzeigen* verwenden, wenn der Ankerpunkt des Bildes oder Textes von seiner Standardposition versetzt ist. So erstellen Sie einen Führungslinie für eine Anmerkung:
    1. Wählen Sie das Anmerkungsobjekt aus.
    2. In der Mitte des Rechteckes wird der Ankerpunkt der Führungslinie als sehr kleines, grünes `X` sichtbar. Verschieben Sie diesen Ankerpunkt, indem Sie darauf klicken (nicht halten) und dann erneut, auf die neue Position, klicken.
    3. Drücken Sie in den Anmerkungseigenschaften hinter dem Eintrag *Hinweislinie anzeigen* die Schaltfläche …, um die *Führungslinien-Eigenschaften* zu konfigurieren.

  Tipp

  Wenn Sie eine Sprechblasenführungslinie auf eine Text- oder Bildanmerkung anwenden, erhalten Sie die alte Darstellung der Sprechblasen-Anmerkung.

- Sie können einen *Referenzmaßstab* konfigurieren. Damit legen Sie einen Kartenmaßstab fest, auf den sich Symbol- oder Textgrößen beziehen, die papierbasierte Einheiten (wie Millimeter oder Punkte) verwenden. Die Größen werden entsprechend skaliert, wenn die Karte in einem anderen Maßstab betrachtet wird. So wird z. B. ein Linienmerkmal mit einer Breite von 2 mm im Maßstab 1:2.000 *Referenzmaßstab* mit 4 mm wiedergegeben, wenn die Karte im Maßstab 1:1.000 betrachtet wird.
- Setzen eines *Z-index*: eine Beschriftung mit einem höheren Index wird über Beschriftungen mit einem niedrigeren Index gelegt. Änderungen hier sind sowohl für die Anzeige, als auch für die Auswahl von Beschriftungen hilfreich.
- Änderung der Einstellungen für die *Layer Darstellung*.

Sprechblasen-Anmerkungen

Sie können Sprechblasen-Anmerkungen über das Menü *Bearbeiten* ► *Beschriftung hinzufügen* ► oder über die *Anmerkungswerkzeugleiste* hinzufügen:

- 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.

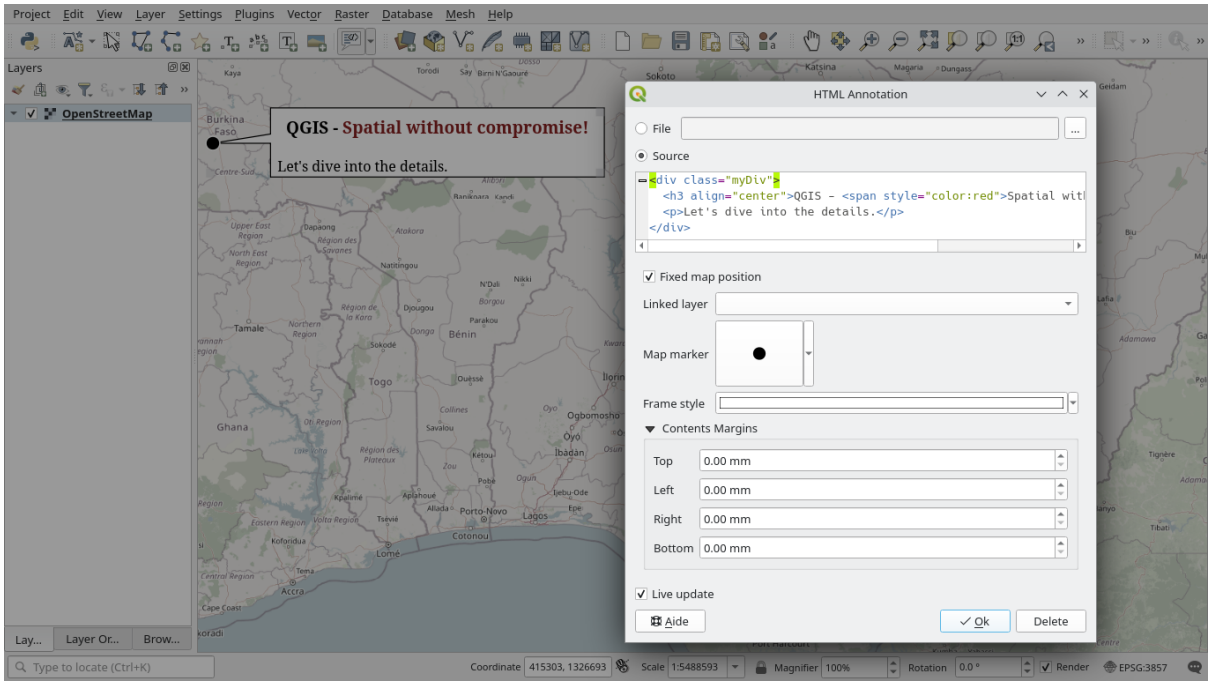


Abb. 7.13: Example of a HTML balloon annotation

Um die gewünschte Sprechblasenbeschriftung hinzuzufügen, wählen Sie das entsprechende Werkzeug aus und klicken dann auf die Kartenansicht. Eine leere Sprechblase wird hinzugefügt. Nach einem Doppelklick darauf öffnet sich ein Dialog mit verschiedenen Optionen. Dieses Dialogfeld ist für alle Beschriftungsarten nahezu identisch:

- *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.
- *Verknüpfter Layer* verbindet die Beschriftung mit einem Layer aus dem Projekt, sodass die Beschriftung nur dann sichtbar ist, wenn dieser Layer sichtbar ist.
- *Kartenmarkierung* legt das Symbol fest, das am Sprechblasen-Anker angezeigt wird (verwendet *QGIS Symbologie*); wird nur angezeigt, wenn *Feste Kartenposition* aktiviert ist.
- *Rahmenstil* legt die Hintergrundfarbe, die Transparenz, die Strichfarbe oder die Breite der Sprechblase mit QGIS-Symbolen fest.
- *Inhaltsränder* legt die inneren Abstände des Beschriftungsrahmens fest.
- ☑ *Laufende Aktualisierung* ermöglicht Ihnen eine Live-Vorschau Ihrer Änderungen.

Wenn ein Sprechblasenwerkzeug aktiv ist, können Sie auch:

- Eine Anmerkung auswählen
- Größe einer Anmerkung anpassen
- Die Sprechblase oder den Anker auf der Kartenansicht verschieben
- Eine Sprechblase löschen: markieren Sie diese und drücken Sie dann entweder die `Entf` Taste oder die `Rücktaste`, oder doppelklicken Sie darauf und drücken Sie dann *Löschen* im Eigenschaften-Dialog.
- Mit einem Rechts-Klick das Kontextmenü öffnen; hier können Sie:
  - *Koordinaten kopieren*: die Koordinaten des Ankers kopieren
  - Mit *Bearbeiten* das Eigenschaften-Fenster der Beschriftung öffnen
  - Die Beschriftung *Löschen*

7.1.8 Messen

Allgemeine Informationen

QGIS bietet vier Möglichkeiten, Geometrien zu messen:

- Interaktives Messen-Werkzeug
- Messen mit dem Feldrechner
- Abgeleitete Messungen mit dem *Objekte abfragen* Werkzeug
- das Vektoranalysewerkzeug: *Vektor* ► *Geometrie-Werkzuge* ► *Geometriespalten exportieren/hinzufügen*

Das Messen funktioniert in projizierten Koordinatensystemen (z.B. UTM) und unprojizierten (z.B. WGS84). Für die ersten drei Messwerkzeuge gelten die folgenden, globalen Projekteinstellungen gleichermaßen:

- Im Gegensatz zu den meisten anderen GIS ist die Standard-Messung ellipsoidisch, unter Verwendung des Projekt-Ellipsoids, das in *Projekt* ► *Eigenschaften…* ► *Allgemein* definiert ist. Dies gilt sowohl für geografische als auch für projizierte Koordinatensysteme.
- Wenn Sie die projizierte/planimetrische Fläche oder Entfernung mit kartesischer Mathematik berechnen wollen, muss das Messellipsoid auf „Keine/Planimetrie“ gesetzt werden (*Projekt* ► *Eigenschaften…* ► *Allgemein*). Wenn für die Daten und das Projekt jedoch ein geografisches (d. h. nicht projiziertes) KBS definiert ist, erfolgt die Flächen- und Entfernungsmessung ellipsoidisch.

Allerdings werden weder das Identifizierungswerkzeug noch der Feldrechner Ihre Daten vor der Messung in das Projekt KBS umwandeln. Wenn Sie dies erreichen wollen, müssen Sie das Vektoranalysewerkzeug verwenden: *Vektor* ► *Geometriewerkzeuge* ► *Geometrieattribute hinzufügen*…. Hier ist die Messung planimetrisch, es sei denn, Sie wählen die ellipsoidische Messung.

Interaktives Messen von Längen, Flächen, Kurs und Winkeln

Das Messen-Werkzeug finden Sie in der Attributwerkzeugleiste. Mit dem Auswahlmenü auf der rechten Seite des Icons können Sie zwischen Linie, Fläche, Kurs und Winkel umschalten. Die Standardeinheit, die im Dialog verwendet wird, ist die, die im Menü *Projekt* ► *Eigenschaften…* ► *Allgemein* eingestellt ist.

Für *Linie messen* und *Fläche messen* kann ausgewählt werden, ob die Messungen *Kartesisch* oder *Ellipsoid* vorgenommen werden sollen. Informationen dazu hält das Werkzeug bereit.

Bemerkung

Konfigurieren des Messwerkzeugs

Während der Messung von Längen oder Flächen öffnet ein Klick auf die Schaltfläche *Konfiguration* am unteren Rand des Messen-Fensters das Menü *Einstellungen* ► *Optionen* ► *Kartenwerkzeuge*. Hier können Sie während der laufenden Messung Einstellungen zur Gummibandfarbe, der Genauigkeit und den Einheiten vornehmen. Sie können auch Ihre bevorzugten Maß- oder Winkeleinheiten wählen, aber bedenken Sie, dass diese Werte im aktuellen Projekt durch die Auswahl im Menü *Projekt* ► *Eigenschaften…* ► *Allgemein* und durch die Auswahl im Messen-Fenster überschrieben werden.

Alle Messmodule verwenden die Fangeinstellungen des Digitalisierungsmodus (siehe Abschnitt *Einrasttoleranz und Suchradius einstellen*). Wenn Sie also genau entlang einer Linie oder um ein Polygon-Feature herum messen wollen, stellen Sie zuerst die Toleranz für das Einrasten des Layers ein. Wenn Sie nun die Messwerkzeuge verwenden, rastet jeder Mausklick (innerhalb der Toleranzeinstellung) an diesem Layer ein.

Linie messen: Mit diesem Werkzeug werden Entfernungen zwischen zwei Punkten gemessen. Jede Segmentlänge sowie die Gesamtsumme werden im Messen-Fenster angezeigt. Hier sehen Sie auch die Koordinaten aller Punkte. Die erste Zeile enthält nur Koordinaten, da sie den Ausgangspunkt darstellt. Mit der Schaltfläche *Kopieren* können Sie alle Linienmessungen in die Zwischenablage kopieren. Wenn Sie auf die Schaltfläche *Konfiguration* klicken, gelangen Sie zu *Messwerkzeugkopiereinstellungen*, wo Sie die *Kopieroptionen* festlegen können. Um die Messung zu beenden, klicken Sie mit der rechten Maustaste.

Mit der Dropdown-Liste neben der Gesamtsumme können Sie die *Messeinheiten* ändern, auch während einer laufenden Messung. Diese Einstellung wird dann für das Messen-Werkzeug beibehalten, bis ein neues Projekt erstellt oder ein anderes Projekt geöffnet wird.

Der Abschnitt *Information* im Fenster erklärt, wie die Berechnungen gemäß den verfügbaren KBS-Einstellungen durchgeführt werden.

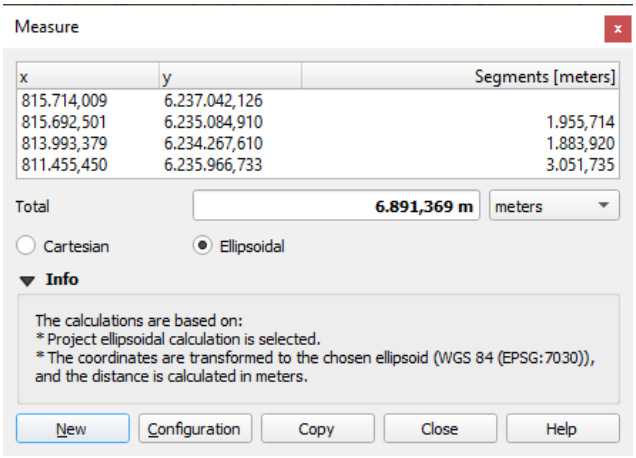


Abb. 7.14: Entfernung messen

Fläche messen: Im Messen-Fenster wird die aufsummierte Flächengröße angezeigt. Mit einem Rechtsklick können Sie das Messen beenden. Auch hier werden Informationen zur Art der Messung zur Verfügung gestellt und es kann in laufenden Messungen die *Flächeneinheit* gewechselt werden.

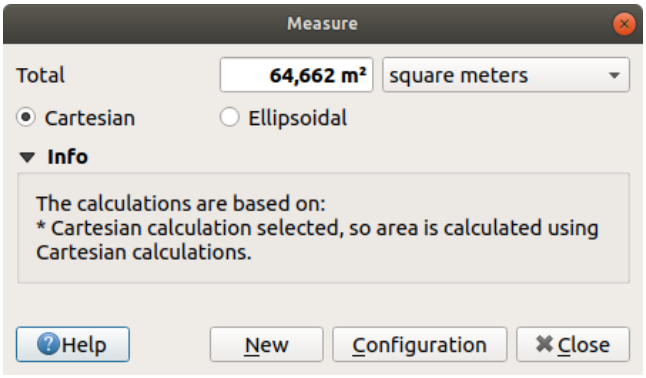


Abb. 7.15: Fläche messen

Kurs messen: Um einen Kurs zu messen, müssen Sie zunächst mit einem Klick einen Punkt setzen. Bewegen Sie dann den Cursor, um auf den zweiten Punkt zu klicken. Das Ergebnis der Messung wird in einem Pop-up-Dialog angezeigt.

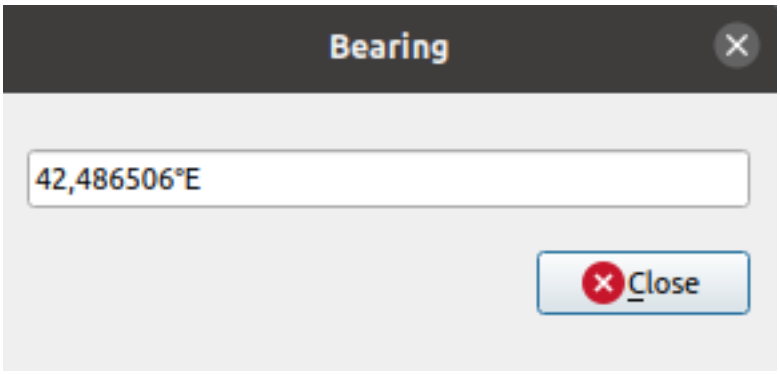


Abb. 7.16: Kurs messen

Winkel messen: Setzten Sie zwei Punkte, um das erste Segment des Winkels zu zeichnen, den Sie messen möchten, und bewegen Sie dann den Cursor, um den gewünschten Winkel zu zeichnen. Das Ergebnis der Messung wird in einem Pop-up-Dialog angezeigt.

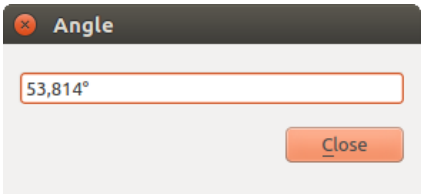


Abb. 7.17: Winkel messen

7.1.9 Weitere Kartenansichten hinzufügen

Es ist möglich, zusätzliche Kartenansichten zu öffnen, deren Darstellung von den jeweils aktuellen Einstellungen im *Layer* Bedienfeld abweichen kann. Um eine neue Kartenansicht hinzuzufügen, gehen Sie zu *Ansicht* ► *Neue Kartenansicht*. Ein neues Fenster öffnet sich, in dem zunächst die Darstellung der Hauptkartenansicht übernommen wird. Sie können so viele Kartenansichten hinzufügen, wie Sie benötigen. Sie können schwebend sein, nebeneinander oder als Reiter übereinander gestapelt werden.

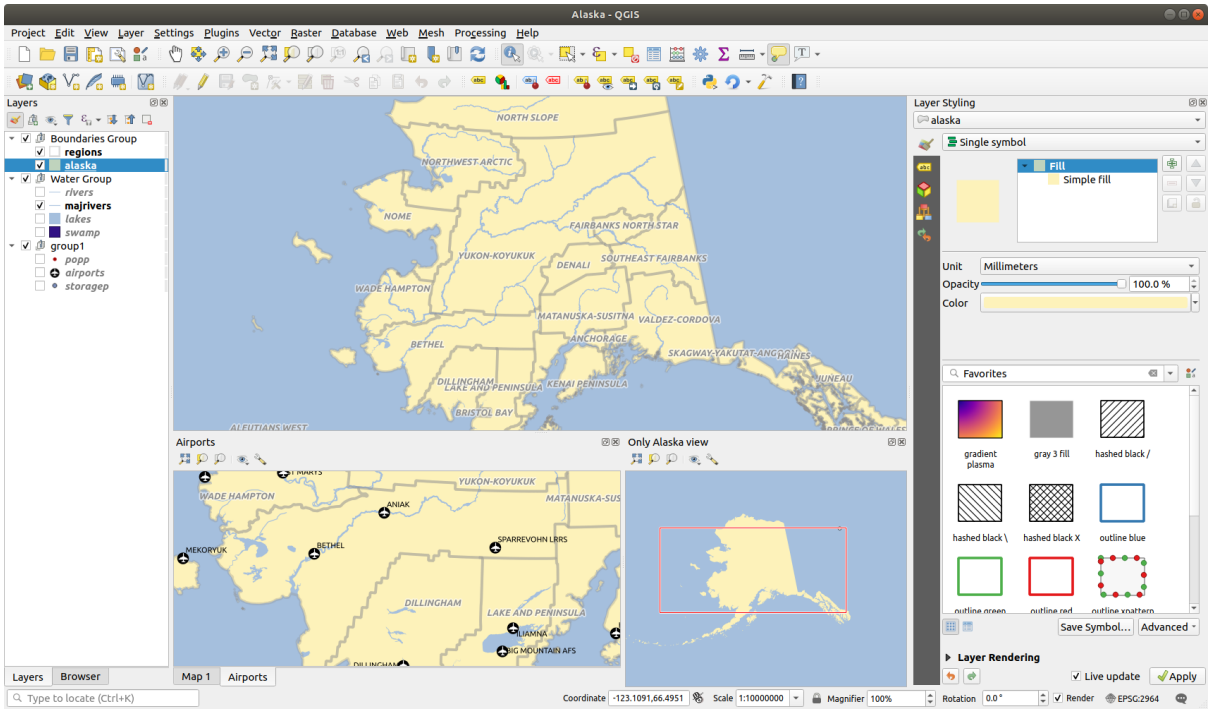


Abb. 7.18: Verschiedene Kartenansichten mit unterschiedlichen Einstellungen

Am oberen Rand jeder Kartenansicht befindet sich eine Werkzeugleiste mit den folgenden Möglichkeiten:

- Volle Ausdehnung, Zu ausgewählten Objekten zoomen und Auf Layer zoomen um in der Ansicht zu navigieren
- Kartenthema der Ansicht setzen: hiermit können Sie das *Kartenthema* zur Anzeige in dieser Kartenansicht auszuwählen. Wenn es auf `(keine)` eingestellt ist, folgt die Darstellung den Änderungen im *Layer* Bedienfeld.
- Ansichtseinstellungen: hier können Sie die Kartenansicht wie folgt konfigurieren:
  - *Ansichtszentrum mit Hauptkarte synchronisieren*: synchronisiert die Mitte der Kartenansichten, ohne den Maßstab zu ändern; so können Sie eine Übersichtskarte oder eine vergrößerte Karte erstellen, die der Mitte der Kartenansicht folgt.
  - *Ansicht mit Auswahl synchronisieren*: das gleiche wie *Zu ausgewählten Objekten zoomen*
  - *Maßstab*
  - *Drehung*
  - *Vergrößerung*
  - *Maßstäbe synchronisieren*: der Maßstab wird mit dem der Hauptkarte synchron gehalten. Ein *Maßstabsfaktor* kann dann angegeben werden, so dass Sie eine Ansicht erhalten, die immer z.B. den zweifachen Maßstab der Hauptkarte hat.
  - *Anmerkungen anzeigen*

- *Cursor-Position anzeigen*
- *Hauptkartenausdehnung anzeigen*
- *Beschriftungen anzeigen*: erlaubt es die Beschriftungen auszublenden, auch wenn sie in den Einstellungen des Layers aktiviert sind
- *Karten-KBS ändern…*
- *Ansicht umbenennen…*

7.1.10 Exportieren der Kartenansicht

Die von Ihnen erstellten QGIS-Projekte können als komplexe *Layouts oder Berichte* in verschiedene Formate exportiert werden. Es ist aber auch möglich, die aktuelle Darstellung in der Kartenansicht direkt zu exportieren, ohne ein zusätzliches Layout zu erstellen. Dieser schnelle „Screenshot“ der Kartenansicht hat einige praktische Funktionen.

Um die aktuelle Darstellung im Kartenfenster zu exportieren, gehen Sie wie folgt vor:

1. Wählen Sie aus der Menüleiste *Projekt* ► *Import/Export*
2. abhängig davon, was Sie exportieren möchten, wählen Sie eine der folgenden Optionen:
   - *Karte als Bild speichern …*
   - *Karte als PDF exportieren…*

Die beiden Export-Dialoge bieten Ihnen verschiedene Einstellungsmöglichkeiten:

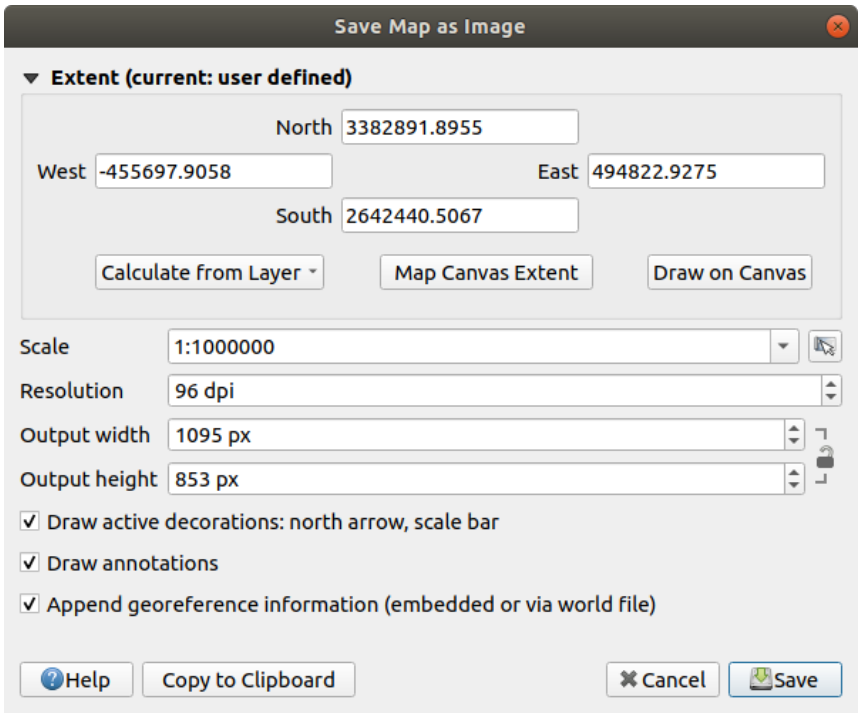


Abb. 7.19: Der Dialog zum Speichern der Kartenansicht als Bild-Datei

1. Wählen Sie die zu exportierende *Ausdehnung*: Optionen sind die aktuellen Kartengrenzen (Standard), die Ausdehnung eines Layers oder eine benutzerdefinierte Ausdehnung, die über der Kartenansicht aufgezogen wird. Die Koordinaten des ausgewählten Bereichs werden angezeigt und können manuell bearbeitet werden.
2. Sie können auch einen *Maßstab* eingeben oder ihn aus der Liste der *Vordefinierten Maßstäbe* auswählen. Wenn Sie den Maßstab ändern, wird die Ausdehnung für den Export vom Zentrum aus angepasst.

3. Geben Sie eine *Auflösung* für die Ausgabe an
4. Kontrollieren Sie die *Ausgabebreite* und *Ausgabehöhe* des Bildes in Pixeln. Sie basieren standardmäßig auf der aktuellen Auflösung und Ausdehnung, können aber angepasst werden und verändern den Ausschnitt der Kartenansicht (vom Zentrum aus). Das Größenverhältnis kann fixiert werden, was besonders praktisch ist, wenn die Ausdehnung auf der Kartenansicht gezeichnet wird.
5. ☑ *Aktive Dekorationen anzeigen*: Wenn Sie *Dekorationen* (Maßstabsleiste, Titel, Gitter, Nordpfeil…) für die Kartenansicht aktiviert haben, werden diese mit exportiert
6. ☑ *Bemerkungen zeichnen*: Alle *Beschriftungen* der Kartenansicht werden mit exportiert
7. ☑ *Geofererenzierungsinformationen anhängen (eingebettet oder als World-Datei)*: Je nach Ausgabeformat wird eine gleichnamige World-Datei (mit der Erweiterung `PNGW` für `PNG`-Bilder, `JPGW` für `JPG`, …) im gleichen Ordner wie Ihr Bild gespeichert. Das `PDF`-Format bettet die Informationen in die PDF-Datei ein.
8. Beim PDF-Export stehen im Dialog *Karte als PDF speichern…* noch weitere Optionen zur Verfügung:

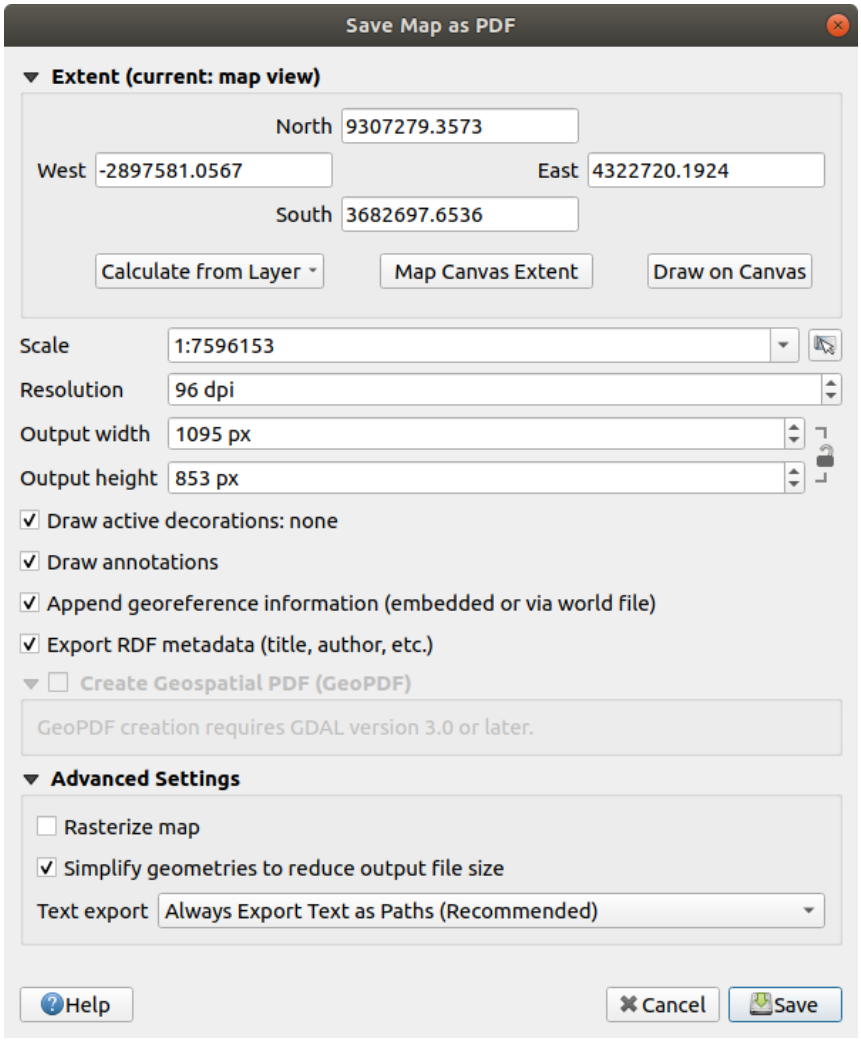


Abb. 7.20: Der Dialog zum Speichern der Kartenansicht als PDF-Datei

- ☑ *RDF-Metadaten exportieren* wie z. B. Titel, Autor, Datum, werden mit exportiert
- ☐ *Geospatial-PDF erzeugen*: Erzeugt eine georeferenzierte PDF-Datei. Sie haben folgende Optionen:
  - Wählen Sie das Geodaten-PDF *Format*

  - *Vektorlayerinformationen einschließen*: wenn diese Option aktiviert ist, werden alle Geometrie- und Attributinformationen von Objekten, die auf der Karte sichtbar sind, in die ausgegebene Geodaten-PDF-Datei einbezogen. ACHTUNG!!! Es werden tatsächlich ALLE Attribute aus den Tabellen ALLER Layer eingeschlossen. Vorsicht also bei sensiblen Daten! Durch die Entfernung des entsprechenden Häckchens unter *Attribute einschließen* kann dieses Verhalten angepasst werden.

> **Bemerkung**
>
> Eine Geospatial-PDF-Datei kann ebenfalls als Datenquelle in QGIS verwendet werden. Weitere Informationen zur Unterstützung von Geospatial-PDF-Dateien in QGIS finden Sie unter https://north-road.com/2019/09/03/qgis-3-10-loves-geospatialpdf/.

- *Karte rastern*
- *Geometrien für kleinere Dateigrößen vereinfachen*: Geometrien werden während des Exports der Karte vereinfacht, indem Knotenpunkte entfernt werden, die sich bei der Exportauflösung nicht deutlich unterscheiden (z.B. wenn die Exportauflösung `300 dpi` beträgt, werden Eckpunkte, die weniger als `1/600 Zoll` auseinander liegen, entfernt). Dies kann die Größe und Komplexität der Exportdatei reduzieren (sehr große Dateien können u. U. in anderen Anwendungen nicht geladen werden).
- Stellen Sie mit der Option *Textexport* ein, ob Textbeschriftungen immer oder vorzugsweise als *Text oder Pfade* exportiert werden sollen.

9. Klicken Sie *Speichern*, um einen Namen und einen Pfand festzulegen

   Beim Exportieren als Bild ist es auch möglich, das erwartete Ergebnis der obigen Einstellungen in die Zwischenablage zu kopieren und die Karte in einer anderen Anwendung einzufügen.

7.2 3D-Kartenansicht

3D-Visualisierungen werden durch die 3D-Kartenansicht unterstützt (auch 3D-Szene genannt). Sie können 3D-Kartenansichten über das Menü *Ansicht* ► *3D-Kartenansichten* ► erstellen, verwalten und öffnen:

1. Mit einem Klick auf *Neue 3D-Kartenansicht* können Sie eine neue 3D-Kartenansicht erstellen. Es erscheint ein schwebendes und andockbares QGIS Bedienfeld (siehe *Die 3D-Kartenansicht*). Diese hat die gleiche Ausdehnung wie die Hauptkartenansicht und bietet eine Reihe von Navigationswerkzeugen, um die Ansicht in eine 3D-Ansicht zu überführen.
2. Wenn Sie auf *3D-Kartenansichten verwalten* klicken, gelangen Sie in die 3D-Kartenansichtsverwaltung. Hier haben Sie die Möglichkeit, 3D-Kartenansichten zu öffnen, zu duplizieren, zu entfernen und umzubenennen.
3. Wenn Sie eine oder mehrere 3D-Kartenansichten erstellt haben, werden diese in *3D-Kartenansichten* aufgelistet. Sie können sie ein- und ausschalten, indem Sie darauf klicken. Die Karten werden beim Speichern des Projekts gespeichert, auch wenn sie nicht aktiv sind.

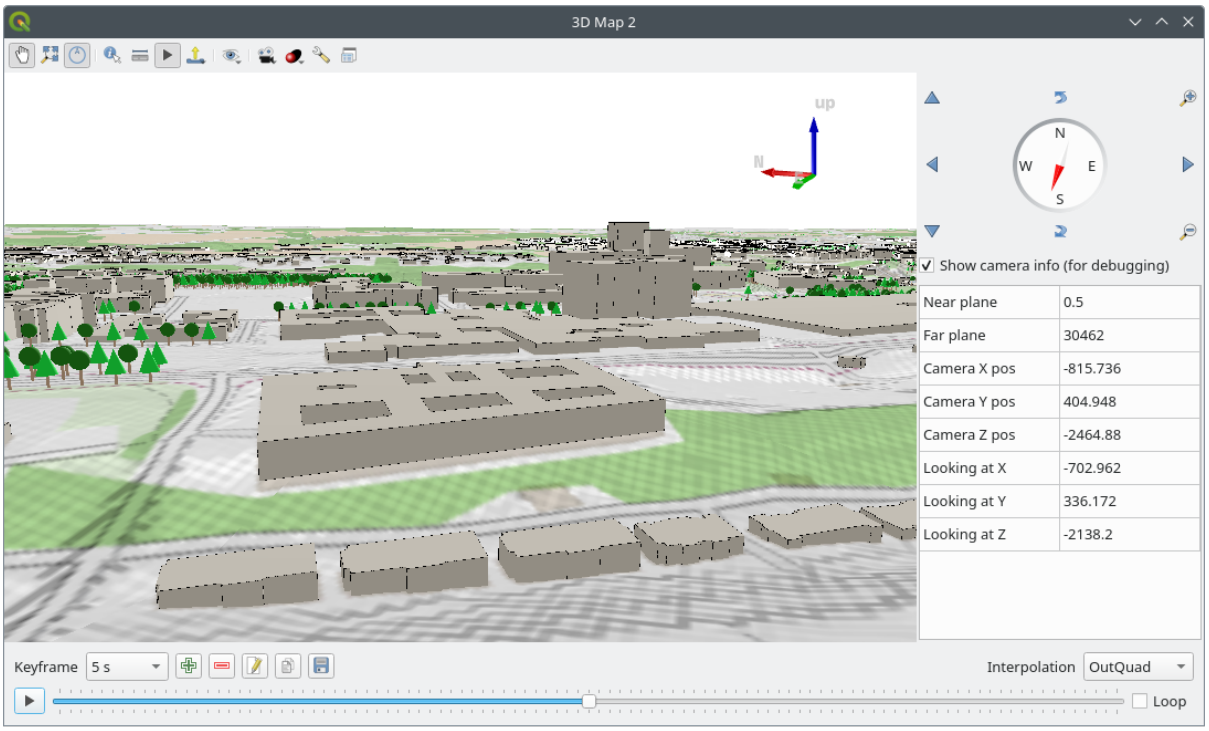


Abb. 7.21: Die 3D-Kartenansicht

Am oberen Rand des Bedienfelds für die 3D-Kartenansicht befinden sich die folgenden Werkzeuge:

- Kamerasteuerung: bewegt die Ansicht, wobei der gleiche Winkel und die gleiche Richtung der Kamera beibehalten werden
- Volle Ausdehnung: ändert die Größe der Ansicht auf die gesamte Ausdehnung des Layers, oder auf die *Referenzausdehnung*, falls gesetzt
- Bildschirmnavigation umschalten: zeigt/versteckt das Navigations-Widget (das die Steuerung der Kartenansicht erleichtern soll)
- Abfragen: liefert Informationen über den angeklickten Punkt des Geländes oder die angeklickten 3D-Elemente – Mehr Details unter *Objekte abfragen*
- 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
- Animation: zeigt/versteckt das *Animationsspieler* Widget
- Exportieren ermöglicht das Speichern der Szene in verschiedene Formate:
  - Als Bild speichern…: exportiert die aktuelle Ansicht in eine Bilddatei
  - 3D-Szene exportieren: exportiert die aktuelle Ansicht als 3D-Szene (`.obj`-Datei), was eine Nachbearbeitung in Anwendungen wie Blender ermöglicht. Das Gelände und die Vektor-Objekte werden als 3D-Elemente exportiert. Sie haben die folgenden Einstellungsmöglichkeiten für den Export (die Einstellungen hier überschreiben die in den *Layereigenschaften* und in der *3D-Konfiguration*:
    * *Szenenname* und Ziel- *Ordner*

    - *Geländeauflösung*
    - *Geländetexturauflösung*
    - *Modellmaßstab*
    - *Kantenglättung*
    - *Normale exportieren*
    - *Texturen exportieren*
- Kartenthema der Ansicht setzen: ermöglicht die Auswahl der Layer, die in der Kartenansicht angezeigt werden sollen, aus einem vordefinierten *Kartenthema*
- Das **|Kamera|** Kamera Menü hilft Ihnen dabei, die Beziehung zwischen der 2D- und der 3D-Ansicht zu steuern:
    - Synchronisieren der Ansichten: *2D Kartenansicht folgt 3D Kamera* und/oder *3D Kamera folgt 2D Kartenansicht*
    - *Sichtbaren Kamerabereich in 2D-Kartenansicht anzeigen*
    - *3D-Szenenausmaß auf 2D-Kartenansicht setzen*: Ermöglicht es, die 3D-Szene zu beschneiden und nur das Gelände und die Objekte anzuzeigen, die einen auf der 2D-Kartenfläche gezeichneten Bereich schneiden. Weitere Optionen sind auf dem Reiter *Allgemein* in den 3D-Konfigurationen verfügbar.
    - 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.
- Effekte ermöglicht es Ihnen, visuelle Effekte zur 3D-Darstellung hinzuzufügen: *Schatten anzeigen*, *Eye-Dome-Lighting* oder *Umgebungsverdeckung anzeigen*.
- The Configure… button opens the dialog to configure the 3D map view *settings*.
- 3D Kartenansicht andocken: wechselt vom angedockten Fenster zum Fenster in der obersten Ebene

7.2.1 3D-Konfigurationen

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.

Im Fenster 3D-Konfiguration gibt es verschiedene Optionen zur Feinabstimmung der 3D-Szene, welche im Folgenden erläutert werden.

Allgemein

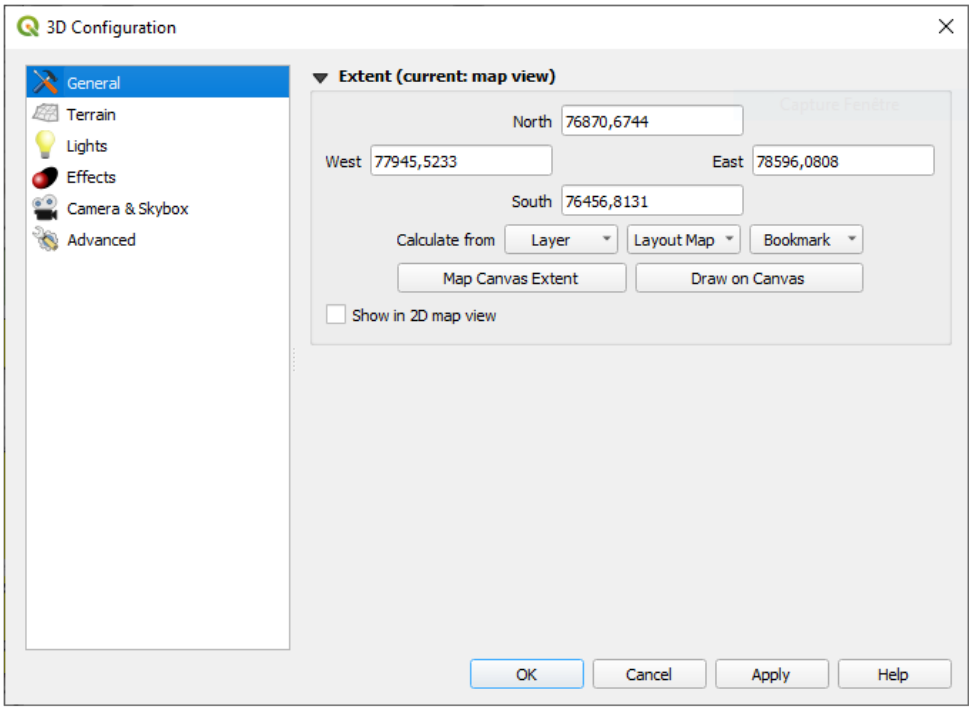


Abb. 7.22: Der Reiter „Allgemein“ des 3D-Konfiguration-Dialogs

Auf dem Reiter *Allgemein* können Sie:

- Den Ausschnitt der 3D-Szene auf eine 2D-Kartenausdehnung begrenzen, indem Sie die *Ausdehnungsauswahl* verwenden: Gelände und Features werden an der angegebenen Ausdehnung beschnitten und nur die Teile, die innerhalb der Ausdehnung liegen, werden in die 3D-Szene geladen. Der Höhenbereich ggf. vorhandener Gelände-, Netz- und Punktwolken-Layer wird ebenfalls berücksichtigt, sodass die Kamera nicht unter dem Inhalt der 3D-Szene positioniert wird, insbesondere wenn die vertikale Skalierungseinstellung des Terrains verwendet wird, um die Höhenunterschiede zu verstärken.

  Die eingestellte Ausdehnung wird auch als Referenzausdehnung verwendet, wenn Sie die Schaltfläche Volle Ausdehnung in der 3D-Kartenansicht drücken.

- Das Kontrollkästchen *In 2D-Kartenansicht anzeigen* aktivieren, um in der Kartenansicht ein Gummiband anzuzeigen, das der aktuellen Ausdehnung der 3D-Szene entspricht.

Gelände

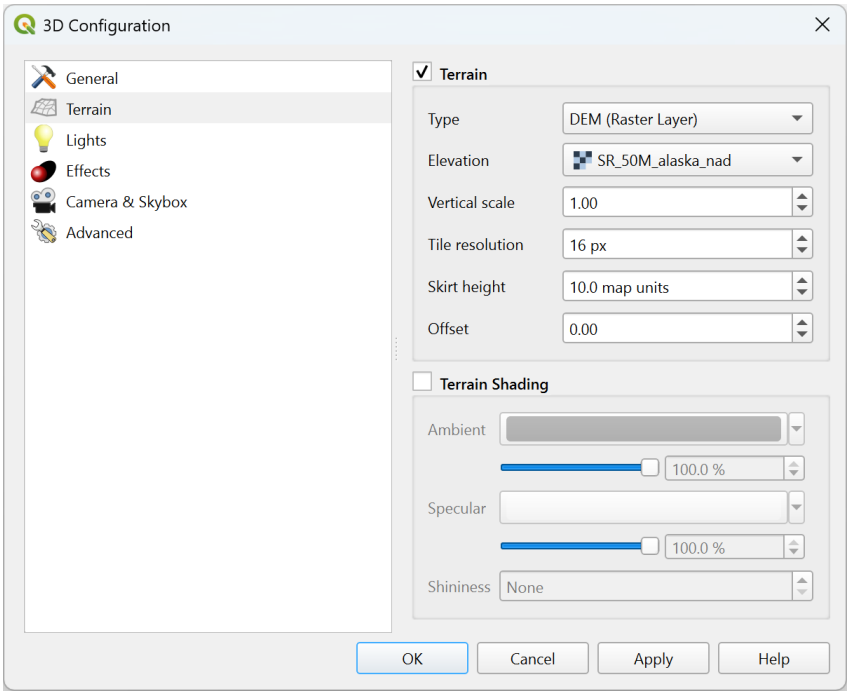


Abb. 7.23: Der Reiter „Gelände“ des 3D-Konfiguration-Dialogs

- *Gelände*: Das Gelände in einer 3D-Ansicht wird durch eine Hierarchie von Geländekacheln dargestellt . Wenn sich die Kamera näher an das Gelände heran bewegt, werden vorhandene Kacheln, die nicht genügend Details aufweisen, durch kleinere Kacheln mit mehr Details ersetzt. Jede Kachel hat eine Netzgeometrie, die aus dem Höhenraster-Layer abgeleitet ist und eine Textur aus 2D-Karten-Layern.
  - Für das Gelände kann ein *Typ* angegeben werden:
    - *Flaches Gelände*
    - *DEM (Rasterlayer)* (muss geladen sein)
    - *Online*-Dienst: lädt von Mapzen-Tools erstellte Höhenkacheln – weitere Einzelheiten unter https://registry.opendata.aws/terrain-tiles/
    - *Netz*-Layer Datensatz (muss geladen sein)
    - *Quantifiziertes Netz*: Ein Netz-Layer aus dem Projekt
  - *Höhe*: Raster- oder Netz-Layer, der für die Erstellung des Geländes verwendet werden soll. Der Rasterlayer muss ein Band enthalten, das die Höhe darstellt. Bei einem Netz-Layer werden die Z-Werte der Knotenpunkte verwendet.
  - *Vertikaler Maßstab*: Skalierungsfaktor für die vertikale Achse. Wenn Sie den Maßstab erhöhen, wird die Höhe der Landformen überhöht dargestellt.
  - *Kachelauflösung*: Stellt ein, wie viele Samples aus dem Terrain Raster Layer für jede Kachel verwendet werden sollen. Ein Wert von 16px bedeutet, dass die Geometrie jeder Kachel aus 16x16 Erhebungssamples besteht. Höhere Werte erzeugen detailliertere Geländekacheln auf Kosten einer höheren Rendering-Komplexität.
  - *Randhöhe*: Manchmal bilden sich kleine Risse zwischen den Geländekacheln. Wenn Sie diesen Wert erhöhen, werden vertikale Wände um die Geländekacheln hinzugefügt, um die Risse zu verbergen.
  - *Versatz*: Verschiebt das Gelände nach oben oder unten, z. B. um seine Höhe in Bezug auf das Bodenniveau anderer Objekte in der Szene anzupassen.

Dies kann nützlich sein, wenn es eine Diskrepanz zwischen der Höhe des Geländes und der Höhe der Layer in Ihrer Szene gibt (z. B. bei Punktwolken, die nur eine relative vertikale Höhe verwenden). In diesem Fall kann die manuelle Anpassung der Geländehöhe an die Höhe der Objekte in Ihrer 3D-Szene die Navigation verbessern.

- Wenn ein Netz-Layer als Gelände-Typ verwendet wird, können Sie die *Dreieckseinstellungen* (Drahtgitter anzeigen, Dreiecke glätten, Detailtiefe) und die *Darstellungsfarbeinstellungen* (als Einzelfarbe oder *Farbverlauf-basiert*) konfigurieren. Weitere Einzelheiten finden Sie im Abschnitt *Netz-Layer 3D Eigenschaften*.
- *Geländeschummerung*: Hier können Sie festlegen, wie das Gelände dargestellt werden soll:
  - Schummerung deaktiviert - die Farbe des Geländes wird nur von der Kartentextur bestimmt
  - Schattierung aktiviert - die Farbe des Geländes wird mit Hilfe des Phong-Schattierungsmodells bestimmt, wobei die *Stimmung*, die Farbe der *Spiegelung* sowie der *Glanz* eingestellt werden können.

Leuchten

Wenn Sie auf dem the *Leuchten* Reiter das Plus drücken, können Sie folgendes hinzufügen:

- Bis zu acht *Punktlichter*: strahlt Licht in alle Richtungen aus, wie eine Lichtkugel. Objekte, die sich näher am Licht befinden, werden heller, weiter entfernte Objekte werden dunkler. Ein Punktlicht hat eine bestimmte Position (*X*, *Y* und *Z*), eine *Farbe*, eine *Intensität* und eine *Dämpfung*.
- Bis zu viermal *Gerichtetes Licht*: ahmt die Beleuchtung nach, die man von einer riesigen Lichtquelle erhalten würde, welche sehr weit von den Objekten entfernt und immer zentriert ist (wie z. B. die Sonne). Sie sendet parallele Lichtstrahlen in eine Richtung aus, wobei das Licht bis ins Unendliche reicht. Ein gerichtetes Licht kann mit einem *Azimut* gedreht werden, hat einen *Höhenwinkel*, eine *Farbe* und eine *Intensität*.

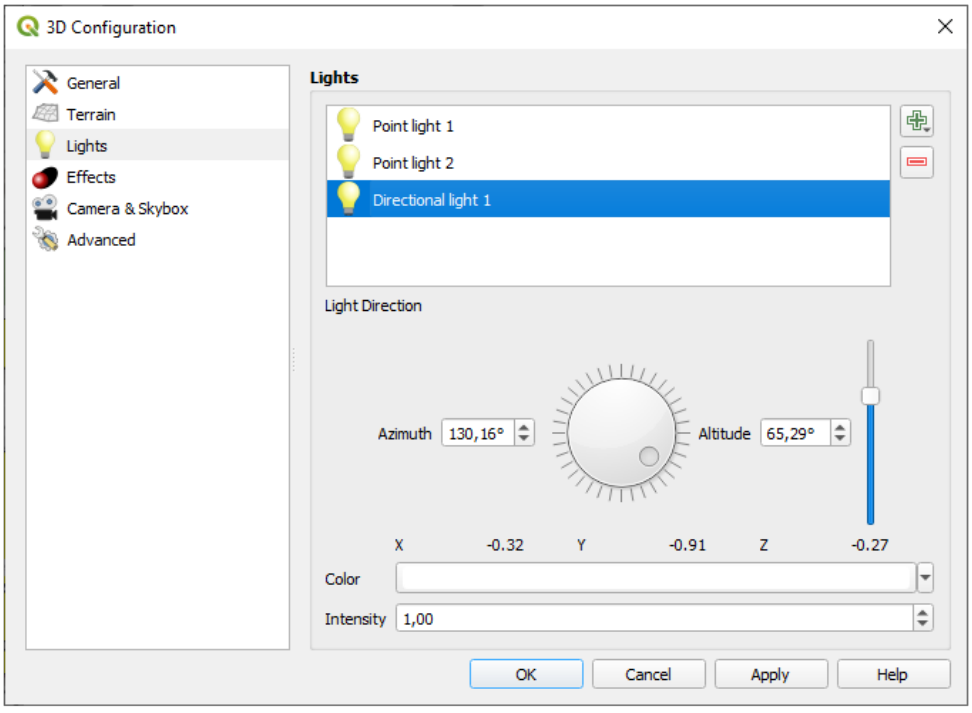


Abb. 7.24: Der Reiter „Leuchten“ des 3D-Konfiguration-Dialogs

Effekte

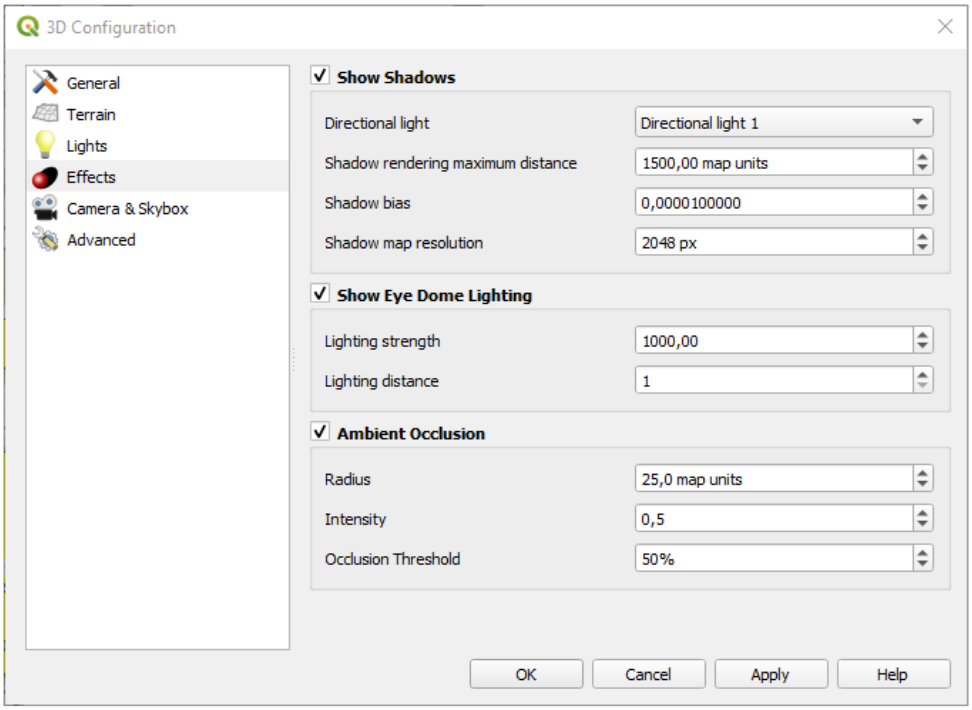


Abb. 7.25: Der Reiter „Effekte“ des 3D-Konfiguration-Dialogs

- Aktivieren Sie *Schatten anzeigen*, um Schatten in Ihrer Szene anzuzeigen, wobei Sie die folgenden Optionen haben:
  - *Gerichtetes Licht auswählen*
  - *Maximaler Abstand für Schattendarstellung*: hiermit können Sie die Darstellung von Schatten zu weit entfernter Objekte vermeiden, insbesondere wenn die Kamera entlang des Horizonts nach oben schaut.
  - *Schattenverzerrung*: zur Vermeidung von Selbstverschattungs-Effekten, die einige Bereiche aufgrund von Unterschieden in der Kartengröße dunkler machen könnten als andere. Je niedriger, desto besser.
  - *Schattenkartenauflösung*: um Schatten schärfer aussehen zu lassen. Es kann zu Leistungseinbußen führen, wenn der Auflösungsparameter zu hoch ist.
- ☐ *Eye-Dome-Lighting* (EDL) anzeigen: ein Post-Processing-Effekt, der die Tiefenwahrnehmung verbessert. Die Tiefe jedes Pixels (Abstand von der Kamera) wird mit der Tiefe der benachbarten Pixel verglichen und entsprechend dieser Tiefendifferenz hervorgehoben, wodurch die Kanten betont werden. Wirkt sich auf die gesamte Szene aus und kann mit *Umgebungsverdeckung* kombiniert werden. Die folgenden Parameter können eingestellt werden:
  - *Beleuchtungsstärke*: erhöht den Kontrast und ermöglicht eine bessere Tiefenwahrnehmung
  - *Beleuchtungsabstand*: stellt den Abstand der verwendeten Pixel vom mittleren Pixel dar und bewirkt, dass die Kanten dicker werden.
- ☐ *Umgebungsverdeckung* (engl.: „screen-space ambient occlusion“ kurz: SSAO): ein Nachbearbeitungseffekt, der die Tiefenwahrnehmung verbessert, indem er eine dunklere Schattierung auf Bereiche anwendet, die dem Umgebungslicht weniger ausgesetzt sind. Wirkt sich auf die gesamte Szene aus und kann mit *Eye-Dome-Lighting* kombiniert werden. Die folgenden Parameter können eingestellt werden:
  - *Radius*: wie weit die Umgebungsverdeckung berechnet werden soll
  - *Intensität*: wie stark der Effekt sein soll (höhere Werte machen Objekte dunkler)

– *Verdeckungsschwelle*: wie viele benachbarte Punkte verdeckt sein müssen, damit der Effekt erscheint (niedrigere Werte als 50% machen die Darstellung dunkler, bieten aber möglicherweise einen größeren Bereich der Verdeckung)

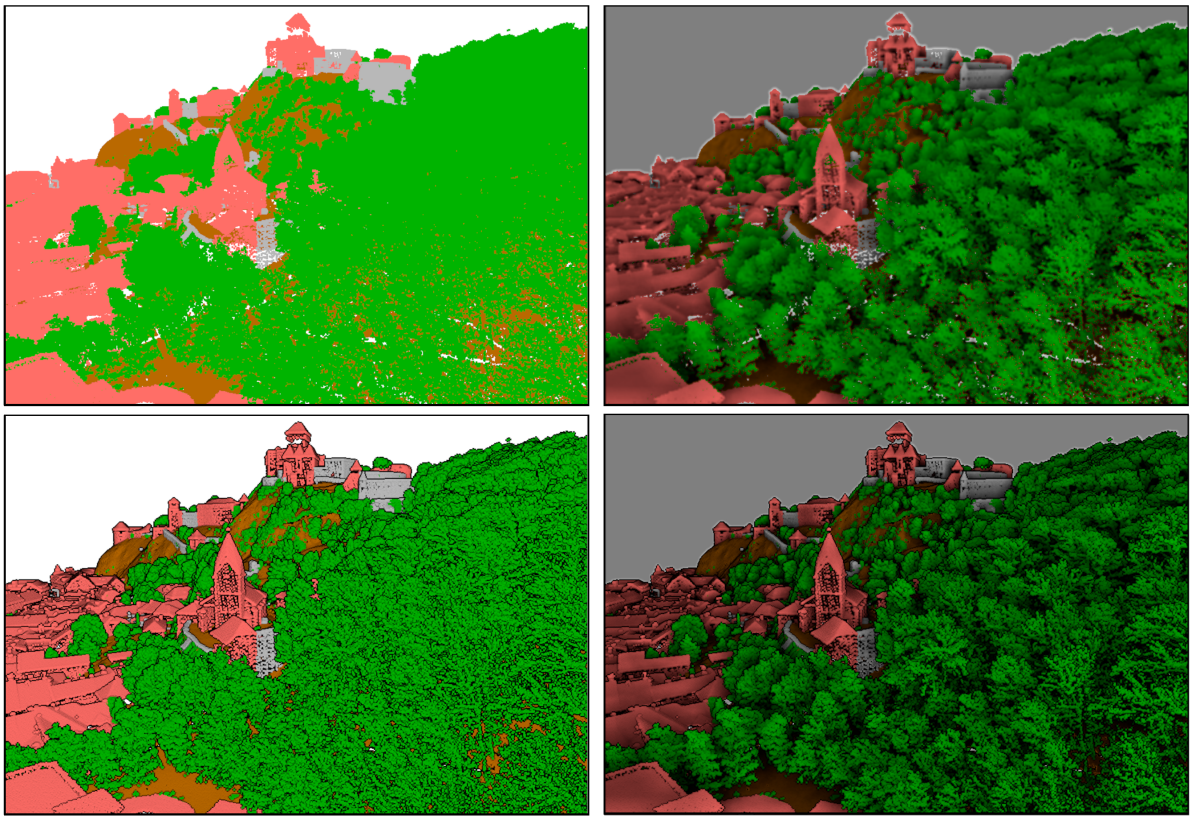


Abb. 7.26: Darstellung von Punktwolken-Layern in 3D-Karten mit Eye Dome Lighting (EDL) und/oder Umgebungsverdeckung
Von links-oben nach rechts-unten: Kein Effekt – Nur Umgebungsverdeckung – Nur EDL – Umgebungsverdeckung und EDL

Kamera & Skybox

Auf diesem Reiter können Sie verschiedene Parameter wie Kamera, 3D-Achse, Navigationssynchronisation und Skybox einstellen.

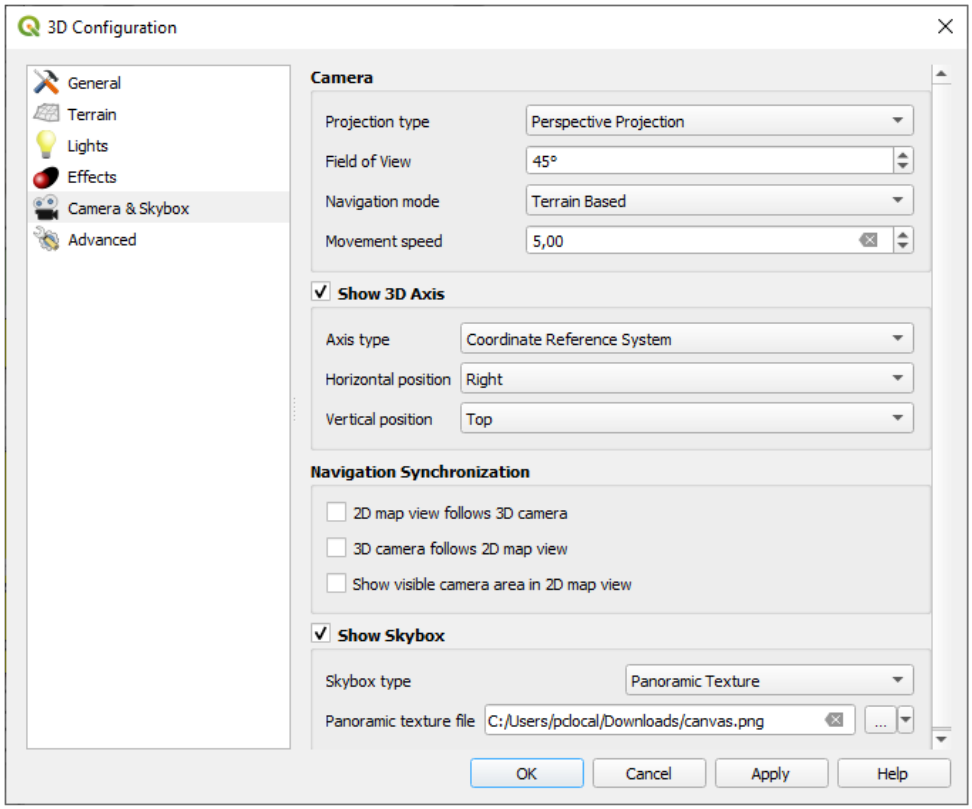


Abb. 7.27: Der Reiter „Kamera“ des 3D-Konfiguration-Dialogs

- Mit den Einstellungen im Abschnitt *Kamera* können einige *Standardkameraeinstellungen* überschrieben werden, die im Dialog *Einstellungen* ► *Optionen* ► *3D* vorgenommen werden können.
- Aktivieren Sie *3D-Achse anzeigen*, um das 3D-Achsenwerkzeug zu aktivieren. In diesem Bereich können Sie den Achsentyp und seine Position festlegen:
  - Bei Auswahl des Achsentyps *Koordinatenbezugssystem* wird eine orthogonale Achse dargestellt.
  - Mit dem Typ *Würfel* wird ein 3D-Würfel dargestellt. Die Würfelseiten können verwendet werden, um die Kameraperspektive zu ändern: Klicken Sie z. B. auf die *Nord*-Seite, um die Kamera so einzustellen, dass sie von Norden sieht.

Tipp

Klicken Sie mit der rechten Maustaste auf die 3D-Achse, die in der Kartenansicht erscheint, wenn Sie sie aktiviert haben, um deren Position und Typ sowie die Kameraansicht schnell festzulegen.

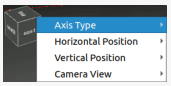


Abb. 7.28: Das 3D-Achse Kontextmenü

- Im Bereich *Navigationssynchronisation* können Optionen zur Synchronisation der 2D-Ansicht mit der 3D-Kameraposition oder der 3D-Kameraposition mit der 2D-Ansicht oder der bidirektionalen Synchronisation hinzugefügt werden. Die letzte Option zeigt den von der 3D-Kamera sichtbaren Bereich über der 2D-Kartenansicht an.
- Aktivieren Sie *Skybox anzeigen*, um die Darstellung der Skybox in der Szene zu aktivieren. Der Skybox-Typ kann sein:
  - *Panoramatextur* mit einer einzigen Datei, die Sicht auf 360° bietet

  - *Einzeloberflächen* mit einer Texturdatei für jede der sechs Seiten eines Kastens, der die Szene enthält

Texturbilddateien der Skybox können Dateien auf der Festplatte, Remote-URLs oder in das Projekt eingebettet sein (*weitere Einzelheiten*).

Erweitert

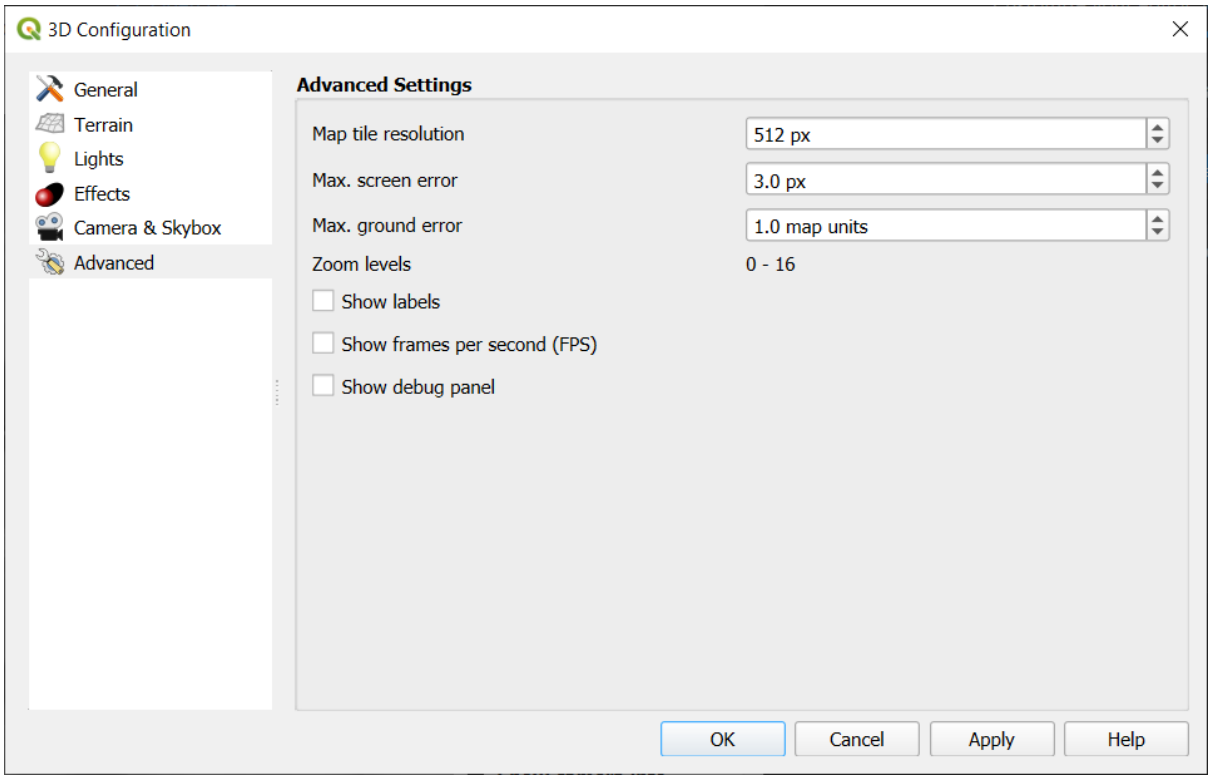


Abb. 7.29: Der Reiter „Erweitert“ des 3D-Konfiguration-Dialogs

- *Kartenkachelauflösung*: Breite und Höhe der 2D-Kartenbilder, die als Texturen für die Geländekacheln verwendet werden. 256px bedeutet, dass jede Kachel in ein Bild von 256x256 Pixeln gerendert wird. Höhere Zahlen erzeugen detailliertere Geländekacheln auf Kosten einer höheren Rendering-Komplexität.
- *Max. Bildschirmfehler*: Bestimmt den Schwellenwert für den Austausch von Geländekacheln gegen detailliertere Kacheln (und umgekehrt) - d.h. wie schnell die 3D-Ansicht höherwertige Kacheln verwendet. Niedrigere Zahlen bedeuten mehr Details in der Szene auf Kosten einer erhöhten Rendering-Komplexität.
- *Max. Grundfehler*: Die Auflösung der Geländekacheln, bei der die Aufteilung der Kacheln in detailliertere Kacheln aufhört (da die Aufteilung ohnehin keine zusätzlichen Details liefern würde). Dieser Wert begrenzt die Tiefe der Kachelhierarchie: niedrigere Werte machen die Hierarchie tiefer, was die Komplexität des Renderings erhöht.
- *Zoomstufen*: Zeigt die Anzahl der Zoomstufen an (abhängig von der Kartenkachelauflösung und dem maximalen Grundfehler).
- ☐ *Beschriftungen anzeigen*: Schaltet die Kartenbeschriftungen ein/aus
- ☐ *Einzelbilder pro Sekunde anzeigen (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:
  - ☐ *Kartenkachelinfo anzeigen*: Umrandung und Kachelnummern für die Geländekacheln anzeigen (hilfreich bei der Fehlersuche in Bezug auf das Gelände)

  - ☐ *Boundingboxen anzeigen*: 3D-Begrenzungsrahmen der Geländekacheln anzeigen (hilfreich bei der Fehlersuche in Bezug auf das Gelände)
  - ☐ *Sichtzentrum der Kamera anzeigen*
  - ☐ *Drehzentrum der Kamera anzeigen*
  - ☐ *Lichtquellen anzeigen*: zeigt eine Kugel an den Ursprüngen der Lichtquellen, was eine einfachere Neupositionierung und Platzierung der Lichtquellen relativ zum Szeneninhalt ermöglicht
  - ☐ *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
  - ☐ *Schattenkarte debuggen*: rendert die Szene als rot-schwarzes Bild aus der Sicht des für Schatten verwendeten Lichts (zur Fehlersuche). Das Widget wird mit einer proportionalen *Größe* zur 3D-Kartenansicht gesetzt und in einer *Ecke* angedockt.
  - ☐ *Kartentiefe debuggen*: stellt die Kartentiefe der Szene als Bild dar, wobei nähere Pixel dunkler sind (zur Fehlersuche). Das Widget wird mit einer proportionalen *Größe* zur 3D-Kartenansicht gesetzt und in einer *Ecke* angedockt.
  - *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

> **Bemerkung**
>
> 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 Navigationsmöglichkeiten

So erkunden Sie die Kartenansicht in 3D:

- Neigen des Geländes (Drehen um eine horizontale Achse, die durch die Mitte des Fensters verläuft)
  - Drücken Sie die Nach oben kippen und Nach unten kippen Tasten
  - Drücken Sie `Umschalt` und verwenden Sie die Auf-/Ab-Tasten
  - Ziehen Sie die Maus bei gedrückter mittlerer Maustaste vorwärts/rückwärts
  - Drücken Sie `Umschalt` und ziehen Sie die Maus mit gedrückter linker Maustaste vorwärts/rückwärts
- Drehen Sie das Terrain (um eine vertikale Achse, die durch die Mitte des Fensters verläuft)
  - Drehen Sie den Kompass des Navigations-Widgets in die Blickrichtung
  - Drücken Sie `Umschalt` und verwenden Sie die Links/Rechts-Tasten
  - Ziehen Sie die Maus mit gedrückter mittlerer Maustaste nach rechts/links
  - Drücken Sie `Umschalt` und ziehen Sie die Maus mit gedrückter linker Maustaste nach rechts/links
- Ändern Sie die Kameraposition (und das Zentrum der Ansicht), indem Sie sie in einer horizontalen Ebene bewegen

  - Ziehen Sie die Maus mit gedrückter linker Maustaste und aktivierter Kamerasteuerung-Taste
  - Drücken Sie die Richtungspfeile des Navigations-Widgets
  - Verwenden Sie die Tasten Auf/Ab/Links/Rechts, um die Kamera vorwärts, rückwärts, rechts bzw. links zu bewegen
- Ändern Sie die Kamerahöhe: Drücken Sie die Tasten `Bild auf`/`Bild ab`
- Ändern Sie die Kameraausrichtung (die Kamera bleibt an ihrer Position, aber der Mittelpunkt der Ansicht wird verschoben)
  - Drücken Sie `Strg` und verwenden Sie die Pfeiltasten, um die Kamera nach oben, unten, links und rechts zu bewegen
  - Drücken Sie `Strg` und ziehen Sie die Maus mit gedrückter linker Maustaste
- Hinein- und Herauszoomen
  - Drücken Sie die entsprechenden Werkzeuge Hineinzoomen und Herauszoomen des Navigations-Widgets
  - Scrollen Sie mit dem Mausrad (halten Sie `Strg` gedrückt, um feinere Zooms zu erhalten)
  - Ziehen Sie die Maus mit gedrückter rechter Maustaste, um zu vergrößern (nach unten ziehen) oder zu verkleinern (nach oben ziehen)

Um die Kameraansicht zurückzusetzen, klicken Sie auf die Schaltfläche Volle Ausdehnung oben auf dem Bedienfeld der 3D-Kartenansicht.

7.2.3 Eine Animation erstellen

Eine Animation basiert auf einer Reihe von sogenanten „Schlüsselbildern“. Dies sind Kamerapositionen zu bestimmten Zeitpunkten, die vom Nutzer festgelegt werden. Die dazwischen liegenden Bilder werden von QGIS automatisch durch Interpolation erzeugt. So erstellen Sie eine Animation:

1. Aktivieren Sie das Werkzeug Animation damit das Animations-Widget angezeigt wird.
2. Klicken Sie auf die Schaltfläche Schlüsselbild hinzufügen und geben Sie eine *Schlüsselbildzeit* in Sekunden ein. Das Auswahlfeld *Schlüsselbild* zeigt nun die eingestellte Zeit an.
3. Bewegen Sie die Kamera mit den Navigationswerkzeugen an die Position, die mit der aktuellen Schlüsselbildzeit verknüpft werden soll.
4. Wiederholen Sie die vorherigen Schritte, um so viele Schlüsselbilder hinzuzufügen, wie Sie benötigen.
5. Klicken Sie auf die Start-Schaltfläche , um eine Vorschau der Animation zu erhalten. QGIS generiert Szenen, indem es die Kamerapositionen und -drehungen zu bestimmten Zeiten verwendet und sie zwischen diesen Schlüsselbildern interpoliert. Es stehen verschiedene *Interpolation*-Modi für Animationen zur Verfügung (z.B. linear, inQuad, outQuad, inCirc… – mehr Details unter https://doc.qt.io/archives/qt-5.15/qeasingcurve.html#EasingFunction-typedef).

   Die Animation kann auch durch Verschieben des Zeitreglers in der Vorschau angezeigt werden. Wenn Sie das Kontrollkästchen *Schleife* aktivieren, wird die Animation wiederholt ausgeführt, während ein Klick auf eine laufende Animation anhält.

Klicken Sie auf Animationsbilder exportieren, um eine Reihe von Bildern zu erzeugen, die die Szene darstellen. Neben dem Dateinamen *Vorlage* und dem *Ausgabeverzeichnis* können Sie die Anzahl der *Einzelbilder pro Sekunde*, die *Ausgabebreite* und die *Ausgabehöhe* einstellen.

7.2.4 3D-Vektor-Layer

Ein Vektorlayer mit Höhenwerten kann in der 3D-Kartenansicht angezeigt werden, indem Sie das Kontrollkästchen *3D-Ansicht aktivieren* im Abschnitt *3D-Ansicht* der Eigenschaften des Vektorlayers aktivieren. Es gibt eine Reihe von Optionen zur Steuerung der Darstellung des 3D-Vektorlayers.

7.3 Geländehöhenprofil

Das Bedienfeld *Geländehöhenprofil* ist ein Plot-Tool für die Seitenansicht, um Höhendaten entlang einer Linie zu visualisieren. Es unterstützt Vektor-, Raster-, Netz- und Punktwolken-Layer. Die Daten können in 2D oder 3D vorliegen.

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.

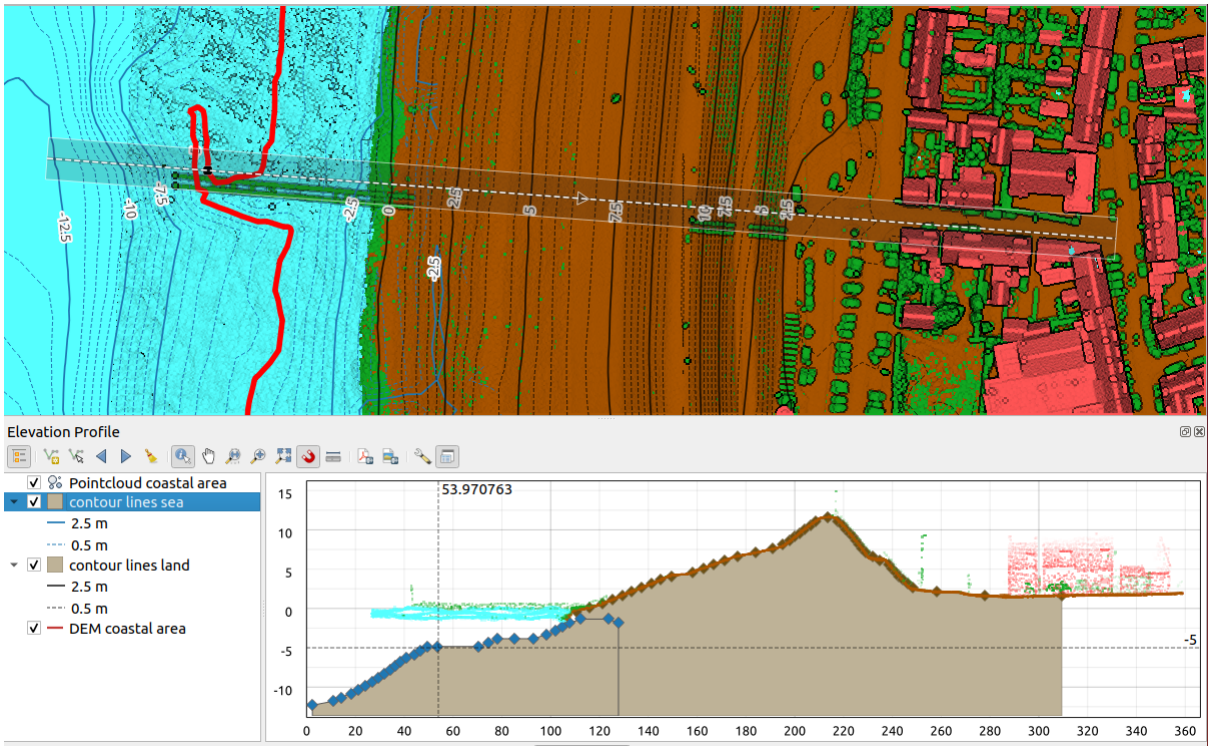


Abb. 7.30: Der Höhenprofil-Dialog, gedockt unter der Kartenansicht
Die Abbildung zeigt eine Küstenlinie mit einer Stadt hinter einem Deich. Eine größere Toleranz, die auf die Höhenprofillinie angewendet wird, liefert mehr Punktwolkenobjekte.

7.3.1 Die Benutzeroberfläche

Am oberen Rand des *Geländehöhenprofil* Fensters befindet sich eine Werkzeugleiste, die die folgenden Werkzeuge zur Verfügung stellt:

Tab. 7.5: Werkzeugleiste im Geländehöhenprofil-Fenster

Werkzeug	Tastenkürzel	Beschreibung
Layer hinzufügen	Halten Sie `Strg` beim Drag & Drop	Ermöglicht das Hinzufügen neuer Layer zu einem Plot.
Layerbaum anzeigen		Blendet eine Liste von Projekt-Layern ein oder aus, um die Darstellung in der Profilansicht zu konfigurieren.
Kurve aufnehmen		Zeichnet interaktiv eine Linie über die Kartenansicht, um die Profilkurve darzustellen.
Kurve aus Objekt erfassen		Wählt ein bestehendes Linien-Objekt auf der Kartenansicht aus und erzeugt eine Profilkurve entlang dieser Linie
Nach links schubsen	`Strg+Alt+,`	Ermöglicht es Ihnen, die Profillinie auf der Karte langsam nach links zu verschieben (z. B. um die optimale Profillinie anhand der Höhe zu finden)
Nach rechts schubsen	`Strg+Alt+.`	Ermöglicht es Ihnen, die Profillinie auf der Karte langsam nach rechts zu verschieben (z. B. um die optimale Profillinie auf der Grundlage der Höhe zu finden)
Leeren		Entfernt die Profillinie und alle im *Geländehöhenprofil* Fenster angezeigten Diagramme
Objekte abfragen		Identifiziert Objekte in der Kartenansicht entweder durch einen einzelnen Klick oder durch Klicken und Ziehen eines Rechtecks. Die Ergebnisse werden im üblichen Abfrage Ergebnisse-Dock angezeigt
Verschieben	`Leertaste`	Klicken und ziehen Sie, um die Kartenansicht zu verschieben. Kann auch mit der mittleren Maustaste durchgeführt werden
Zoom X-Achse		Vergrößern/Verkleinern entlang der horizontalen Achse unter Beibehaltung des vertikalen Verhältnisses
Zoom	`Strg+Leertaste`	Klicken oder ziehen Sie ein Rechteck über das Diagramm, um es zu vergrößern. Drücken Sie `Alt` und klicken Sie, um stattdessen herauszuzoomen
Volle Ausdehnung		Zoomt die Ansicht auf die Ausdehnung der Erfassungslinie
Einrasten aktivieren		Ermöglicht das Einrasten an der Kante oder dem Knoten der Objekte in der Profilansicht. Praktisch für den genauen Abruf von Koordinaten oder Entfernungsmessungen
Entfernungen Messen		Misst horizontale und vertikale Abstände
Als PDF exportieren		Exportiert Plots in PDF (als qualitativ hochwertige Vektorobjekte)
Als Bild exportieren		Exportiert Plots in verschiedene Bildformate
Ergebnisse exportieren		Exportiert Plots als 3D-Features, 2D-Profil und Distanz-/Höhen-Tabelle. (Die Ergebnisse können als DXF-Dateien, CSV-Dateien oder in einem Vektor-Format gespeichert werden). WEitere Detaiuls unter *Exportieren aus dem Höhenprofil*.
Optionen ►		Ermöglicht den Zugriff auf die Konfigurationseinstellungen für Geländehöhenprofile
► *Abstands-/Höhenmaßstäbe arretieren*		Stellt sicher, dass die horizontalen und vertikalen Skalen immer gleich bleiben (sodass z. B. eine 45°-Neigung in der Profilansicht als 45°-Neigung erscheint)

Fortsetzung auf der nächsten Seite

Tab. 7.5 – Fortsetzung der vorherigen Seite

Werkzeug	Tastenkürzel	Beschreibung
► *Abstandseinheiten*		Ermöglicht die Darstellung von Entfernungen in der Profilkarte in anderen Einheiten als in der Kartenansicht
► *Toleranz*		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.
► *Profil umbenennen …*		Ermöglicht die Umbenennung der Profilansicht
Höhenprofilansicht andocken		Umschalten zwischen angedocktem und schwebendem Status des Fensters

Unten links kann eine Kopie des *Layer* Bedienfelds durch Drücken der Schaltfläche *Layerbaum anzeigen* eingeblendet werden. Hierbei handelt es sich um ein unabhängiges Werkzeug mit einem eigenen Satz sichtbarer Layer, die in einer benutzerdefinierten Stapelreihenfolge angeordnet sind. Es erlaubt Ihnen, die Darstellung und das Verhalten der Layer innerhalb der Profilansicht separat zu steuern:

- Aktivieren Sie das Kontrollkästchen neben dem Namen des Layers, um festzulegen, ob er in der Profilansicht dargestellt werden soll.
- Ziehen Sie die Layer nach oben oder unten, um die Reihenfolge der Layer zu ändern.
- Darstellung von Layern in der Profilansicht: Doppelklicken Sie auf einen Layer oder klicken Sie mit der rechten Maustaste und wählen Sie *Eigenschaften…*, um den Reiter *Höheneinstellungen* des Layers zur Konfiguration zu öffnen. Eine Zusammenfassung der Höheneinstellungen wird als Tooltip angezeigt, wenn Sie den Mauszeiger über den Layer bewegen.

Auf der rechten Seite des Layer-Baums wird eine Vorschau der Profilansicht mit den jeweils aktivierten Layer angezeigt. Sie basiert auf einem skalierten Raster, in dem die horizontale Achse die Länge der Profillinie und die vertikale Achse die Z-Höhe der ausgewählten Objekte anzeigt. Die Vorschau ermöglicht auch eine Reihe von Interaktionen wie Zoomen, Schwenken, Messen, Identifizieren von Merkmalen, … mit den Werkzeugen am oberen Rand.

7.3.2 Erstellen eines Geländehöhenprofils

Um eine Profilansicht zu erstellen, gehen Sie wie folgt vor:

1. Gehen Sie in zum Menüpunkt *Ansicht* ► *Geländehöhenprofil*, um das entsprechende Fenster zu öffnen.
2. Erstellen Sie die Profillinie, entlang derer das Gelände und die Elemente dargestellt werden sollen. Wählen Sie dazu ein Zeichenwerkzeug:
   - Kurve aufnehmen: klicken Sie mit der linken Maustaste auf die Hauptkartenansicht, um Eckpunkte hinzuzufügen und klicken Sie mit der rechten Maustaste, um die Linie abzuschließen, die als Profillinie verwendet werden soll
   - Kurve aus Objekt erfassen: Klicken Sie auf ein Linien-Objekt in der Hauptkartenansicht, um dieses als Profillinie auszuwählen. Wenn an dem angeklickten Punkt mehrere Objekte vorhanden sind, wird ein Popup-Menü angezeigt, in dem Sie zwischen ihnen wählen können.

   Alle Funktionen der Linien-Digitalisierung wie die *Fang-Optionen*, *Verfolgung*, die einfache *Digitalisierung* sowie die *erweiterte Digitalisierung* stehen dabei zur Verfügung.

   In der Profilansicht können einige Objekte dargestellt werden.

3. Der nächste Schritt ist die Konfiguration der Höheneigenschaften der Layer, die Sie darstellen möchten.

    1. Drücken Sie auf die Schaltfläche Layerbaum anzeigen, um die Liste der verfügbaren Layer anzuzeigen. Standardmäßig werden nicht alle Layer des Projekts geladen und in der Profilansicht angezeigt; z.B. werden Rasterlayer, in deren Eigenschaften nicht angeben ist, dass sie über Höhendaten verfügen, ignoriert. Wenn Sie auf die Schaltfläche Layer hinzufügen klicken, wird eine gefilterte Liste verfügbarer Layer angezeigt, die derzeit nicht im Plot enthalten sind. Wenn ein Layer hinzugefügt wird, wird in den Layer-Eigenschaften automatisch ergänzt, dass er über Höhendaten verfügt.

        Tipp

        Wenn Sie die `Strg` Taste gedrückt halten, können Sie mehrere Layer auswählen, die gleichzeitig dem Geländehöhenprofil hinzugefügt werden können.

    2. Sie können die Sichtbarkeit der Layer ein- und ausschalten, indem Sie das entsprechende Kontrollkästen aktivieren/deaktivieren. Die Einstellungen hier sind unabhängig von den Einstellungen im *Layer* Bedienfeld in der Hauptkartenansicht.

    3. Doppelklicken Sie auf einen Layer-Namen oder klicken Sie mit der rechten Maustaste und wählen Sie *Eigenschaften*. Der Reiter *Höhe* des Layers öffnet sich. Hier können Sie einstellen, wie jedes Objekt oder Gelände in der Profilansicht dargestellt werden soll. Die verfügbaren Eigenschaften hängen vom Typ des Layers ab:

        - *Raster-Layer Höhen-Eigenschaften*
        - *Vektor-Layer Höhen-Eigenschaften*
        - *Punktwolken-Layer Höhen-Eigenschaften*
        - *Netz-Layer Höhen-Eigenschaften*

        Die Profilansicht beginnt mit der Darstellung von Gelände oder Objekten der aktiven Layer, die die angegebene Profilkurve kreuzen, sobald die Höheneigenschaften eingestellt sind.

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.

Bemerkung

Einschränkungen bei der Polygon-Extrusion

Die Geometrie-Extrusion kann in den Eigenschaften *Höhe* eines Layers festgelegt und in der Profilansicht gerendert werden. Wenn die Toleranz aktiviert ist, ist es jedoch nicht einfach, extrudierte Polygone zu rendern, daher wird die Polygon-Extrusion derzeit ignoriert.

7.3.3 Interaktion mit der Profilansicht

Wenn eine Höhenprofillinie erstellt wird, wird die Profilansicht auf diesen Bereich vergrößert. Auf der X-Achse sehen Sie die Länge des Profils und auf der Y-Achse den Höhenbereich zwischen der minimalen und der maximalen erfassten Höhe, beides in den aktuellen Karteneinheiten.

Wenn Sie den Mauszeiger in der Profilansicht bewegen, sehen Sie zwei sich kreuzende Punktlinien:

- die vertikale Linie zeigt die Höheninformation an
- die horizontale Linie zeigt die Entfernung vom Anfangspunkt der Profillinie an

Wenn Sie den Mauszeiger in der Profilansicht bewegen, bewegt sich auch ein schwarzer Punkt entlang der Höhenprofillinie in der Hauptkartenansicht. In der Mitte der Profillinie zeigt ein Pfeil die Richtung des Profils an.

Wie bei der Hauptkartenansicht bietet QGIS die Möglichkeit, in der Profilansicht zu navigieren:

- Verschieben wird verwendet, um die Ausdehnung des Höhenprofils in jede beliebige Richtung zu schwenken. Sie können aber auch die Leertaste gedrückt halten, während Sie die Maus bewegen, um die Ansicht zu verschieben.
- Zoom X-Achse wird verwendet, um entlang der horizontalen Achse zu zoomen, wobei das Verhältnis der vertikalen Achse (die Höhe) gleich bleibt. Klicken Sie mit der linken Maustaste, um die Darstellung entlang der X-Achse zu vergrößern, wobei der angeklickte Punkt in die Mitte der Achse gelegt wird, oder ziehen Sie ein Bereich auf, um die Darstellung entlang der X-Achse auf diesen Bereich zu vergrößern. Halten Sie `Alt` gedrückt, während Sie Zoom X-Achse verwenden, um entlang dieser Achse zu verkleinern.
- Zoom wird verwendet, um auf einen bestimmten Punkt zu zoomen (mit der linken Maustaste) oder auf einen bestimmten Bereich (Ziehen Sie ein Rechteck darüber auf). Halten Sie `Alt`, während Sie Zoom verwenden, um herauszuzoomen. In Kombination mit der Taste `Strg` können Sie feiner hinein- oder herauszoomen.
- Volle Ausdehnung ist die zu Beginn verwendete Standard-Zoomstufe und zeigt die volle Ausdehnung der Profillinie mit allen ermittelten Objekten. Verwenden Sie diese Option, um die Zoomstufe zurückzusetzen.

Es ist auch möglich, mit den in der Profilansicht angezeigten Elementen zu interagieren:

- Drücken Sie die Taste Einrasten aktivieren, um Punkte, Knoten oder Kanten der Objekte exakt zu erfassen, um genaue Messergebnisse oder Koordinatenabfragen zu erhalten.
- Objekte abfragen dient zur Identifizierung von Objekten der sichtbaren Layer. Sie können ein Rechteck über mehrere Objekte in der Profilansicht ziehen, um alle gleichzeitig abzufragen. Wenn für das Format verfügbar (z. B. Vektor, Punktwolke), werden diese Objekte in der Hauptkartenansicht hervorgehoben.
- Entfernungen messen: Klicken Sie auf zwei Punkte in der Profilansicht oder wählen Sie diese aus, um den horizontale *Abstand*, die Differenz der *Höhe* und die *Gesamtlänge* zwischen den Punkten in Karteneinheiten zu ermitteln.
- Nach links schubsen oder Nach rechts schubsen wird verwendet, um die Position der Höhenprofillinie in der Hauptkartenansicht rechtwinkelig zu ihrer Ausrichtung nach links oder rechts zu verschieben. Die Profilansicht wird neu gezeichnet und zeigt nun Objekte und Gelände, die Profillinie (inkl. Puffer) überlappen. Die Linie wird unter Verwendung des Wertes *Toleranz* im Menü Optionen seitlich verschoben.

> **Warnung**
>
> Wenn Sie eine Höhenprofilansicht oder das Projekt schließen, wird die Ansicht gelöscht.

Weitere Einzelheiten finden Sie in der Präsentation QGIS elevation profile/cross section tool – a deep dive!, von Nyall Dawson.

7.3.4 Exportieren aus dem Höhenprofil

Die obere Symbolleiste des Dialogfelds *Geländehöhenprofil* bietet eine Vielzahl von Werkzeugen zum Exportieren eines Höhenprofils, mit denen Sie die Diagramme in einer anderen Software wiederverwenden können:

- Als PDF exportieren: Exportiert Diagramme als PDF (als hochwertige Vektorobjekte). Die folgenden Optionen werden für die Erstellung der Ausgabedatei benötigt:
  - *Seitengröße*: Auswahl der Größe aus einer vordefinierten Liste oder Festlegen benutzerdefinierter Abmessungen und der Seitenausrichtung
  - *Diagrammbereich* zur Steuerung der Abmessungen des zu exportierenden Diagramms:
    - auf der X-Achse den *Minimalabstand* und *Maximalabstand* vom Startpunkt der Profilkurve
    - Auf der Y-Achse die Werte *Minimale Geländehöhe* und *Maximale Geländehöhe*
  - Die *Distanzachseneinstellungen*: helfen Ihnen bei der Konfiguration der horizontalen Skalierung des Rasters, über das das Höhenprofil gerendert wird. Sie können das *Beschriftungsintervall*, das *Hauptgitterlinienintervall* und das *Untergitterlinienintervall* anpassen.
  - Die *Geländehöhenachseneinstellung*, wie oben, für vertikale Teilung
- Als Bild exportieren: Exportiert Diagramme in ein Bilddateiformat. Spezifische *Exportoptionen* wie *Bildbreite* und *Bildhöhe* werden zusammen mit den unter *Als PDF exportieren* erläuterten *Diagrammbereiche*, *Distanzachseneinstellungen* und *Geländehöhenachseneinstellungen* abgefragt.
- Ergebnisse exportieren: Exportiert die Profilergebnisse als eine Reihe von Features eines Vektorlayers (`.DXF`, `.CSV`, `.SHP`, `.GPKG`, …). Je nach Ausgabeformat können mehrere Layer zurückgegeben werden, wenn sich die Geometrietypen der exportierten Features unterscheiden.
  - *3D-Objekte exportieren*: Exportiert Profilkurven als 3D- Objekte, wobei die Höhenwerte aus den Höhenschnitten übernommen und in den exportierten Geometrie-Z-Werten gespeichert werden.
  - *2D-Profil exportieren*: Exportiert Profile als 2D-Profil-Linien, wobei die Höhe in der exportierten Geometrie-Y-Dimension und die Entfernung in der X-Dimension gespeichert wird (wie im Höhenprofil-Widget dargestellt).
  - *Abstand-/Höhentabelle exportieren*: Exportiert Profile als Tabelle mit entnommenen Entfernungs- und Höhenwerten.

KAPITEL 8

Allgemeine Werkzeuge

8.1 Hilfe

Wann immer Sie Hilfe zu einem bestimmten Thema benötigen, können Sie die entsprechende Seite im aktuellen Benutzerhandbuch über die Schaltfläche *Hilfe* aufrufen, die in den meisten Dialogen verfügbar ist — bitte beachten Sie, dass Plugins von Drittanbietern auf spezielle Webseiten verweisen können.

8.2 Bedienfelder

Standardmäßig stellt QGIS viele Panels zur Verfügung, mit denen Sie arbeiten können. Einige dieser Panels werden im Folgenden beschrieben, während andere in verschiedenen Teilen des Dokuments zu finden sind. Eine vollständige Liste der von QGIS bereitgestellten Standardpanels finden Sie über die *View* ► *Panels* ► Menü und unter *Bedienfelder*.

8.2.1 Layer-Bedienfeld

Das *Layer* Bedienfeld (auch `Kartenlegende` genannt) listet alle Layer im Projekt auf und hilft Ihnen, deren Sichtbarkeit zu verwalten und die Karte zu gestalten. Sie können das Bedienfeld ein- und ausblenden, indem Sie `Strg+1` drücken.

QGIS bietet verschiedene Möglichkeiten, Layer zu einem Projekt hinzuzufügen:

- über die Schaltfläche *Hinzufügen* im Dialogfeld *Datenquellenverwaltung*
- in den Bedienfeldern *Browser* und *DB Manager*: Doppelklick, Drag-and-Drop von Dateien und Layern oder über das Kontextmenü
- Drag-and-Drop von Dateien aus einem Dateiexplorer des Betriebssystems

In all diesen Fällen können Sie einen oder mehrere Layer auf einmal öffnen. Neue Layer werden dem Bedienfeld *Layers* hinzugefügt:

1. wenn sie über das *Layer* Bedienfeld gezogen werden, genau an der Stelle, an der sie abgelegt werden
2. in den anderen Fällen an einer Stelle, wie es in die globale Einstellung *Verhalten beim Hinzufügen neuer Layer* festgelegt ist
3. und bei mehreren Layern werden sie so sortiert, dass ihre Stapelung logisch ist und die Merkmale so weit wie möglich sichtbar sind, und zwar nach folgender Logik (von oben nach unten):

- Punkt-Vektorlayer
- Linien-Vektorlayer
- Polygon-Vektorlayer
- Punktwolken-Layer
- Netz-Layer
- Rasterlayer

Am oberen Rand des *Layer* Bedienfelds befindet sich eine Werkzeugleiste, mit der Sie Folgendes tun können:

- Layergestaltungsfester öffnen (F7): schaltet das *Lagergestaltung* Bedienfeld an und aus.
- Neue Gruppe hinzufügen: siehe *Mit Layern und Gruppen arbeiten*
- Kartenthemen verwalten: steuert die Sichtbarkeit von Layern und arrangiert sie in verschiedenen *map themes*.
- Layer im Legendenbaum filtern:
  - *Legende nach Karteninhalt filtern*: es werden nur die Layer im Bedienfeld dargestellt, die sichtbar sind und deren Objekte im aktuellen Kartenausschnitt liegen. Andernfalls wird für den Layer ein allgemeines NULL-Symbol im Layerbaum gezeigt. So lässt sich leicht feststellen, welche Objekte aus welchen Layern bestimmte Bereiche abdecken.
  - *Private Layer anzeigen*: eine Möglichkeit, um nur *private Layer* im Bedienfeld *Layers* anzeigen zu lassen und mit ihm zu interagieren, ohne die Projekteinstellungen zu ändern.
  - *Nur defekte Layer anzeigen*: Es werden nur Layer mit fehlerhaften Datenquellen angezeigt.
- Legende nach Ausdruck filtern: einen Ausdruck anwenden, um Stile aus dem ausgewählten Layer-Baum zu entfernen, die keine Objekte haben, welche die Bedingung erfüllen. Dies kann verwendet werden, um Features hervorzuheben, die sich innerhalb eines bestimmten Bereichs/Features eines anderen Layers befinden. In der Dropdown-Liste können Sie den aktuell angewendeten Ausdruck bearbeiten und löschen.
- Alle ausklappen or Alle einklappen: alle Layer oder Gruppen werden aus- oder eingeklappt
- Layer/Gruppe entfernen entfernt die aktuell ausgewählten Elemente

Abb. 8.1: Die Werkzeugleiste im Layer-Bedienfeld

Bemerkung

Werkzeuge zur Verwaltung des Layer-Bedienfeld sind auch für Karten- und Legendenelemente in Drucklayouts verfügbar.

Kartenthemen konfigurieren

Die Manage Map Themes drop-down Schaltfläche bietet einen Zugang zu den Einstellungen zur Sichtbarkeit der Layer im *Layers* Panel:

- *Alle Layer anzeigen*
- *Alle Layer ausblenden*
- *Zeige gewählte Layer*
- *Ausblenden von gewählten Layern*
- *Gewählte Layer umschalten*: ändert die Sichtbarkeit des obersten, ausgewählten Layers im Bedienfeld und wendet diesen Zustand auf die anderen ausgewählten Layer an. Auch über die Tastenkombination `Space` erreichbar.
- *Gewählte Layer einzeln umschalten*: ändert den Status aller gewählten Layer
- *Abgewählte Layer verbergen*

Neben der einfachen Steuerung der Sichtbarkeit von Layern ermöglicht das Menü Kartenthemen verwalten die Konfiguration von **Kartenthemen** in der Legende und den Wechsel von einem Kartenthema zu einem anderen. Ein Kartenthema ist ein **Schnappschuss** der aktuellen Kartenlegende, der aufgezeichnet wird:

- die im *Layer* Bedienfeld als sichtbar eingestellten Layer
- **und** für jeden sichtbaren Layer:
  - einen Verweis auf den für den Layer angewandten *Style*
  - die sichtbaren Klassen des Stils, d. h. die im *Layer* Bedienfeld aktivierten Unterkategorien der Symbologie (falls vorhanden, also nicht bei ref:*Symbologien <vector_style_menu>* mit Einzelsymbol)
  - den eingeklappten/ausgeklappten Zustand des/der Layerbaum-Knoten und der Gruppe(n), in der/denen er sich befindet

So erstellen Sie ein Kartenthema:

1. Markieren Sie einen Layer, den Sie anzeigen möchten
2. Konfigurieren Sie die Layer-Eigenschaften (Symbologie, Diagramm, Beschriftungen…) wie gewohnt
3. Erweitern Sie das Menü *Stile* ► am unteren Rand und klicken Sie auf *Hinzufügen…*, um die Einstellungen als *einen neuen, in das Projekt eingebetteten Style zu speichern*

   > **Bemerkung**
   >
   > In einem Kartenthema werden nicht die tatsächlichen Details der Eigenschaften gespeichert, sondern nur ein Verweis auf den Namen des Stils. Wenn Sie also Änderungen an dem Stil vornehmen (z. B. die Darstellung der Symbologie ändern), wird das Kartenthema bei einem Aufruf mit der neuen Symbolgie dargestellt.

4. Wiederholen Sie die vorangegangenen Schritte nach Bedarf für die anderen Layer
5. Falls erforderlich, erweitern oder reduzieren Sie Gruppen oder sichtbare Layerbaum-Knoten im *Layer* Bedienfeld.
6. Klicken Sie auf die Schaltfläche Kartenthemen verwalten oben auf dem Bedienfeld, und dann auf *Thema hinzufügen…*
7. Geben Sie den Namen des Kartenthemas ein und klicken Sie auf *OK*.

Das neue Thema wird im unteren Teil des Dropdown-Menüs aufgeführt.

Sie können so viele Kartenthemen erstellen, wie Sie benötigen: Wann immer Sie die aktuelle Kombination in der Kartenlegende (sichtbare Layer, aktiver Stil, Knoten in der Kartenlegende) als neues Kartenthema speichern wollen, klicken Sie auf *Thema hinzufügen…* oder verwenden Sie *Thema ersetzen* ►, um ein Kartenthema zu aktualisieren. Sie können den Namen des aktiven Kartenthemas mit *Aktuelles Thema umbenennen…* ändern oder es mit der Schaltfläche *Aktuelles Thema entfernen* löschen.

Kartenthemen sind hilfreich, um schnell zwischen verschiedenen vorkonfigurierten Kombinationen zu wechseln: Wählen Sie ein Kartenthema in der Liste aus, um seine Kombination wiederherzustellen. Alle konfigurierten Themen sind auch im Drucklayout verfügbar, sodass Sie verschiedene Layouts auf der Grundlage bestimmter Themen und unabhängig von der aktuellen Darstellung auf der Kartenansicht erstellen können (siehe *Layer-Kartenelement*).

Übersicht über das Kontextmenü des Layer-Bedienfelds

Im Layer-Bedienfeld ist die Hauptkomponente der Bereich, in dem alle dem Projekt hinzugefügten Layer aufgelistet sind, optional in Gruppen geordnet. Ein Layer, neben dem das Kontrollkästchen aktiviert ist, wird in der Kartenansicht angezeigt, es sei denn, für den Layer ist eine *maßstabsäbhängige Sichtbarkeit* eingestellt. Ein Layer kann ausgewählt und in der Legende nach oben oder unten gezogen werden, um die Darstellungsreihenfolge zu ändern. Darstellungsreihenfolge bedeutet, dass Layer, die weiter oben in der Legende aufgeführt sind, über Layern gezeichnet werden, die weiter unten in der Legende aufgeführt sind. Auch kann ein Layer oder eine Gruppe von Layern über mehrere Instanzen von QGIS gezogen werden.

> **Bemerkung**
>
> Die Darstellungsreihenfolge kann mit dem Bedienfeld *Layerreihenfolge* überschrieben werden.

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*).

Tab. 8.1: Kontextmenü bei den verschiedenen Elementen im *Layer* Bedienfeld

Option	Gruppe	Vektorlaye	Rasterlaye	Netz-Layer	Punktwolk Layer	3D-Layer
Auf Layer/Gruppe zoomen	☑	☑	☑	☑	☑	☑
Auf Auswahl zoomen		☑				
In der Übersicht anzeigen		☑	☑	☑	☑	☑
Objektanzahl anzeigen		☑				
Beschriftung anzeigen		☑		☑		
Layer/Gruppe kopieren	☑	☑	☑	☑	☑	☑
Layer/Gruppe umbenennen	☑	☑	☑	☑	☑	☑
Auf eigene Auflösung zoomen (100%)			☑			
Strecken mit aktueller Ausdehnung			☑			
Create Raster Attribute Table			☑			
Open Raster Attribute Table			☑			
Load Raster Attribute Table from VAT.DBF			☑			
SQL-Layer aktualisieren…		☑				
Execute SQL…		☑				

Fortsetzung auf der nächsten Seite

Tab. 8.1 – Fortsetzung der vorherigen Seite

Option	Gruppe	Vektorlaye	Rasterlaye	Netz-Layer	Punktwolk Layer	3D-Layer
Virtuellen Layer. bearbeiten..		☑				
Gruppe hinzufügen	☑					
Layer dublizieren		☑	☑	☑	☑	☑
Layer/Gruppe löschen…	☑	☑	☑	☑	☑	☑
Aus Gruppe entfernen		☑	☑	☑	☑	
Nach oben bringen	☑	☑	☑	☑	☑	
Nach unten schieben	☑	☑	☑	☑	☑	
inkl aller Eltern aktiv schalten		☑	☑	☑	☑	
Gewählte Gruppe		☑	☑	☑	☑	
Attributtabelle öffnen		☑				
Bearbeitungsstatus umschalten		☑		☑		
Aktuelle Änderungen ►		☑		☑		
Filter…		☑	☑		☑	
Datenquelle ändern…		☑	☑	☑	☑	
Datenquelle reparieren…		☑	☑	☑	☑	
Auswahl ► (in edit mode)		☑				
► *Objekte dublizieren*		☑				
► *Objekte duplizieren und digitalisieren*		☑				
Maßstabsabhängige Sichtbarkeit setzen…		☑	☑	☑	☑	☑
Auf sichtbaren Maßstab zoomen		☑	☑	☑	☑	
Layer-KBS►		☑	☑	☑	☑	☑
► *Layer-KBS dem Projekt zuweisen*		☑	☑	☑	☑	☑
► *Auf … setzen* (kürzlich verwendete KBS)				☑	☑	☑
► *Layer-KBS setzen…*		☑	☑	☑	☑	☑
KBS für Gruppe setzen…	☑					
WMS-Daten für Gruppe setzen…	☑					
Gegenseitig ausschließende Gruppe	☑					
Inklusive aller Kinder anhacken (Strg-Klick)	☑					
Inklusive aller Kinder abhacken (Strg-Klick)	☑					
Dauerhaft machen		☑				
Exportieren ►	☑	☑	☑	☑	☑	☑
► *Speichern als…*			☑			
► *Objekte speichern als…*		☑				

Fortsetzung auf der nächsten Seite

Tab. 8.1 – Fortsetzung der vorherigen Seite

Option	Gruppe	Vektorlaye	Rasterlaye	Netz-Layer	Punktwolk Layer	3D-Layer
► *Ausgewählte Objekte speichern als…*		☑				
► *Als Layer-Definitionsdatei speichern…*	☑	☑	☑	☑	☑	☑
► *Als QGIS Layerstildatei speichern…*		☑	☑	☑	☑	☑
Stile ►		☑	☑	☑	☑	☑
► *Stil kopieren*		☑	☑	☑	☑	☑
► *Stil einfügen*	☑	☑	☑	☑	☑	
► *Hinzufügen…*		☑	☑	☑	☑	☑
► *Aktuellen umbenennen…*		☑	☑	☑	☑	☑
► *Symbol bearbeiten…*		☑				
► *Symbol kopieren*		☑				
► *Symbol einfügen*		☑				
Layernotizen hinzufügen…		☑	☑	☑	☑	☑
Layernotizen bearbeiten…		☑	☑	☑	☑	
Layernotizen entfernen		☑	☑	☑	☑	
Eigenschaften…		☑	☑	☑	☑	☑

Für GRASS Vektorlayer ist Bearbeitung umschalten nicht verfügbar. Siehe Abschnitt *Digitalisieren und Editieren eines GRASS Vektorlayers* für Informationen zur Bearbeitung von GRASS-Vektor-Layern.

Mit Layern und Gruppen arbeiten

Die Layer im Legendenfenster können in Gruppen organisiert werden. Es gibt dafür verschiedene Möglichkeiten:

1. Drücken Sie das Symbol, um eine neue Gruppe hinzuzufügen. Geben Sie einen Namen für die Gruppe ein und drücken Sie `Enter`. Klicken Sie nun auf einen vorhandenen Layer und ziehen Sie ihn in die Gruppe.
2. Wählen Sie mehr als einen Layer aus und drücken Sie dann auf das Symbol. Die ausgewählten Layer werden automatisch in die neue Gruppe eingefügt.
3. Wählen Sie einige Layer aus, klicken Sie mit der rechten Maustaste in das Legendenfenster und wählen Sie *Gewählte gruppieren*. Die ausgewählten Layer werden automatisch in eine neue Gruppe eingefügt.

Um einen Layer aus einer Gruppe zu entfernen, ziehen Sie ihn heraus, oder klicken Sie mit der rechten Maustaste darauf und wählen Sie *Aus Gruppe entfernen*: Der Layer wird aus der Gruppe entfernt und über der Gruppe platziert. Gruppen können auch innerhalb anderer Gruppen verschachtelt werden. Wenn ein Layer in einer verschachtelten Gruppe platziert ist, wird er mit *Aus Gruppe entfernen* aus allen verschachtelten Gruppen entfernt.

Um eine Gruppe oder einen Layer an den Anfang des Bedienfelds zu verschieben, ziehen Sie ihn entweder an den Anfang oder wählen Sie *Nach oben bringen*. Wenn Sie diese Option auf einen Layer anwenden, der in einer Gruppe verschachtelt ist, wird der Layer in seiner aktuellen Gruppe an den Anfang verschoben. Die Option *Nach unten verschieben* folgt der gleichen Logik, um Layer und Gruppen nach unten zu verschieben.

Das Kontrollkästchen für eine Gruppe blendet die markierten Layer der Gruppe mit einem Klick ein oder aus. Wenn Sie `Strg` gedrückt halten, schaltet das Kontrollkästchen auch alle Layer in der Gruppe und ihren Untergruppen ein oder aus.

`Strg`-Klick auf einen markierten / nicht markierten Layer hebt die Markierung / Markierung des Layers und aller seiner Eltern auf.

Wenn Sie die Option **Gegenseitig ausschließende Gruppe** aktivieren, können Sie festlegen, dass in einer Gruppe nur ein Layer gleichzeitig sichtbar ist. Immer wenn ein Layer innerhalb der Gruppe sichtbar gemacht wird, werden die anderen nicht sichtbar geschaltet.

Es ist möglich, mehrere Layer oder Gruppen gleichzeitig auszuwählen, indem Sie die Taste `Strg` gedrückt halten, während Sie weitere Layer anklicken. Sie können dann alle ausgewählten Layer gleichzeitig in eine neue Gruppe verschieben.

Sie können auch mehrere Layer oder Gruppen auf einmal löschen, indem Sie mehrere Elemente mit der Taste `Strg` auswählen und dann `Strg+D` drücken: alle ausgewählten Layer oder Gruppen werden aus der Liste der Layer entfernt.

Weitere Informationen zu Layern und Gruppen über das Indikatorsymbol

In einigen Fällen ändern sich die Formatierungen oder Symbole neben dem Layer oder der Gruppe im Bedienfeld *Layers*, um weitere Informationen über den Layer/die Gruppe zu liefern. Diese Elemente sind:

- um anzuzeigen, dass sich der Layer im Bearbeitungsmodus befindet und Sie die Daten ändern können
- , um anzuzeigen, dass der bearbeitete Layer ungespeicherte Änderungen enthält
- , um den auf den Layer angewendeten *Filter* anzuzeigen. Bewegen Sie den Mauszeiger über das Symbol, um den Filterausdruck zu sehen, und doppelklicken Sie darauf, um die Abfrage zu aktualisieren
- zur Kennzeichnung von Layern, die im Projekt *erforderlich* sind, also nicht entfernbar
- zur Kennzeichnung einer *eingebetteten Gruppe oder Layer* und dem Pfad zu ihrer ursprünglichen Projektdatei
- , um einen Layer zu identifizieren, dessen Datenquelle beim Öffnen der Projektdatei nicht verfügbar war (siehe *Behandlung von fehlerhaften Dateipfaden*). Klicken Sie auf das Symbol, um den Quellpfad zu aktualisieren, oder wählen Sie den Eintrag *Datenquelle reparieren…* aus dem Kontextmenü des Layers.
- , um Sie daran zu erinnern, dass der Layer ein *Temporärlayer* ist und sein Inhalt verworfen wird, wenn Sie das Projekt schließen. Um Datenverluste zu vermeiden und den Layer dauerhaft zu machen, klicken Sie auf das Symbol, um den Layer in einem der von QGIS unterstützten GDAL-Vektorformate zu speichern.
- , um einen Layer zu identifizieren, der im *Offline-Bearbeitungsmodus* verwendet wird.
- zur Kennzeichnung eines Layers, der kein/ein unbekanntes KBS hat
- für Layer mit Koordinaten, die in einem Koordinatenbezugssystem gespeichert sind, das von Haus aus eine geringe Genauigkeit aufweist (erfordert die Aktivierung der *entsprechenden Einstellung*)
- zur Kennzeichnung eines zeitlichen Layers, der über die Zeitsteuerung animiert werden kann
- zur Kennzeichnung eines Layers, dem *Bemerkungen* zugeordnet ist
- Ein ausgegrauter Name, wenn der aktuelle Maßstab der Kartenansicht außerhalb des Sichtbarkeitsbereichs des Layers liegt (wie in dessen *Darstellung*-Eigenschaften festgelegt). Wählen Sie die Option *Auf sichtbaren Maßstab zoomen* im Kontextmenü, um die Karte auf den nächstgelegenen Sichtbarkeitsbereich des Layers zu zoomen.

Die Darstellung von Layern durch Gruppierung steuern

Gruppen sind ein Mittel zur Strukturierung von Layern innerhalb eines Baums im Projekt, aber sie können sich auch auf die Art und Weise auswirken, wie ihre Komponenten-Layer gerendert werden, nämlich als ein einziges, abgeflachtes Objekt während des Kartenrenderns.

Die Option für eine solche Darstellung ist im Bedienfeld *Layergestaltung* verfügbar, wenn eine Gruppe ausgewählt ist. Aktivieren Sie unter dem Reiter Symbologie das Kontrollkästchen *Layer als Gruppe darstellen*, um eine Reihe von Optionen zu aktivieren, mit denen Sie das Erscheinungsbild der untergeordneten Layer als Ganzes und nicht als einzelne Layer steuern können:

- *Transparenz*: Merkmale aus untergeordneten Layern, die von anderen untergeordneten Layern verdeckt werden, bleiben verdeckt, und die Deckkraft gilt nur für die „gesamte Gruppe“.

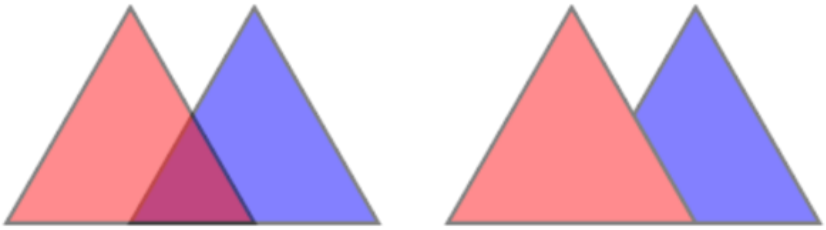


Abb. 8.2: Einstellen der Transparenz von einzelnen Layern im Vergleich zu Layer in einer Gruppe
Das linke Bild zeigt zwei Layer, die mit einer Deckkraft von 50 % dargestellt werden (die darunter liegenden Features sind sichtbar, aber durch das darüber liegende rote Feature mit einer Deckkraft von 50 % halbmaskiert). Das zweite Bild zeigt das Ergebnis der Einstellung der Deckkraft für die Gruppe (Teile der darunter liegenden blauen untergeordneten Ebene werden durch die darüber liegende rote Ebene vollständig verdeckt und das Ergebnis wird dann mit 50 % Deckkraft wiedergegeben).

- *Überblendmodi*: Genau wie die Transparenz führt das Setzen eines *Mischmodus* (wie Multiplikation, Überlagerung, …) für eine ganze Gruppe zunächst zu einer Verflachung der Merkmale der untergeordneten Layer, wobei die oberen die unteren verdecken. Das Rendering wird dann durch Überblenden der flachen Gruppe und der darunter liegenden Layer erreicht.
  - Wenn den untergeordneten Layern Mischmodi zugewiesen sind, werden diese vor dem Reduzieren angewendet, aber der Geltungsbereich ist darauf beschränkt, dass nur andere untergeordnete Layer dieser Gruppe betroffen sind, nicht aber andere Layer, die sich unterhalb der gesamten Gruppe befinden.
  - Für untergeordnete Layer in Gruppen stehen im Reiter *Symbologie* einige weitere Optionen zur Verfügung, die während des Renderns andere untergeordnete Layer „ausschneiden“. Sie können z.B. den Inhalt eines Layers durch den Inhalt eines zweiten „Masken“-Layers beschneiden.
- *Layereffekte*: wendet *Zeicheneffekte* nur auf die reduzierte Darstellung der untergeordneten Layer an; so würde z.B. ein Schlagschatteneffekt, der auf die Gruppe angewendet wird, für verdeckte untergeordnete Layer nicht sichtbar sein.

Wenn eine Gruppe auf *Layer als Gruppe darstellen* eingestellt ist, wird nur die Gruppe in der Liste des *Layerreihenfolge* Bedienfelds angezeigt. Die untergeordneten Ebenen der Gruppe sind in dieser Liste nicht sichtbar, da ihre Reihenfolge durch die Platzierung des Layers der Gruppe bestimmt wird.

Layer-Stil bearbeiten

Über das *Layer* Bedienfeld können Sie die Darstellung der Layer schnell und einfach ändern.

Klicken Sie mit der rechten Maustaste auf einen Layer und wählen Sie in der Liste *Stile* ►, um:

- sehen Sie die *Stile*, die derzeit für den Layer verfügbar sind. Wenn Sie mehrere Stile für den Layer definiert haben, können Sie von einem zum anderen wechseln und die Darstellung des Layers wird automatisch auf der Kartenansicht aktualisiert.
- Kopieren eines Teils oder des gesamten aktuellen Stils und ggf. Einfügen eines kopierten Stils von einem anderen Layer
- *Aktuellen umbenennen* … Stil
- Stil *Hizufügen*, der eine Kopie des aktuellen Stils ist
- *Aktuellen löschen*: nur wenn mehrere Stile verfügbar sind

> **Tipp**
>
> **Schnelles Teilen eines Layer-Stils**
>
> Kopieren Sie den Stil eines Layers aus dem Kontextmenü und fügen Sie ihn in eine Gruppe oder eine Auswahl von Layern ein: Der Stil wird auf alle Layer angewendet, die vom gleichen Typ sind (Vektor, Raster, Netz, Punktwolke, …) wie der ursprüngliche Layer und bei Vektorlayern, die den gleichen Geometrietyp haben (Punkt, Linie oder Polygon).

Wenn Sie eine Symbologie verwenden, die auf der Klassifizierung von Merkmalen basiert (z.B. *kategorisiert*, *abgestuft* oder *regelbasiert* für Vektorlayer oder *Klassifizierung* für Punktwolken-Layer), können Sie durch einen Rechtsklick auf einen Klasseneintrag im *Layer* Bedienfeld die Sichtbarkeit der Klassen (und ihrer Merkmale) bearbeiten und so vermeiden, dass sie einzeln umgeschaltet werden müssen:

- *Elemente umschalten*
- *Alle anzeigen*
- *Alle verstecken*

Bei Vektorlayern ermöglicht das Kontextmenü eines Klassen-Eintrags auch den Zugriff auf:

- *Objekte wählen*: wählt im Layer alle Objekte aus, die dieser Klasse entsprechen
- *In Attributtabelle zeigen*: öffnet eine Attributtabelle, die nur die Merkmale anzeigt, die dieser Klasse entsprechen
- die *Symbolfarbe* : unter Verwendung eines **Farbrads** anpassen. Der Einfachheit halber sind die zuletzt verwendeten Farben auch am unteren Ende des Farbkreises verfügbar.
- *Symbol bearbeiten…*: öffnet die *Symbolauswahl*, um das Klassen-Symbol zu ändern (Form, Größe, Farbe…).
- *Symbol kopieren*
- *Symbol einfügen*

> **Tipp**
>
> Ein Doppelklick auf einen Klassen-Eintrag öffnet auch die :guilabel:`Symbolauswahl.

8.2.2 Das Layergestaltung-Bedienfeld

Das Bedienfeld *Layergestaltung* (auch mit `Strg+3` zu aktivieren) ist eine Verknüpfung zu einigen Funktionen des *Layereigenschaften* Dialogs. Es bietet eine schnelle und einfache Möglichkeit, die Darstellung und das Verhalten eines Layers zu ändern, ohne den Layereigenschaften-Dialog öffnen zu müssen.

Mit dem Bedienfeld Layergestaltung lässt sich nicht nur das blockierende Layereigenschaften-Fenster verhindern, sondern auch die Überfrachtung des Bildschirms mit Dialogen: Wenn Sie z. B. auf Farbschaltflächen im Bedienfeld für die Layereigenschaften klicken, wird das Dialogfeld für die Farbauswahl im Bedienfeld für die Ebeneneigenschaften selbst und nicht in einem separaten Dialogfeld geöffnet.

Wählen Sie aus einer Dropdown-Liste der aktuellen Layer im Bedienfeld ein Element aus und:

- Je nach aktivem Element, einstellen:
  - *Symbologie* für Gruppen (see *Die Darstellung von Layern durch Gruppierung steuern*)
  - *Symbologie*, *Transparenz*, and *Histogramm* Eigenschaften für Rasterlayer. Diese Optionen sind die gleichen wie im *Der Eigenschaften-Dialog von Rasterlayern*.
  - *Symbologie*, *Beschriftungen*, *Masken* and *3D-Ansicht* Eigenschaften für Vektorlayer. Diese Optionen sind die gleichen wie im *Der Eigenschaften-Dialog von Vektorlayern* und können durch benutzerdefinierte Eigenschaften erweitert werden, die durch Plugins von Drittanbietern hinzugefügt werden.
  - *Symbology*, *Labels* and *3D View* properties for mesh layer. These options are the same as in the *Mesh Dataset Properties*.
  - *Symbology*, *3D View* and *Elevation* properties for point cloud or 3D Tiles layer. These options are the same as in the *Punktwolkenlayer-Eigenschaften* or *3D Tiles Eigenschaften*.
  - *Symbology* and *Labels* properties for vector tiles layer. These options are the same as in the *Eigenschaften von Vektorkacheldatensätzen*.
  - *Symbology* properties for annotations layer. These options are the same as in the *feature annotation properties*.
- Aktivieren und konfigurieren der *global map shading* Eigenschaften.
- Verwalten Sie die zugehörige(n) Stil(e) in der *Stilverwaltung* (weitere Einzelheiten unter *Verwalten von benutzerdefinierten Stilen*).
- Sehen Sie sich die Historie der Änderungen an, die Sie im aktuellen Projekt auf den Layer-Stil angewendet haben: Sie können daher jeden Zustand rückgängig machen oder wiederherstellen, indem Sie ihn in der Liste auswählen und auf *Anwenden* klicken.

Eine weitere leistungsstarke Funktion dieses Bedienfelds ist das Kontrollkästchen *Live update*. Aktivieren Sie das Kontrollkästchen, um Ihre Änderungen sofort auf der Kartenansicht darzustellen: Sie müssen dann nicht mehr auf die Schaltfläche *Anwenden* klicken.

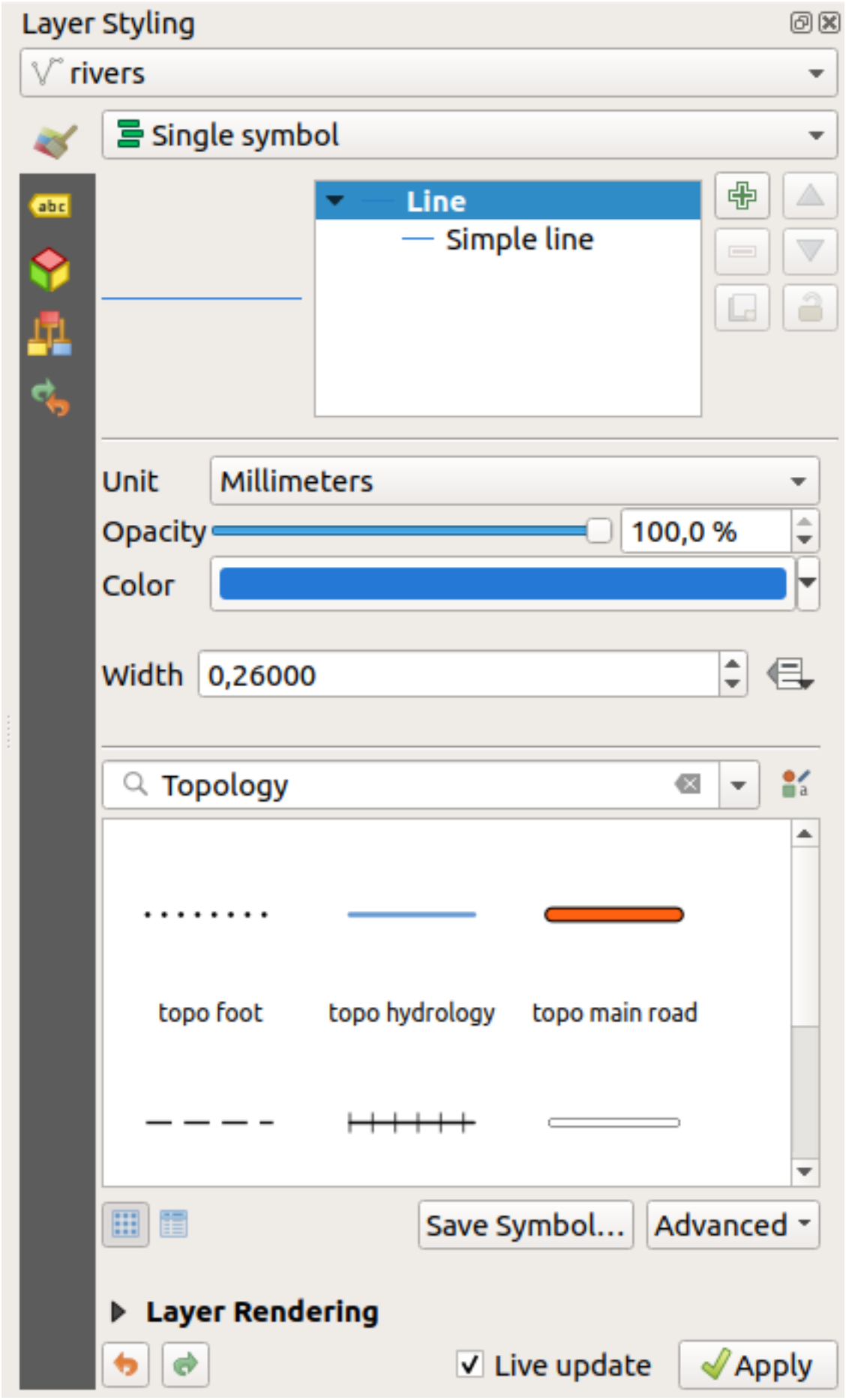


Abb. 8.3: Definieren der Symbologie eines Layers über das Layergestaltung-Bedienfeld

8.2.3 Das Layerreihenfolge-Bedienfeld

Standardmäßig werden die Layer in der Kartenansicht von QGIS in der Reihenfolge des Bedienfeldes *Layers* gezeichnet: je weiter oben sich ein Layer im Bedienfeld befindet, desto höher (und damit sichtbarer) wird er in der Kartenansicht gezeichnet.

Mit dem Bedienfeld *Layerreihenfolge* (Menü *Ansicht* ► *Bedienfelder* ► oder `Strg+9`) können Sie die Reihenfolge der Layer unabhängig von der Reihenfolge im Layer-Bedienfeld festlegen. Aktivieren Sie das Kontrollkästchen *Bestimmte Zeichenreihenfolge* unterhalb der Liste der Layer und ordnen Sie die Layer im Bedienfeld wie gewünscht an. Diese Reihenfolge wird auf die Kartenansicht angewendet. In Abb. 8.4 können Sie zum Beispiel sehen, dass die `Flughäfen` über dem `Alaska`-Polygon angezeigt werden, obwohl diese Layer im Bedienfeld platziert sind.

Die Deaktivierung von *Bestimmte Zeichenreihenfolge* kehrt zum Standardverhalten zurück.

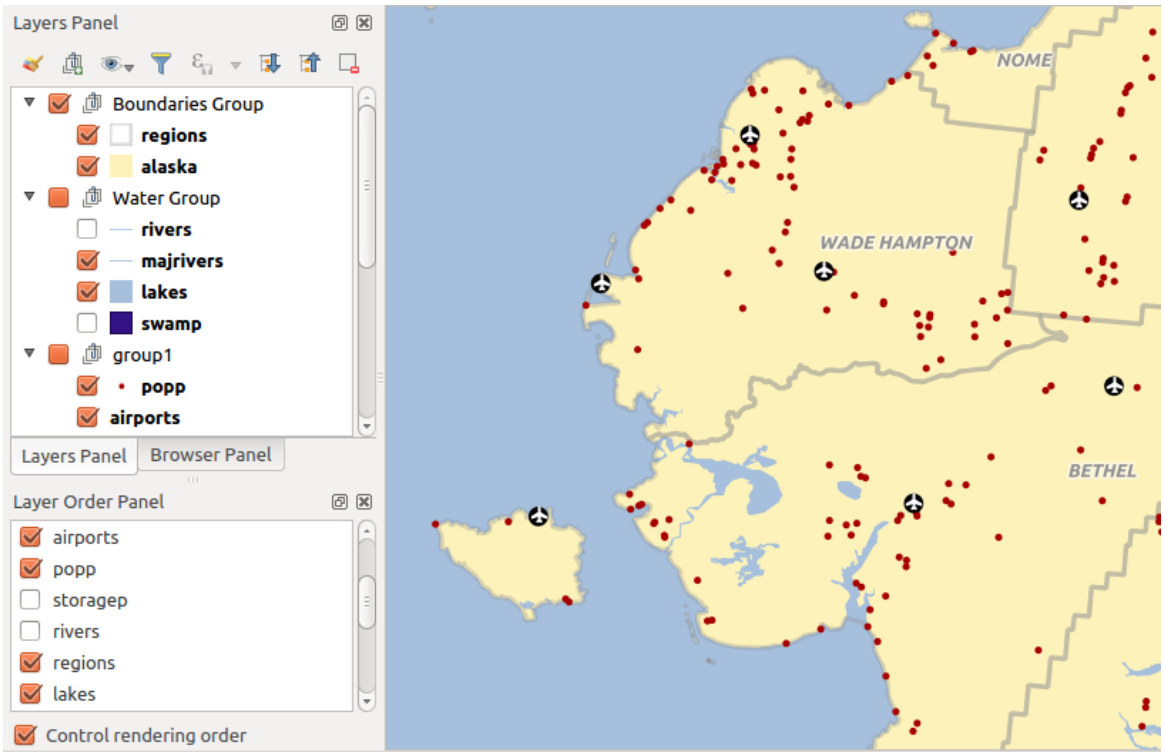


Abb. 8.4: Festlegung der Layer-Reihenfolge unabhängig von der Legende

8.2.4 Das Übersicht-Bedienfeld

Das Bedienfeld *Übersicht* (`Strg+8`) zeigt eine Karte mit ausgewählten Layern in ihrer vollen Ausdehnung an. Mithilfe der Option *In Übersicht anzeigen* aus dem *Layer* Menü oder dem Kontextmenü der Layer können LAyer ausgewählt werden, die in der Übersicht angezeigt werden sollen. Innerhalb der Übersicht zeigt ein rotes Rechteck die aktuelle Ausdehnung der Kartenansicht an, sodass Sie schnell feststellen können, welchen Bereich der gesamten Karte Sie gerade betrachten. Wenn Sie das rote Rechteck im Übersichtsrahmen anklicken und verschieben, wird die Ausdehnung der Hauptkartenansicht entsprechend aktualisiert.

Beachten Sie, dass Beschriftungen in der Übersichtskarte nicht angezeigt werden, auch wenn Beschriftungen für diese Layer eingerichtet wurden.

8.2.5 Das Protokoll-Bedienfeld

Beim Laden oder Verarbeiten einiger Vorgänge können Sie Nachrichten, die in verschiedenen Reitern erscheinen, mithilfe des Bedienfelds *Log Messages Panel* verfolgen. Das Bedienfeld kann kann über das Symbol ganz rechts in der unteren Statusleiste aktiviert werden.

8.2.6 Das Rücknahme/Wiederholung-Bedienfeld

Für jeden bearbeiteten Layer zeigt das Bedienfeld *Rückgängig/Wiederherstellen* (`Strg+5`) die Liste der durchgeführten Aktionen an, sodass Sie eine Reihe von Aktionen schnell rückgängig machen können, indem Sie die oben aufgeführte Aktion auswählen. Weitere Einzelheiten finden Sie unter *Rückgängig machen und Wiederherstellen von Bearbeitungen*.

8.2.7 Das Statistik-Bedienfeld

Das *Statistik* Bedienfeld (`Strg+6`) liefert zusammengefasste Informationen über jeden beliebigen Vektor-Layer. Mit diesem Bedienfeld können Sie auswählen:

- den Vektorlayer, für den die Statistik berechnet werden soll: die Optionen sind im oberen Dropdown-Menü wählbar oder werden mit dem aktiven Layer im *Layer* Bedienfeld über das Kontrollkästchen *Dem gewählten Layer folgen* am unteren Ende der Dropdown-Liste für die Statistik synchronisiert.
- das zu verwendende Feld oder den **|Ausdruck|** *Ausdruck*: für jeden Layer wird der letzte Eintrag gespeichert und bei erneuter Auswahl des Layers automatisch berechnet.
- die Statistiken, die zurückgegeben werden sollen, mithilfe der Dropdown-Schaltfläche unten rechts im Dialogfeld. Je nach Typ des Feldes (oder der Werte des Ausdrucks) sind folgende Statistiken verfügbar:

Tab. 8.2: Die für die verschiedenen Feldtypen verfügbaren Statistiken

Statistik	Text	Ganzzahl	Dezimalzahl	Datum
Anzahl	✓	✓	✓	✓
Anzahl (eindeutige)	✓			✓
Anzahl (fehlende)	✓	✓	✓	✓
Sum		✓	✓	
Mittel		✓	✓	✓
Standardabweichung		✓	✓	
Standardabweichung (Auswahl)		✓	✓	
Minimum	✓	✓	✓	✓
Maximum	✓	✓	✓	✓
Bereich		✓	✓	✓
Minderheit	✓	✓	✓	
Mehrheit	✓	✓	✓	
Varietät		✓	✓	
Q1		✓	✓	
Q3		✓	✓	
IQB		✓	✓	
Minimallänge	✓			
Maximallänge	✓			
Mittlere Länge	✓			

Die statistische Zusammenfassung kann sein:

- für den gesamten Layer oder *Nur ausgewählte Objekte*
- in die Zwischenablage kopiert und als Tabelle in eine andere Anwendung eingefügt
- Neuberechnung über die Schaltfläche |Aktualisieren|, wenn sich die zugrunde liegende Datenquelle ändert (z. B. neue oder entfernte Objekte/Felder, Änderung von Attributen)

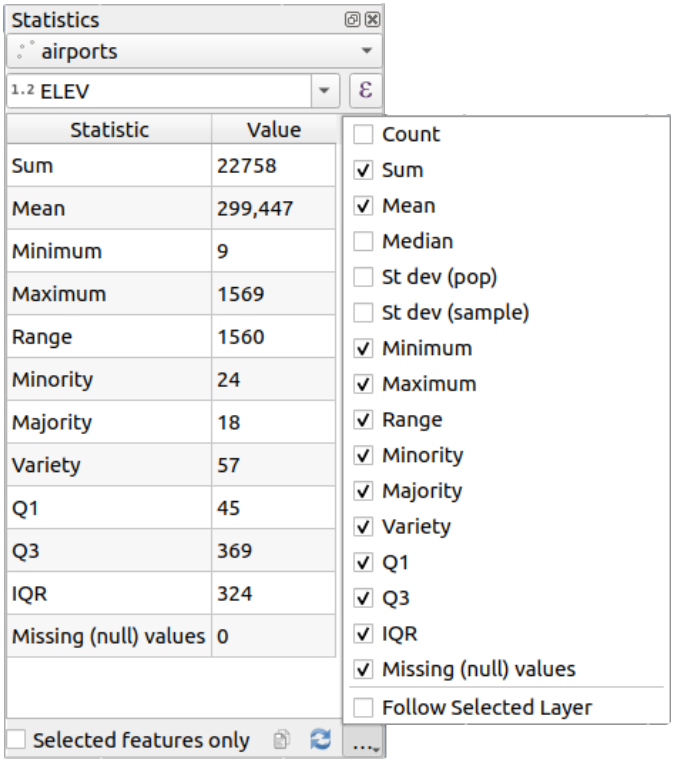


Abb. 8.5: Statistiken zu einem Feld anzeigen

8.2.8 Das Diagnose/Entwicklungswerkzeuge-Bedienfeld

Das *Diagnose/Entwicklungswerkzeuge* Bedienfeld (`F12`) bietet einen einheitlichen Ort für die Handhabung und das Debugging von Aktionen innerhalb von QGIS. Die verfügbaren Werkzeuge sind unter den folgenden Reitern organisiert:

- *API Documentation*: embeds developer oriented documentation
- *Netzwerkprotokoll*: Zeichnet Netzwerk-Anfragen und Interaktionen auf und zeigt sie an.
- *Abfrageprotokollierung*: Protokolliert von Datenanbietern ausgegebene SQL-Befehle und deren Ausführungsdauer.
- *Profilnahme*: Liefert Ladezeiten für alle Aktionen, um Ursachen für Verlangsamungen zu erkennen.

Bemerkung

Plugin-Autoren können das Bedienfeld mit benutzerdefinierten Reitern zum Debuggen und Entwickeln ihrer eigenen Plugins erweitern. Dies wird mit der Methode `registerDevToolWidgetFactory` gemacht.

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.

Netzwerkprotokoll

Der Reiter *Networkprotokoll* hilft Ihnen, Netzwerkanfragen aufzuzeichnen und anzuzeigen, zusammen mit einer ganzen Reihe nützlicher Details wie Anfrage- und Antwortstatus, Header, Fehler, SSL-Konfigurationsfehler, Timeouts, Cache-Status, etc.

In der oberen Werkzeugleiste haben Sie die folgenden Optionen:

- *Datensatzprotokoll*: mit dem die Protokollierung gestartet oder gestoppt wird.
- *Protokoll löschen*: löscht den Verlauf.
- *Protokoll speichern…*: zeigt zunächst eine Warnung an, dass das Protokoll sensibele Daten enthalten kann und vertraulich behandelt werden sollte, und erlaubt Ihnen dann, das Protokoll zu speichern.
- Klicken Sie auf das *Einstellungen* Dropdown-Menü, um die folgenden Optionen auszuwählen: *Erfolgreiche Abfagen anzeigen*, *Zeitüberschreitungen anzeigen* und *Aus dem Cache gelieferte Antworten anzeigen*.
- *Cache abschalten*: deaktiviert den Cache, sodass jede Abfrage neu ausgeführt werden muss.
- *Abfragen filtern* auf Grundlage von Teilen der URL oder des Abfragestatus

Mit einem Rechtsklick auf eine Abfrage können Sie:

- *URL öffnen* wodurch die URL in Ihrem Standardbrowser geöffnet wird.
- *URL kopieren*
- *Als cURL kopieren* um sie in einem Terminal zu verwenden.
- *Als JSON kopieren*: kopiert die Werte des Baums als Json-String in die Zwischenablage, um sie in Fehlerberichten oder für die Remote-Unterstützung verwenden zu können.

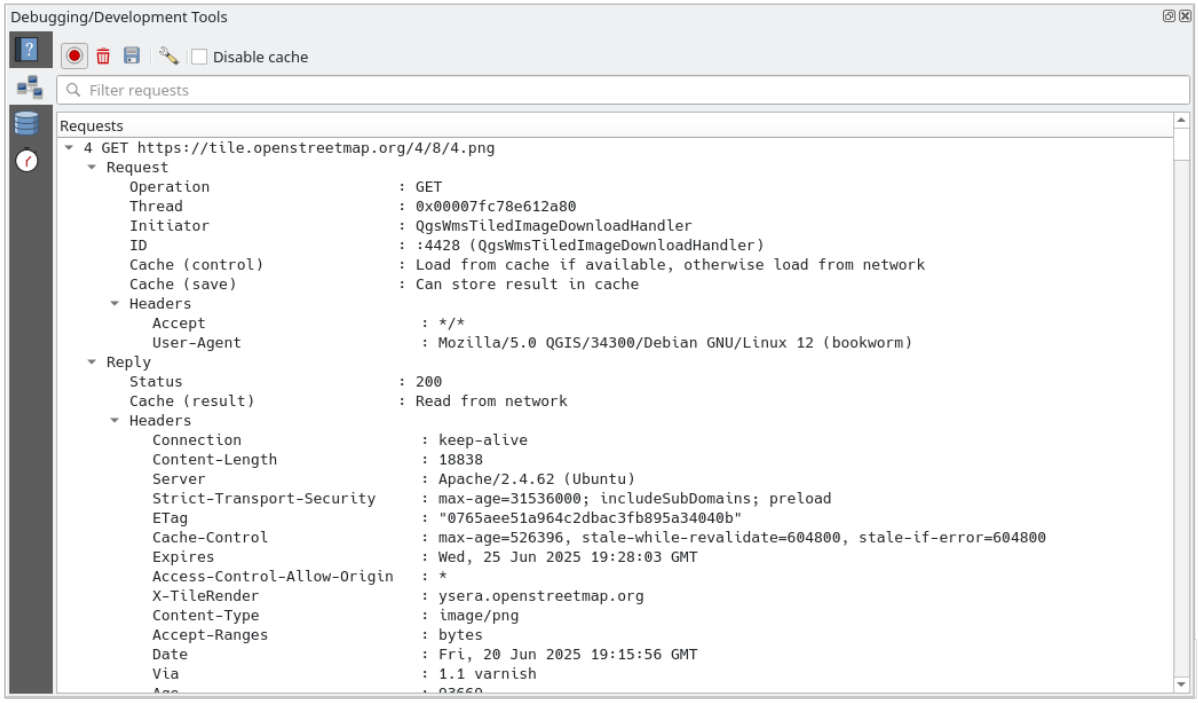


Abb. 8.6: Netzwerk-Protokoll-Ausgabe für eine GET-Anfrage

Abfrageprotokollierung

Die *Abfrageprotokollierung* ist der Ort, an dem die SQL-Befehle, die vom Datenprovider und der Verbindungs-API an die Backend-Datenbank gesendet werden, zusammen mit ihrer von QGIS gemessenen Ausführungszeit (d.h. in dem Client, der die Befehle gesendet hat) protokolliert werden. Dies kann nützlich sein, um die Leistung eines bestimmten Layers während der Fehlersuche oder der Entwicklung eines QGIS-Algorithmus oder Plugins zu untersuchen.

In der oberen Werkzeugleiste haben Sie die folgenden Optionen:

- *Datensatzprotokoll*: mit dem die Protokollierung gestartet oder gestoppt wird.
- *Protokoll löschen*: löscht den Verlauf.
- *Protokoll speichern…*: zeigt zunächst eine Warnung an, dass das Protokoll sensibele Daten enthalten kann und vertraulich behandelt werden sollte, und erlaubt Ihnen dann, das Protokoll zu speichern.
- *Abfragen filtern* basierend auf einem Teil der Zeichenkette oder Details wie Providertyp, Startzeit, Initiator, …

Klicken Sie mit der rechten Maustaste auf eine Abfrage, und Sie können:

- *SQL kopieren* kopiert den Befehl, der von QGIS in der Datenbank aufgerufen wird
- *Als JSON kopieren*: kopiert die Werte des Baums als Json-String in die Zwischenablage, um sie in Fehlerberichten oder für die Remote-Unterstützung verwenden zu können.

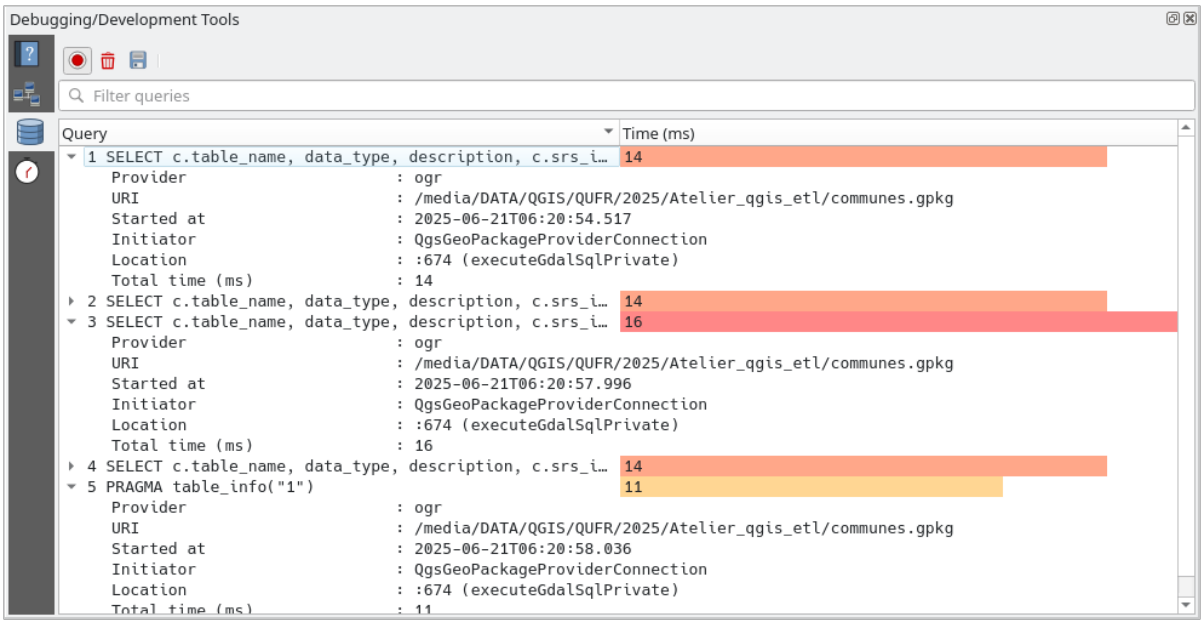


Abb. 8.7: Ausgabe einer Abfrageprotokollierung

Profilnahme

Der Reiter *Profilnahme* ermöglicht es, die Ladezeiten für jede einzelne Operation zu ermitteln, die durch Aktionen aufgerufen wurde, die der Nutzer ausgeführt hat. Je nach Kontext kann es sich dabei um das Lesen von Einstellungen, die Erstellung von Menüs, Kartenansichten oder 3D-Ansichten, das Auflösen von Referenzen auf Kartenebenen, das Laden von Lesezeichen oder Layouts, … handeln. Dies hilft bei der Ermittlung der Ursachen für langsame Ladezeiten.

Die folgenden Optionen können aus dem Dropdown-Menü *Kategorie* ausgewählt werden:

- *Start* hilft Ihnen, ungewöhnlich lange Startzeiten von QGIS zu diagnostizieren (und zu beheben).
- *Projektladen* ermöglicht es Ihnen, die verschiedenen Phasen der Projektladung aufzuschlüsseln, um die Ursachen für langsame Projektladezeiten zu ermitteln.
- *Kartenzeichnen* ermöglicht es Ihnen, Schwachstellen bei der Kartendarstellung zu identifizieren und genau zu ermitteln, welche Layer zu einem langen Neuaufbau der Karte führen.

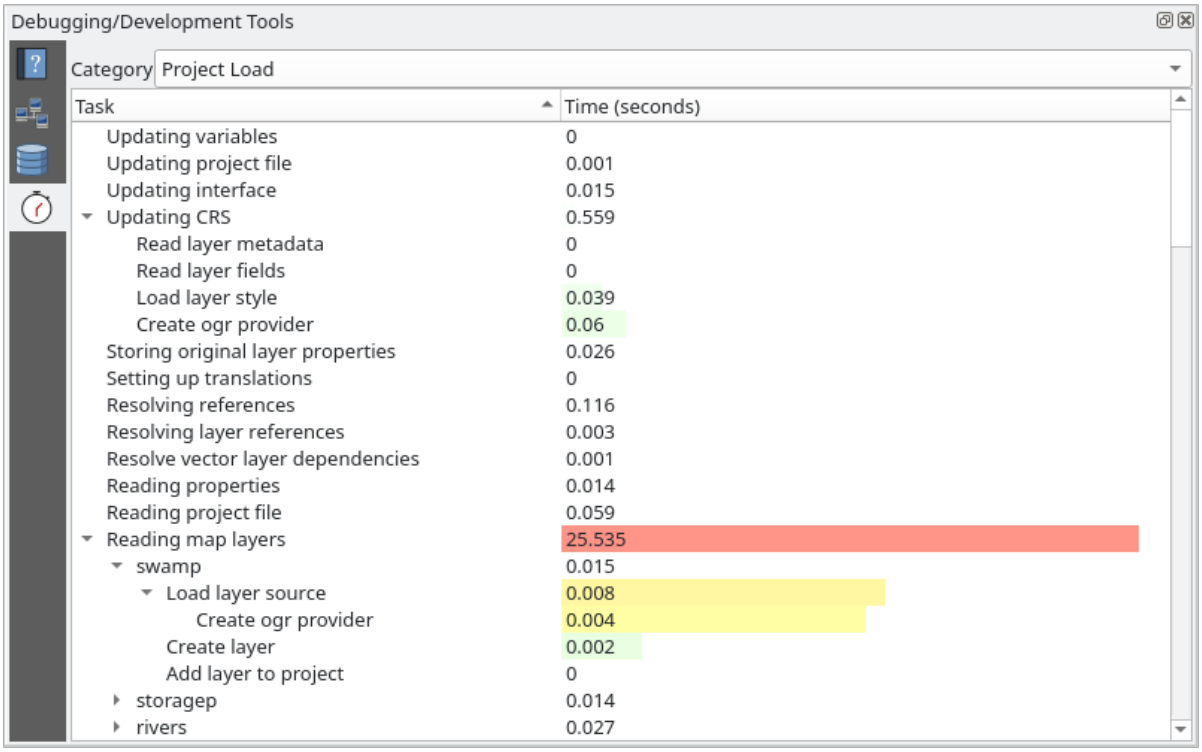


Abb. 8.8: Profilnahme für den Start von QGIS

8.3 Layer aus externen Projekten einbinden

Es kann vorkommen, dass Sie einen Layer in verschiedenen Projekten verwenden möchten, wobei dieser in allen Projekten denselben Stil haben soll. Dies können Sie entweder erreichen, indem Sie einen *Vorgabe-Stil* für diese Layer erstellen, Sie können aber auch einen Layer aus einem anderen Projekt einbetten und so denselben Stil in allen Projekten verwenden.

Das Einbetten von Layern und Gruppen aus einem bestehenden Projekt hat einige Vorteile gegenüber dem Styling:

- Alle Arten von Layern (Vektor oder Raster, lokal oder online…) können hinzugefügt werden
- Durch das Abrufen von Gruppen und Layern können Sie die gleiche Baumstruktur der „Hintergrund“-Layer in Ihren verschiedenen Projekten beibehalten
- Die eingebetteten Layer können zwar bearbeitet werden, aber ihre Eigenschaften, wie z. B. Symbolik, Beschriftungen, Formulare, Standardwerte und Aktionen, können nicht geändert werden, so dass die Konsistenz über alle Projekte hinweg gewährleistet ist.
- Ändern Sie die Elemente im ursprünglichen Projekt, werden die Änderungen auf alle anderen Projekte übertragen.

Wenn Sie Layer aus anderen Projektdateien in Ihr Projekt einbetten möchten, wählen Sie *Layer* ► *Eingebettete Layer und Gruppen…*:

1. Klicken Sie auf die Schaltfläche …, um nach einem Projekt zu suchen: Sie können dann den Inhalt des ausgewählten Projekts sehen (siehe Abb. 8.9)
2. Halten Sie `Strg` gedrückt und klicken Sie auf die Layer und Gruppen, die Sie einfügen möchten
3. Klicken Sie auf *OK*

Die ausgewählten Layer und Gruppen werden in das *Layer* Bedienfeld eingebettet und in der Kartenansicht angezeigt. Ein 🔗-Symbol wird neben ihrem Namen zur Kennzeichnung hinzugefügt, wenn Sie den Mauszeiger darüber bewegen, wird ein Tooltip mit dem ursprünglichen Dateipfad des Projekts angezeigt.

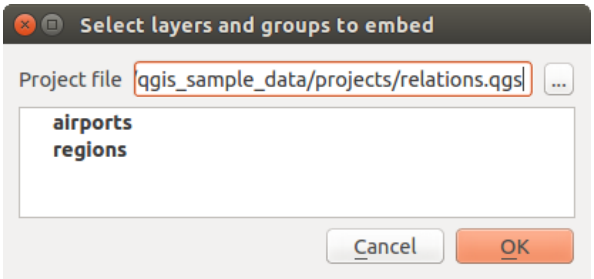


Abb. 8.9: Einzubettende Layer und Gruppen wählen

Wie jeder andere Layer kann ein eingebetteter Layer aus dem Projekt entfernt werden, indem Sie mit der rechten Maustaste auf den Layer klicken und auf Entfernen klicken.

Tipp

Die Symbologie eines eingebetteten Layers ändern

Es ist nicht möglich, die Symbolgie eines eingebetteten Layers zu ändern, dies kann nur in der ursprünglichen Projektdatei vorgenommen werden. Wenn Sie jedoch mit der rechten Maustaste auf einen Layer klicken und *Duplizieren* wählen, wird ein Layer mit vollem Funktionsumfang erstellt, der nicht vom Originalprojekt abhängig ist.

8.4 Interaktion mit Objekten

8.4.1 Objekte auswählen

QGIS bietet verschiedene Werkzeuge, um Objekte auf der Kartenansicht auszuwählen. Die Auswahlwerkzeuge sind im Menü *Bearbeiten* ► *Auswählen* oder in der :guilabel:`Auswahlwerkzeugleiste ` verfügbar.

Bemerkung

Auswahlwerkzeuge arbeiten mit dem aktuell aktiven Layer.

Manuelle Auswahl in der Kartenansicht

Um ein oder mehrere Objekte mit der Maus auszuwählen, können Sie eines der folgenden Werkzeuge verwenden:

- Objekte über Rechteck oder Einzelklick wählen
- Objekte durch Polygon wählen
- Objekte freihändig wählen
- Objekte über Radius wählen

Bemerkung

Anders als Objekte durch Polygon wählen, erlauben diese manuellen Auswahlwerkzeuge die Auswahl von Merkmalen auf der Kartenansicht mit einem einzigen Klick.

Bemerkung

Verwenden Sie das Werkzeug Objekte durch Polygon wählen, um ein vorhandenes Polygon-Objekt (aus einem Projekt-Layer) zu verwenden, um überlappende Objekte im aktiven Layer auszuwählen. Klicken Sie mit der rechten Maustaste auf das Polygon und wählen Sie es aus dem Kontextmenü, das eine Liste aller Polygone anzeigt, die den angeklickten Punkt enthalten. Alle überlappenden Features des aktiven Layers werden ausgewählt.

Tipp

Verwenden Sie das Werkzeug *Bearbeiten* ► *Auswählen* ► *Objekte wiederwählen*, um Ihre letzte Auswahl zu wiederholen. Sehr nützlich, wenn Sie eine große Auswahl getroffen haben und dann versehentlich auf eine andere Stelle klicken und Ihre Auswahl löschen.

Bei Verwendung des Werkzeugs *Objekte über Rechteck oder Einzelklick wählen* kann durch Halten von `Umschalt` oder `Strg` (`Cmd` bei macOS) umgeschaltet werden, ob ein Feature zur aktuellen Auswahl hinzugefügt oder aus ihr entfernt wird.

Bei den anderen Werkzeugen können verschiedene Verhaltensweisen durch Tastenkombinationen ausgeführt werden:

- `Umschalt`: Objekte zur aktuellen Auswahl hinzufügen
- `Strg`: Objekte aus der aktuellen Auswahl entfernen
- `Strg+Umschalt`/`Cmd+Umschalt`: sich mit der aktuellen Auswahl überschneiden, d.h. nur überlappende Objekte der aktuellen Auswahl behalten
- `Alt`: wählen Sie Features aus, die sich vollständig innerhalb der Auswahlform befinden. Kombiniert mit den Tasten `Umschalt` oder `Strg` können Sie Features zur aktuellen Auswahl hinzufügen oder von ihr abziehen.

Automatische Auswahl von Objekten

Die anderen Auswahlwerkzeuge, von denen die meisten über die *Attributtabelle* ausgeführt werden können, führen eine Auswahl basierend auf Attributen oder dem Auswahlstatus eines Objekts durch (beachten Sie, dass die Attributtabelle und die Kartenansicht die gleichen Informationen anzeigen, d.h. wenn Sie ein Objekt in der Attributtabelle auswählen, wird es auch in der Kartenansicht ausgewählt):

- Select Features By Expression… or press `Ctrl+F3` to select features using the expression dialog
- Objekte nach Wert wählen… (`F3`)
- Objekteauswahl aller Layern aufheben (`Strg+Alt+A`) um alle Objekte in allen Layern abzuwählen
- Objektauswahl des aktuellen aktiven Layers aufheben (`Strg+Umschalt+A`)
- Alle Objekte wählen (`Strg+A`) um alle Objekte des aktiven Layers auszuwählen
- Objekauswahl umkehren um die Auswahl der Objekte im aktiven Layer umzukehren
- Auswahl nach Lage zur Auswahl der Objekte auf der Grundlage ihrer räumlichen Beziehung zu anderen Objekten (auf demselben oder einem anderen Layer) - siehe auch *Auswahl nach der Lage*)
- Auswahl innerhalb Entfernung um Objekte auszuwählen, die sich innerhalb des angegebenen maximalen Abstands zu referenzierten Objekten befinden - siehe auch *Select within distance*)

Wenn Sie aus der Datei `regions.shp` der QGIS-Beispieldaten die Stadtbezirke („Borough“) finden möchten, können Sie wie folgt vorgehen:

1. Verwenden Sie das Objekte nach Ausdruck wählen Symbol
2. Erweitern Sie die *Felder und Werte* Gruppe in der mittleren Spalte
3. Doppelklicken Sie auf das Feld, das Sie abfragen möchten („TYPE_2“)
4. Klicken Sie auf *Alle eindeutigen* im Bedienfeld, das auf der unteren, rechten Seite angezeigt wird
5. Wenn Sie jetzt in der Liste auf „Borough“ doppelklicken, wird im Editorfeld *Ausdruck* die folgende Abfrage erzeugt:

```
"TYPE_2"  =  'Borough'
```

6. Wählen Sie dann *Objekte wählen*

Im Dialogfeld zur Erstellung von Ausdrücken können Sie auch *Funktionsliste* ► *Letzte (Auswahl)* verwenden, um eine Auswahl zu treffen, die Sie schon einmal verwendet haben. Der Dialog merkt sich die letzten 20 verwendeten Ausdrücke. Siehe *Ausdrücke* für weitere Informationen und Beispiele.

Tipp

Speichern Sie Ihre Auswahl in einer neuen Datei

Mit *Bearbeiten* ► *Objekte kopieren* und *Bearbeiten* ► *Objekte einfügen als* können ausgewählte Objekte in einem **Neuen Temporörläyer** oder einem **Neuen Vektorlayer** im gewünschten Format gespeichert werden.

Objekte nach Wert auswählen

Mit diesem Auswahlwerkzeug wird das Formular für die Objekte des Layers geöffnet, sodass der Benutzer für jedes Feld auswählen kann, nach welchem Wert gesucht werden soll, ob bei der Suche Groß- und Kleinschreibung beachtet werden soll und welche Operation verwendet werden soll. Das Werkzeug hat auch eine Autovervollständigung, die das Suchfeld automatisch mit vorhandenen Werten füllt.

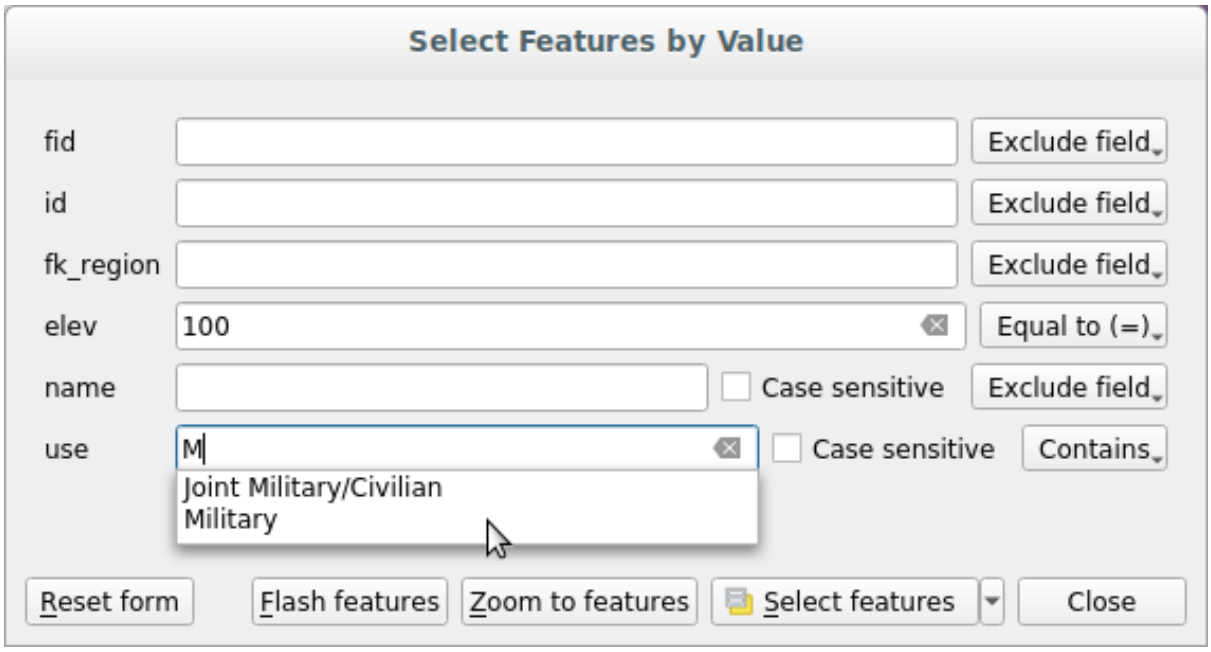


Abb. 8.10: Der Dialog zur Auswahl von Objekten anhand von Werten

Neben jedem Feld befindet sich eine Dropdown-Liste mit Optionen zur Steuerung des Suchverhaltens:

Tab. 8.3: Abfrage-Operatoren nach Datentyp

Option zur Feldsuche	Text	Numeric	Datum
Feld ausschließen von der Suche	☑	☑	☑
Gleich (=)	☑	☑	☑
Ungleich (≠)	☑	☑	☑
Größer als (>)		☑	☑
Kleiner als (<)		☑	☑
Größer gleich (≥)		☑	☑
Kleiner gleich (≤)		☑	☑
Zwischen (inklusive)		☑	☑
Nicht zwischen (inklusive)		☑	☑
Enthält	☑		
Enthält nicht	☑		
Fehlt (NULL)	☑	☑	☑
Fehlt nicht (nicht NULL)	☑	☑	☑
Beginnt mit	☑		
Endet mit	☑		

Für Text-Vergleiche ist es auch möglich, die Option ☑ *Case sensitive* (Groß-und Kleinschreibung beachten) zu verwenden.

Nachdem Sie alle Suchoptionen eingestellt haben, klicken Sie auf *Objekte auswählen*, um die passenden Objekte auszuwählen; dabei haben Sie die folgenden Optionen im Drop-down-Menü:

- *Objekte wählen*
- *Zu aktueller Auswahl hinzufügen*
- *Aus aktuelle Auswahl entfernen*
- *In aktueller Auswahl wählen*

Sie können auch alle Suchoptionen über die Schaltfläche *Formular zurücksetzen* löschen.

Sobald die Bedingungen festgelegt sind, können Sie auch entweder:

- *Zu Objekten zommen* auf der Kartenansicht, ohne dass eine Vorauswahl getroffen werden muss
- *Objekte aufleuchten lassen*, Hervorhebung der passenden Objekte. Dies ist eine praktische Methode, um ein Objekt ohne Auswahl oder Verwendung des Identifizierungswerkzeugs zu identifizieren. Beachten Sie, dass der Blitzeffekt die Ausdehnung der Kartenansicht nicht verändert und nur sichtbar ist, wenn sich das Objekt innerhalb der Grenzen der aktuellen Kartenansicht befindet.

8.4.2 Objekte abfragen

Das Werkzeug Objekte abfragen ermöglicht es Ihnen, mit der Kartenansicht zu interagieren und Informationen zu Objekten oder Pixeln in einem Pop-up-Fenster zu erhalten. Abgefragt werden können die meisten von QGIS unterstützten Layer-Typen (Vektor, Raster, Netz, Punktwolke, WMS, WFS, …). Um ein Element abzufragen, verwenden Sie entweder:

- *Ansicht* ► *Objekte abfragen*
- `Strg+Umschalt+I` (or ✗ `Cmd+Umschalt+I`),

- Objekte abfragen Schaltfläche in der *Attributwerkezugleiste*

Klicken Sie dann auf ein Objekt oder ein Pixel des aktiven Layers. Das identifizierte Element wird in der Kartenansicht hervorgehoben, während sich der Dialog *Identifikationsergebnisse* mit detaillierten Informationen dazu öffnet. Der Dialog zeigt auch eine Reihe von Schaltflächen für die erweiterte Konfiguration.

Das Identifikationsergebnisse-Fenster

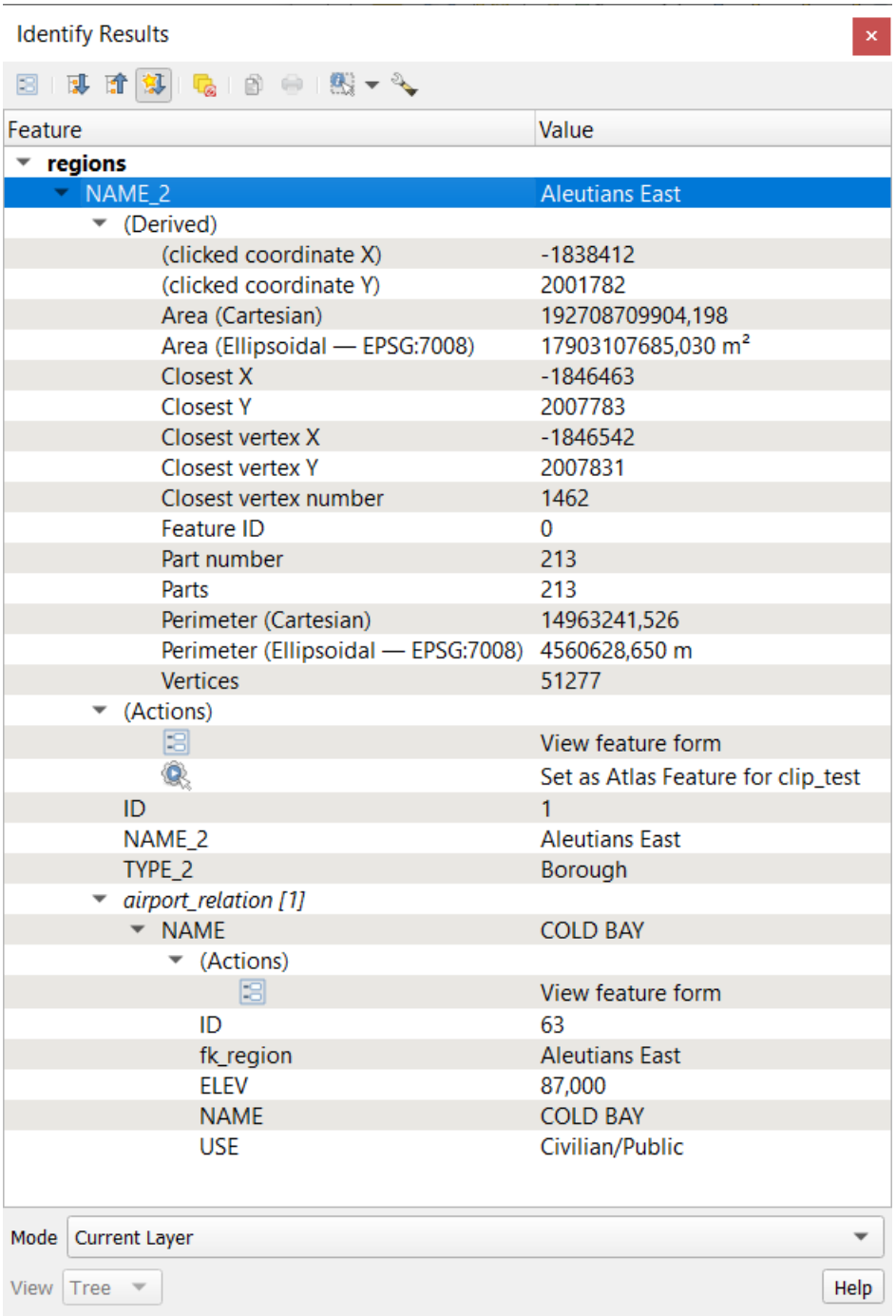


Abb. 8.11: Identifikationsergebnisse-Fenster

Von unten nach oben:

- Die *Sicht* steuert den allgemeinen Aspekt des Dialogs und die Formatierung der Ergebnisse; die Optionen können wie folgt gesetzt werden:
  - **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).
  - **Tabelle**: nur für rasterbasierte Layer verfügbar, erlaubt es, die Ergebnisse als Tabelle anzuzeigen, deren Spalten `Layer`, `FID`, `Attribut` und `Wert` sind
  - **Graf**: nur für rasterbasierte Layer verfügbar
- Der *Modus* hilft Ihnen bei der Auswahl der Layer, von denen Ergebnisse zurückgegeben werden sollen. Diese Layer müssen sichtbar sein, sodass die Daten in der Kartenansicht angezeigt werden, und in der *Projekteigenschaften* ► *Datenquellen* ► *Layereigenschaften* auf *abfragbar* gesetzt sein. Verfügbare Modi sind:
  - **Aktueller Layer**: nur der/die im *Layer* Bedienfeld ausgewählte(n) Layer liefern Ergebnisse. Wenn eine Gruppe ausgewählt ist, werden die Ergebnisse aus den Layern der Gruppe ausgewählt.
  - **Von oben nach unten, beim ersten halten**: die Ergebnisse stammen aus dem Layer des obersten Objekts oder Pixels unter der Maus.
  - **Von oben nach unten**: Die Ergebnisse stammen aus den Layern mit Objekten oder Pixeln unter der Maus.
  - **Layerauswahl**: öffnet ein Kontextmenü, in dem der Benutzer den Layer auswählt, aus dem Objekte identifiziert werden sollen. Wenn sich nur ein einziges Objekt unter der Maus befindet, werden die Ergebnisse automatisch angezeigt.
- 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:
  - Objektformular anzeigen des aktuell ausgewählten Objekts
  - Baum ausklappen
  - Baum einklappen
  - Aufklappen von neuen Ergebnissen voreinstellen um festzulegen, ob die Informationen des nächsten identifizierten Objekts komprimiert oder erweitert werden sollen
  - Ergebnisse löschen
  - Identifiziertes Objekt in Zwischenablage kopieren, geeignet zum Einfügen in anderen Anwendungen.
  - Gewählte HTML-Antwort drucken: eine textbasierte Formatierung der Ergebnisse zum Ausdrucken oder zum Speichern als `.PDF`-Datei
  - the *interaktive Objekte-abfragen-Werkzeug*: ein Dropdown-Menü mit Werkzeugen zur Auswahl von Objekten oder Pixeln auf der Kartenansicht, die identifiziert werden sollen
  - Unter Abfrageeinstellungen, können Sie aktivieren, ob Sie:
    - *Objektformular automatisch öffnen, wenn ein einzelnes Objekt abgefragt wird*: Wenn diese Option aktiviert ist, öffnet sich jedes Mal, wenn ein einzelnes Objekt identifiziert wird, ein Formular mit seinen Attributen. Dies ist eine praktische Möglichkeit, die Attribute eines Objekts schnell zu bearbeiten.

        * *Abgeleitet Attribute nicht in Ergebnissen anzeigen* um nur die tatsächlich im Layer definierten Felder anzuzeigen
        * *NULL-Werte der Ergebnisse verbergen*
        * *Show Relations* to show relations in the vector layer *Identify results tree*
    - Hilfe für den Zugriff auf die aktuelle Dokumentation

Das „Objekte abfragen“-Werkzeug benutzen

In der Standardanzeige (*Sicht: Baum*) bietet das Bedienfeld *Identifikationsergebnisse* mehrere Werkzeuge zur Interaktion mit den abgefragten Layern. Eine gezielte Kombination dieser Werkzeuge mit der *Ziel-Layer Auswahl* ermöglicht noch detailliertere Ergebnisse:

- Objekt(e) abfragen mit einem Klick oder einem Rechteck
    - Einfacher Klick oder Klicken-und-Ziehen: überlagernde Objekte in den Ziel-Layern werden zurückgegeben
    - Rechtsklick: Überlagernde Objekte aus den Ziel-Layern werden im Kontextmenü nach Layern gruppiert aufgelistet. Sie können dann Folgendes wählen:
        * displaydas Ergebnis für ein bestimmtes Objekt anzeigen, the result for a specific feature,
        * das Ergebnis für alle Objekte eines bestimmten Layers anzeigen,
        * für Vektorebenen ist es auch möglich, deren Attributtabelle gefiltert nach den zurückgegebenen Objekten zu öffnen
        * oder alle zurückgegebenen Objekte anzeigen.
- Objekt(e) unter Mauszeiger identifizieren: Bewegen Sie den Mauszeiger über die Kartenansicht und die Elemente in den Ziel-Layern werden hervorgehoben und im Bedienfeld für die Ergebnisse angezeigt.
- Objekte über Ploygon abfragen: gibt Elemente zurück, die ein gezeichnetes oder ausgewähltes Polygon überlappen.
    - Zeichnen Sie ein Polygon (klicken Sie mit der linken Maustaste, um einen Punkt hinzuzufügen, klicken Sie mit der rechten Maustaste, um das Polygon zu schließen), und alle überlagernden Objekte aus den Ziel-Layern werden hervorgehoben und im Bedienfeld angezeigt.
    - Klicken Sie mit der rechten Maustaste und Sie erhalten die Liste aller sichtbaren Polygonobjekte im Projekt unter dem Klick. Wählen Sie einen Eintrag aus und QGIS gibt alle Objekte aus den Ziel-Layern zurück, die das ausgewählte Polygon überlappen.
- Objekte freihand abfragen: gibt Elemente zurück, die sich mit einem freihändig gezeichneten Polygon überschneiden. Zeichnen Sie ein Polygon (klicken Sie mit der linken Maustaste, um zu beginnen, bewegen Sie den Zeiger, um den Bereich zu formen und klicken Sie mit der rechten Maustaste, um das Polygon zu schließen). Alle sich überlagernden Objekte von Ziel-Layern werden hervorgehoben und im Bedienfeld zurückgegeben.
- Identify Features by Radius returns items overlapping a drawn circle. Draw a circle (left-click to indicate the center point, move the pointer to shape the area or enter the radius in the pop-up text box and left-click or press `Enter` to validate the circle). All the overlaying features from target layers are highlighted and returned in the results panel.

> **Tipp**
>
> **Filtern Sie die abzufragenden Layer für das Objekte abfragen Werkzeug**
>
> Unter *Projekt* ► *Eigenschaften…* ► *Datenquellen* ► *Layereigenschaften*, können Sie die Spalte *Identifizierbar* neben einem Layer deaktivieren, um zu verhindern, dass dieser abgefragt wird, wenn Sie das Werkzeug

Objekte abfragen Werkzeug verwenden. Dies ist eine praktische Methode, um nur Objekte von Layern zu erhalten, die für Sie von Interesse sind.

Kontextmenü der Ergebnisse

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 Layer-Eigenschaften speichern und teilen

8.5.1 Verwalten von benutzerdefinierten Stilen

Wenn ein Layer zur Kartenansicht hinzugefügt wird, verwendet QGIS standardmäßig ein zufälliges Symbol/eine zufällige Farbe, um die Objekte darzustellen. Sie können jedoch ein Standardsymbol unter *Projekt* ► *Eigenschaften…* ► *Stile* festlegen, das auf jeden neu hinzugefügten Layer entsprechend seines Geometrietyps angewendet wird.

In den meisten Fällen möchten Sie jedoch lieber einen benutzerdefinierten und komplexeren Stil haben, der automatisch oder manuell (mit weniger Aufwand) auf die Layer angewendet werden kann. Dies können Sie mithilfe des Menüs *Stil* am unteren Rand des Layereigenschaften-Dialogs erreichen. In diesem Menü finden Sie Funktionen zum Erstellen, Laden und Verwalten von Stilen.

Ein Stil speichert verschiedene Informationen, die im Dialogfeld „Layereigenschaften“ oder „Projekteigenschaften“ festgelegt wurden, um den Layer zu rendern oder mit ihm zu interagieren. Dazu gehören:

Tab. 8.4: Komponenten des gespeicherten Stils und ihre Referenzen

Kategorie	Vector	Raster
Layer Konfiguration	*Layereigenschaften*	
	Anzeige	*Anzeige*
Symbolisierung	*Symbolisierung*	*Symbolisierung* ([1])
3D-Ansicht	*3D-Ansicht Eigenschaften*	nicht verfügbar
Beschriftungen	*Beschriftungen Eigenschaften*	nicht verfügbar
Felder	*Eigenschaften Felder* und *Restriktionen*	nicht verfügbar
Attributformular	*Attributformular*	nicht verfügbar
Aktionen	*Aktionen*	nicht verfügbar
Anzeigen	*Anzeige*	*Anzeige*
Diagramme	*Diagrammeigenschaften*	nicht verfügbar
Attributtabellenkonfiguration	*Arbeiten mit der Attributtabelle*	
Darstellung	*Darstellung*	*Darstellung* ([1])
Benutzereigenschaften	*Information*	*Information*
Geometrieoptionen	*Digitalisieren*	nicht verfügbar
Beziehungen	*Festlegung von Beziehungen zwischen mehreren Layern*	nicht verfügbar
Zeiteigenschaften	*Zeitlich*	*Eigenschaften Zeitlich*
Legendeneinstellungen	*Legende*	nicht verfügbar
Geländehöheneigenschaften	*Höhe*	*Höhe*
Hinweise	*Layernotizen*	

Standardmäßig wird der Stil, der auf einen geladenen Layer angewendet wird, `default` genannt. Sobald Sie die gewünschte Darstellung für Ihren Layer eingestellt haben, können Sie diese speichern, indem Sie auf das Kombinationsfeld *Stil* klicken und wählen:

- **Aktuellen umbennen**: Der aktive Stil wird umbenannt und mit den aktuellen Optionen angepasst
- **Hinzufügen…**: Ein neuer Stil wird unter Verwendung der aktuellen Optionen erstellt. Standardmäßig wird er in der QGIS-Projektdatei gespeichert. Siehe unten, um den Stil in einer anderen Datei oder einer Datenbank zu speichern
- **Aktuellen löschen**: Löschen Sie unerwünschte Stile, falls Sie mehr als einen Stil für den Layer definiert haben.

Unten in der Dropdown-Liste „Stil“ sehen Sie die Stile, die für den Layer definiert sind, wobei der aktive gekennzeichnet ist.

Beachten Sie, dass jedes Mal, wenn Sie im Dialogfeld für die Layer-Eigenschaften Änderungen vornehmen, der aktive Stil mit diesen aktualisiert wird.

Sie können beliebig viele Stile für einen Layer erstellen, aber es kann immer nur einer aktiv sein. In Kombination mit *Kartenthemen* bietet dies eine schnelle Möglichkeit, komplexe Projekte zu verwalten, ohne dass ein Layer in der Kartenlegende dupliziert werden muss.

Bemerkung

[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.

Da bei jeder Änderung der Layer-Eigenschaften die Änderungen im aktiven Stil gespeichert werden, sollten Sie immer sicherstellen, dass Sie den richtigen Stil bearbeiten, um zu vermeiden, dass Sie einen Stil versehentlich ändern, wenn dieser z. B. in einem *Kartenthema* verwendet wird.

Tipp

Stile über das Layer-Kontextmenü verwalten

Klicken Sie mit der rechten Maustaste auf den Layer im *Layer*- Bedienfeld, um Layer-Stile zu kopieren, einzufügen, hinzuzufügen oder umzubenennen. Sowohl für Vektor- als auch für Raster-Layer können Sie auswählen, welche Stilkategorien kopiert/eingefügt werden sollen. Wählen Sie *Alle Stilkategorien* oder eine der verfügbaren Kategorien, je nach Layer-Typ. Beachten Sie, dass bei Rasterlayern die Kategorien *Symbology* und *Darstellung* immer zusammen kopiert werden.

8.5.2 Speichern von Stilen in einer Datei oder einer Datenbank

Stile, die über das Kombinationsfeld *Stil* erstellt wurden, werden standardmäßig innerhalb des Projekts gespeichert und können von Layer zu Layer kopiert und eingefügt werden. Es ist jedoch auch möglich, sie außerhalb des Projekts zu speichern, um sie in einem anderen Projekt zu laden.

Stil als Text-Datei speichern

Wenn Sie auf *Stil* ► *Aktuellen Stil speichern…* klicken, können Sie den Stil speichern als:

- QGIS Layerstildatei (`.qml`)
- SLD-Datei (`.sld`) (nur für Vektorlayer)

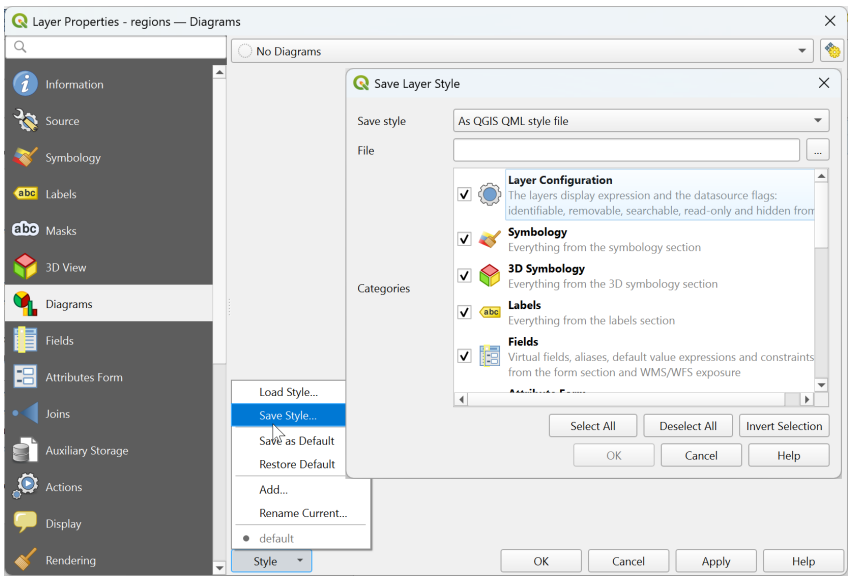


Abb. 8.12: Die Optionen in der Kombo-Box für den Stil eines Vektorlayers

Bei Layern im Dateiformat (`.shp`, `.tab`…) erzeugt *Als Vorgabe speichern* eine `.qml`-Datei für den Layer (mit demselben Namen und im selben Ordner). SLDs können für verschiedene Symbolisierungen (Einzelsymbol, kategorisiert, graduiert oder regelbasiert) exportiert werden, beim Importieren eines SLDs wird jedoch entweder ein Einzelsymbol oder ein regelbasierte Symbolisierung erstellt. Das bedeutet, dass kategorisierte oder graduierte Stile

in regelbasierte umgewandelt werden. Wenn Sie diese Symbolisierungen beibehalten wollen, müssen Sie das QML-Format verwenden. Andererseits bietet dies die Möglichkeit, um Stile einfach in regelbasierte zu konvertieren.

Stil in einer Datenbank speichern

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.

Sie können mehrere Stile für einen Layer in der Datenbank speichern. Jeder Layer kann jedoch nur einen Standardstil haben. Standardstile können in der Layer-Datenbank oder in `qgis.db`, einer lokalen SQLite-Datenbank im aktiven *Benutzerprofil* Verzeichnis, gespeichert werden.

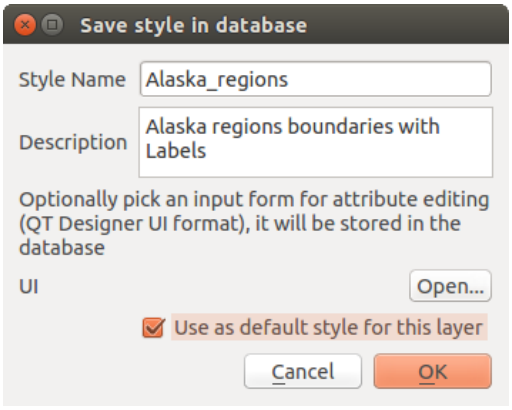


Abb. 8.13: Dialog zum Speichern eines Layer-Stils in einer Datenbank

Tipp

Layer-Stile zwischen Datenbanken austauschen

Sie können einen Stil nur in einer Datenbank speichern, wenn der Layer aus einer solchen Datenbank stammt. Sie können keine Datenbanken mischen (z. B. Layer in Oracle und Stil in MS SQL Server). Verwenden Sie stattdessen eine einfache Textdatei, wenn Sie möchten, dass der Stil zwischen verschiedenen Datenbanken ausgetauscht wird.

Bemerkung

Bei der Wiederherstellung der Tabelle `layer_styles` aus einer PostgreSQL-Datenbanksicherung können Probleme auftreten. Folgen Sie *QGIS layer_style Tabelle und Datenbank Backup*, um dies zu beheben.

Laden eines Stils

Wenn beim Laden eines Layers in QGIS bereits ein Standardstil für diesen Layer existiert, lädt QGIS den Layer mit diesem Stil. Auch *Stil* ► *Vorgabe wiederherstellen* sucht und lädt diese Datei und ersetzt den aktuellen Stil des Layers.

Stil ► *Lade Stil* hilft Ihnen, jeden gespeicherten Stil auf einen Layer anzuwenden. Während Textdateistile (`.sld` oder `.qml`) auf jeden Layer unabhängig von seinem Format angewendet werden können, ist das Laden von in einer Datenbank gespeicherten Stilen nur möglich, wenn der Layer aus derselben Datenbank stammt oder der Stil in der lokalen QGIS-Datenbank gespeichert ist.

Das Dialogfeld *Datenbank-Stilmanager* zeigt eine Liste der Stile an, die sich auf den in der Datenbank gefundenen Layer beziehen, sowie alle anderen in der Datenbank gespeicherten Stile, mit Namen und Beschreibung.

> **Tipp**
>
> **Schnelles Teilen eines Layer-Stils innerhalb des Projekts**
>
> Sie können Layer-Stile auch innerhalb eines Projekts gemeinsam nutzen, ohne einen Datei- oder Datenbankstil zu importieren: Klicken Sie mit der rechten Maustaste auf einen Layer im „Layer“ Bedienfeld und kopieren Sie den Stil eines Layers aus dem Kombinationsfeld „Stile“ und fügen Sie ihn in eine Gruppe oder eine Auswahl von Layern ein: Der Stil wird auf alle Layer angewendet, die vom gleichen Typ (Vektor oder Raster) wie der ursprüngliche Layer sind und im Falle von Vektorlayern den gleichen Geometrietyp (Punkt, Linie oder Polygon) aufweisen.

8.5.3 Layerdefinitionsdatei

Layer-Definitionen können als `Layerdefinitionsdatei` (**:datei:`.qlr`**) gespeichert werden, indem man *Exportieren* ► *Als Layerdefinitionsdatei speichern…* im Kontextmenü der aktiven Layer benutzt. Eine Layerdefinitionsdatei (`.qlr`) enthält Verweise auf die Datenquelle des Layers und die dazugehörigen Stile. `.qlr`-Dateien werden im Browser-Bedienfeld in QGIS angezeigt und können verwendet werden, um die Layer (mit dem gespeicherten Stil) zum Projekt hinzuzufügen. Sie können `.qlr`-Dateien auch aus ihrem Systemdateimanager in die Kartenansicht ziehen und ablegen.

8.6 Dokumentation Ihrer Daten

Neben der Darstellung und Symbolisierung der Daten in den Layern erlaubt QGIS auch das Eintragen von:

- **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.
- **Notizen**: Anmerkungen und Kommentare zu dem Layer im aktuellen Projekt

8.6.1 Metadaten

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:

- *Identifikation*: Grundlegende Zuordnung des Datensatzes (übergeordneter Datensatz, Kennung, Titel, Zusammenfassung, Sprache…);
- *Kategorien* in die die Daten gehören; neben den **ISO**-Kategorien können Sie auch eigene Kategorien hinzufügen;
- *Schlüsselwörter* zum Abrufen der Daten und der zugehörigen Konzepte anhand eines standardisierten Vokabulars;
- *Zugriff* auf die Daten (Lizenzen, Gebühren, Rechte, Restriktionen)
- *Ausdehnung* des Datensatzes, sowohl räumlich (KBS, Kartenausdehnung, Höhenlage) als auch zeitlich;
- *Kontakt*- Daten des Eigentümers oder Erstellers des Datensatzes;
- *Links* zu ergänzenden Ressourcen und weiteren Informationen;
- *Protokoll* zum Datensatz.

Eine Zusammenfassung der ausgefüllten Informationen wird im Reiter *Validierung* angezeigt und hilft Ihnen, mögliche Probleme im Zusammenhang mit dem Formular zu erkennen. Sie können sie dann entweder beheben oder ignorieren.

Metadaten werden standardmäßig in der Projektdatei gespeichert. Das Dropdown-Menü *Metadata* bietet Optionen zum Laden/Speichern von Metadaten aus der Datei `.qmd` und zum Laden/Speichern von Metadaten am „Standard“-Speicherort.

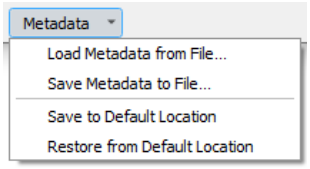


Abb. 8.14: Optionen beim Speichern und Laden von Metadaten

Der „Standard“-Speicherort, der von *In Vorgabeposition speichern* und *Aus Vorgabeposition laden* verwendet wird, hängt von der zugrunde liegenden Datenquelle und ihrer Konfiguration ab:

- Für PostgreSQL-Datenquellen, bei denen die Konfigurationsoption *Speichern/Laden von QGIS Layer-Metadaten in der Datenbank erlauben* aktiviert ist, werden die Metadaten in einer eigenen Tabelle in der Datenbank gespeichert.
- Bei GeoPackage-Datenquellen speichert *In Vorgabeposition speichern* die Metadaten immer in den internen Metadatentabellen des GeoPackage.

  Wenn Metadaten in den internen Tabellen von PostgreSQL oder GeoPackage gespeichert werden, stehen sie für die Suche und Filterung im Browser und im Bedienfeld *Layer-Metadaten-Suche* zur Verfügung.
- Für alle anderen dateibasierten Datenquellen speichert *In Vorgabeposition speichern* die Metadaten in einer `.qmd` Datei im gleichen Ordner wie die Daten selber.
- In allen anderen Fällen speichert *In Vorgabeposition speichern* die Metadaten in einer lokalen `.sqlite` Datenbank.

8.6.2 Layernotizen

Layernotizen ermöglichen es Ihnen, den Layer im aktuellen Projekt zu dokumentieren. Sie können wichtige Nachrichten für die Benutzer des Projekts wie Aufgabenlisten, Anweisungen, Warnungen, … speichern.

Wählen Sie aus dem Kontextmenü des Layers im *Layer* Bedienfeld die Option *Layernotiz hinzufügen…* und tragen Sie in den geöffneten Dialog die gewünschten Informationen ein.

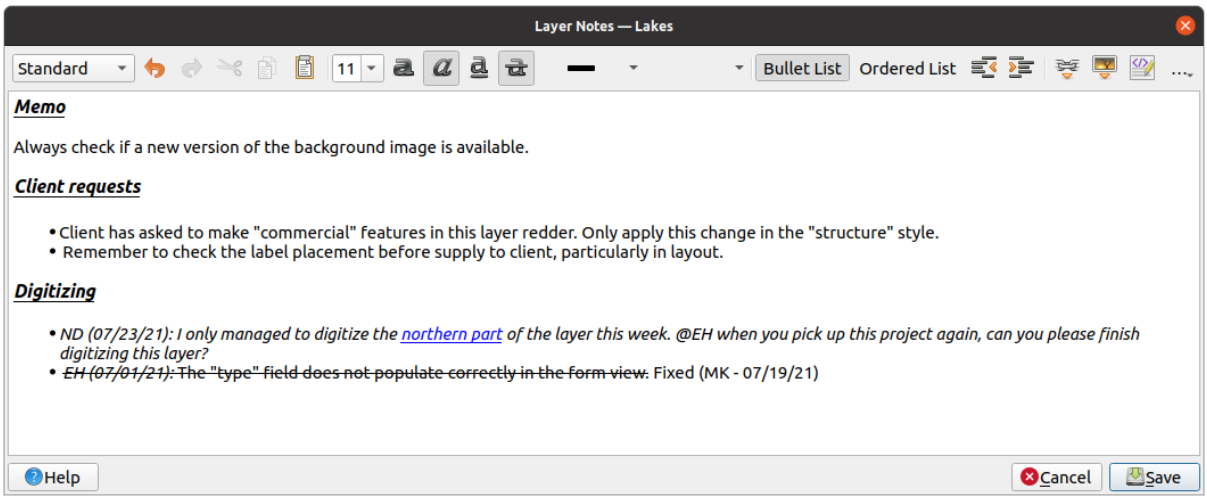


Abb. 8.15: Das Layernotizen-Fenster

Das *Layernotizen* Fenster bietet ein html-basiertes, mehrzeiliges Textfeld mit einem kompletten Satz von Werkzeugen für:

- Textbearbeitung: Ausschneiden, Kopieren, Einfügen, Rückgängig machen, Wiederherstellen
- Zeichenformatierung, die auf den gesamten Inhalt oder Teile davon angewendet wird: Schriftgröße und -farbe, fett, kursiv, unterstrichen, durchgestrichen, Hintergrundfarbe, URL-Hervorhebung
- Absatzstrukturierung: Aufzählungen und nummerierte Listen, Einrückung, vordefinierte Überschriften
- Einfügen von Dateien, auch mit Drag-and-Drop
- Bearbeitung mit HTML-Codierung

Über die Dropdown-Liste … ganz rechts in der Werkzeugleiste können Sie:

- *Alle Formatierungen entfernen*
- *Zeichenformatierung entfernen*
- *Ganzen Inhalt leeren*

Im *Layer* Bedienfeld wird bei einem Layer mit einer Notiz das Symbol angezeigt, wenn sie den Mauszeiger darüber bewegen, wird die Notiz angezeigt. Klicken Sie auf das Symbol, um die Notiz zu bearbeiten. Sie können auch mit der rechten Maustaste auf den Layer klicken und *Layernotiz bearbeiten…* oder *Layernotiz entfernen* wählen.

Bemerkung

Layernotizen sind Teil des *Layer-Stils* und können in einer `.qml` oder `.qlr`-Datei gespeichert werden. Sie können auch von einem Layer auf einen anderen übertragen werden, indem man den Layer-Stil kopiert und einfügt.

8.7 Speichern von Werten in Variablen

In QGIS können Sie Variablen verwenden, um nützliche, wiederkehrende Werte zu speichern (z.B. den Titel des Projekts oder den vollständigen Namen des Benutzers), die in Ausdrücken verwendet werden können. Variablen können auf der globalen Ebene der Anwendung, der Projektebene, der Layer-Ebene, der Verarbeitungsmodellierer-Ebene, der Layout-Ebene und der Ebene der Layout-Elemente definiert werden. Genau wie CSS-Kaskadierungsregeln können Variablen überschrieben werden - z. B. überschreibt eine Variable auf Projektebene alle Variablen auf globaler Anwendungsebene, die mit demselben Namen festgelegt wurden. Sie können diese Variablen verwenden, um Textstrings oder andere benutzerdefinierte Ausdrücke zu erstellen, indem Sie das Zeichen `@` vor dem Variablennamen verwenden. Zum Beispiel können Sie im Drucklayout eine Beschriftung mit folgendem Inhalt erstellen:

```
This map was made using QGIS [% @qgis_version %]. The project file for this
map is: [% @project_path %]
```

Auf der Karte würde diese Beschriftung dann wie folgt aussehen:

```
This map was made using QGIS 3.4.4-Madeira. The project file for this map is:
/gis/qgis-user-conference-2019.qgs
```

Neben den *vorgegebenen Variablen*, können Sie Ihre eigenen, benutzerdefinierten Variablen für jede der oben genannten Ebenen definieren. Sie können verwalten:

- **globale Variablen** über das Menü *Einstellungen* ► *Optionen*
- **Projektvariablen** über den Dialog *Projekteigenschaften* (siehe *Projekteigenschaften*)
- **Vektor-Layer-Variablen** aus dem *Layer-Eigenschaften*-Dialog (siehe *Der Eigenschaften-Dialog von Vektorlayern*);
- **Modellierer-Variablen** aus dem *Model Designer*-Dialog (siehe *Ein Modell erstellen*);
- **Layout-Variablen** aus dem *Layout* Bedienfeld im Drucklayout (siehe *Das Layout-Bedienfeld*);

- **Layoutelementvariablen** aus dem *Elementeigenschaften* Bedienfeld im Drucklayout (siehe *Gemeinsame Optionen der Layout-Elemente*).

Zur Unterscheidung von bearbeitbaren Variablen werden die Namen und Werte von schreibgeschützten Variablen kursiv dargestellt. Dagegen sind Variablen auf höherer Ebene, die von Variablen auf niedrigerer Ebene überschrieben werden, durchgestrichen.

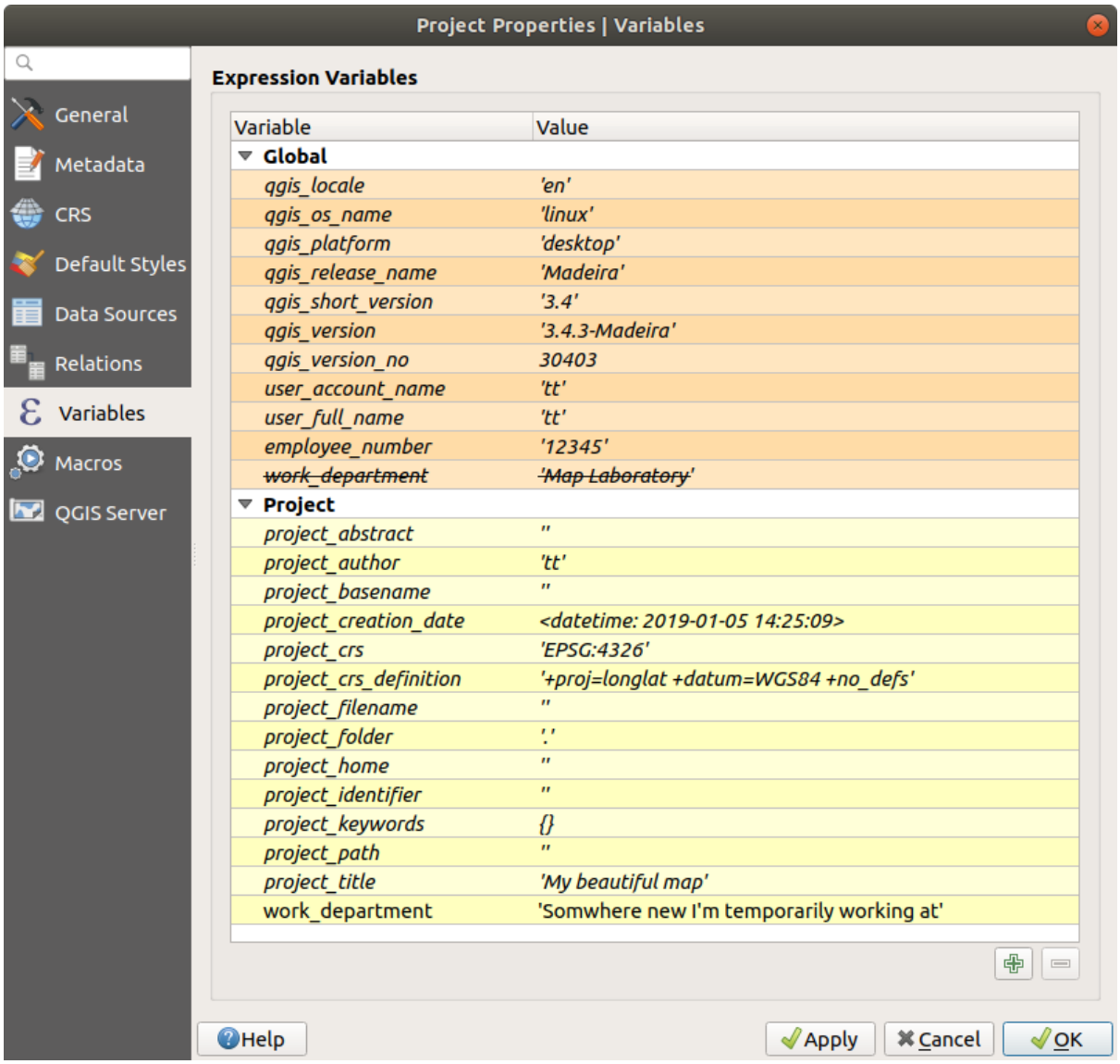


Abb. 8.16: Variablen auf Projektebene bearbeiten

Bemerkung

Mehr über Variablen und einige Beispiele finden Sie in Nyall Dawsons Blogbeiträgen Exploring variables in QGIS 2.12, part 1, part 2 und part 3.

8.8 Häufige wiederkehrende Werkzeuge

In QGIS gibt es einige Einstellungen, mit denen Sie oft arbeiten müssen. Zur Erleichterung stellt QGIS Ihnen spezielle Werkzeuge (Widgets) zur Verfügung, die im Folgenden vorgestellt werden.

8.8.1 Farbauswahl

Der Farbauswahl-Dialog

Der Dialog *Farbe wählen* erscheint immer dann, wenn Sie auf das Symbol klicken, um eine Farbe auszuwählen. Die Objekte dieses Dialogs hängen der Einstellung *Native Farbauswahldialoge verwenden* in *Einstellungen* ► *Optionen…* ► *Allgemein*. Wenn diese Option aktiviert ist, wird der native Farbdialog Ihres Betriebssystems verwendet. Andernfalls wird die Farbauswahl von QGIS verwendet (Standard).

Das QGIS-Dialogfeld für die Farbauswahl verfügt über vier verschiedene Reiter, mit denen Sie Farben über Farbverlauf, Farbkreis, Farbproben oder Farbwahl auswählen können. Mit den ersten beiden Reitern können Sie alle möglichen Farbkombinationen auswählen und auf das Objekt anwenden.

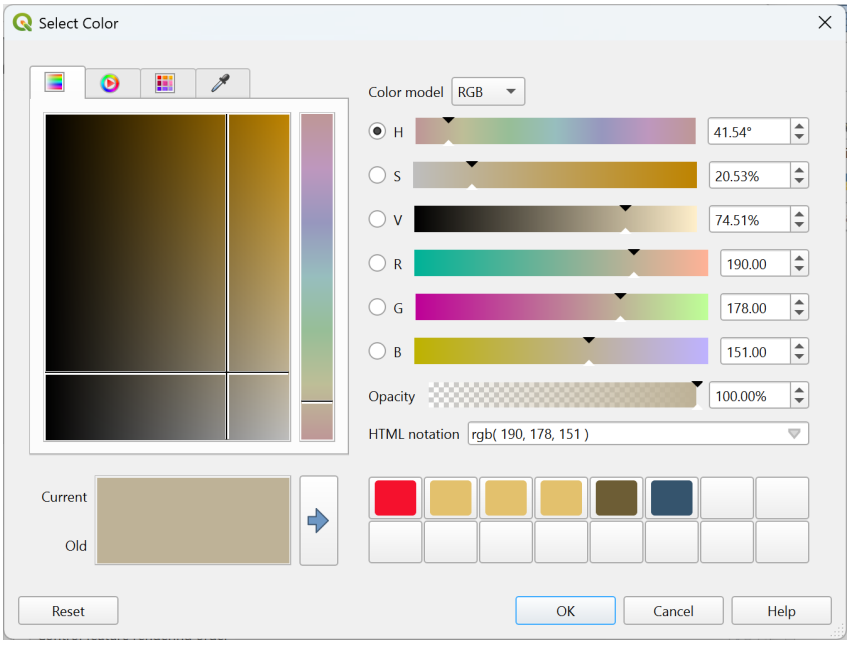


Abb. 8.17: Der Farbverlauf-Reiter

Im Reiter Farbproben können Sie aus einer Liste von Farbpaletten wählen (siehe *Farben*). Alle Paletten außer der Palette *Zuletzt verwendete Farben* können mit den Schaltflächen Aktuelle Farbe hinzufügen und Gewählte Farbe entfernen am unteren Rand des Rahmens geändert werden.

Die Schaltfläche … neben dem Kombinationsfeld der Palette bietet mehrere Optionen:

- Farben kopieren, einfügen, importieren oder exportieren
- Farbpaletten erstellen, importieren oder entfernen
- Hinzufügen der benutzerdefinierten Palette zum Farbauswahl-Widget mit dem Element *In Farbknöpfen anzeigen* (siehe Abb. 8.19)

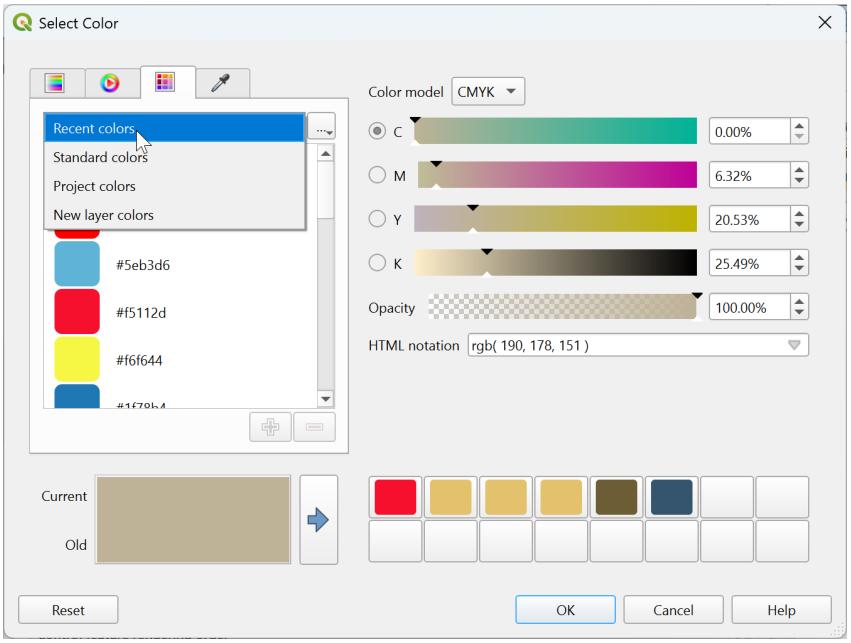


Abb. 8.18: Der Farbproben-Reiter

Eine weitere Möglichkeit ist die Verwendung der Farbwahl, mit dem Sie eine Farbe unter Ihrem Mauszeiger an einer beliebigen Stelle der QGIS-Oberfläche oder auch aus einer anderen Anwendung heraus auswählen können: Drücken Sie die Leertaste, während der Reiter aktiv ist, bewegen Sie die Maus über die gewünschte Farbe und klicken Sie darauf oder drücken Sie erneut die Leertaste. Sie können auch auf die Schaltfläche *Farbe abgreifen* klicken, um die Farbwahl zu aktivieren.

Unabhängig davon, welche Methode Sie verwenden, wird die ausgewählte Farbe immer über eine *Farbmodell* beschrieben, das standardmäßig auf das *Projektfarbmodell* eingestellt ist:

- *RGB*: Die Farbe wird mithilfe von Schiebereglern für die Werte `HSV` (Farbton, Sättigung, Wert) oder `RGB` (Rot, Grün, Blau) eingestellt.
- *CMYK* (Cyan, Magenta, Gelb, Schwarz)

Um eine Farbe zu ändern, klicken Sie einfach auf das Farbrad oder die Rampe oder auf einen der Farbparameter-Schieberegler. Sie können die Farbparameter auch mit der Spinbox daneben oder durch Scrollen des Mausrads über den entsprechenden Schieberegler einstellen. Sie können die Farbe auch in HTML-Schreibweise eingeben. Darüber hinaus steht ein *Deckkraft*-Schieberegler zur Einstellung der Transparenz zur Verfügung.

Der Dialog bietet auch einen visuellen Vergleich zwischen der vorherigen Farbe (*Alt*) und der Farbe, die Sie in der Farbauswahl eingestellt haben (*Aktuell*). Durch Drag-and-Drop oder Drücken der Schaltfläche Farbe zu Farbfeld hinzufügen kann jede dieser Farben in einem Slot gespeichert werden, um einen einfachen Zugriff zu ermöglichen.

Tipp

Schnelles Ändern der Farbe

Ziehen Sie ein Farbwähler-Widget auf ein anderes, um dessen Farbe zu übernehmen.

Die Dropdown-Farbauswahl

Klicken Sie auf den Dropdown-Pfeil rechts neben der Farbschaltfläche , um ein Fenster für die schnelle Farbauswahl anzuzeigen. Dieser Shortcut bietet Zugriff auf:

- ein Farbrad zur Auswahl einer Farbe
- einen Schieberegler zum Ändern der Farbdeckkraft
- die durch *In Farbknöpfen anzeigen* eingestellten Farbpaletten
- Kopieren und Einfügen (in einem anderen Fenster) der aktuellen Farbe
- Wählen einer beliebigen Farbe auf dem Bildschirm
- Wählen einer Farbe über den Farbe-auswählen-Dialog
- per Drag-and-Drop eine Farbe von einem Widget zum anderen ziehen

Tipp

Bewegen Sie das Mausrad über ein Farbauswahl-Widget, um die Deckkraft der zugehörigen Farbe schnell zu ändern.

Bemerkung

Wenn das Farb-Widget durch die daten-definierten Überschreibungseigenschaften auf *Projektfarbe* gesetzt ist, sind die oben genannten Funktionen zum Ändern der Farbe nicht verfügbar. Sie müssen zuerst die Definition mit *Farbe entkoppeln* oder *Löschen* ändern.

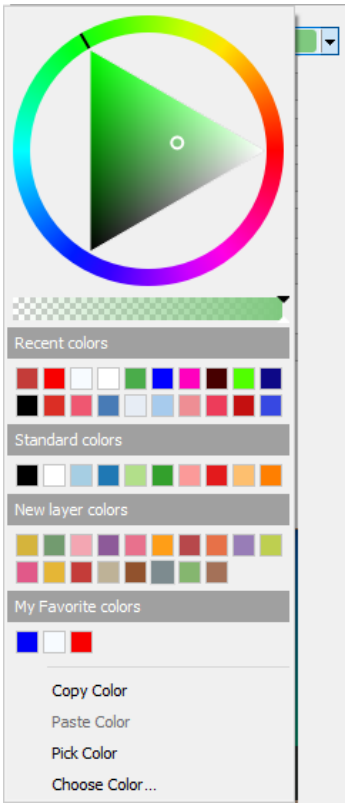


Abb. 8.19: Das Dropdown-Farbauswahl-Fenster

Die Dropdown- Schaltfläche für Farbverläufe

Farbverläufe sind eine praktische Möglichkeit, eine Reihe von Farben auf ein oder mehrere Objekte anzuwenden. Ihre Erstellung wird im Abschnitt *Einstellen einer Farbrampe* beschrieben. Wie bei den Farben öffnet das Drücken der Schaltfläche Farbverlauf den entsprechenden Farbverlauf -Dialog, in dem Sie die Eigenschaften ändern können.

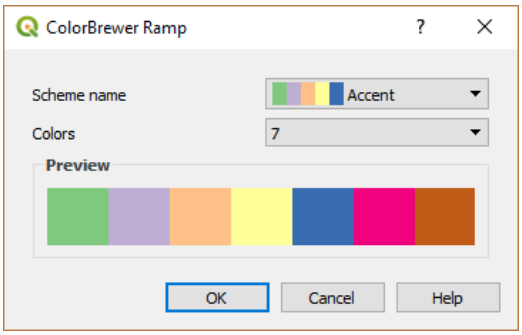


Abb. 8.20: Anpassen eines ColorBrewer-Farbverlaufs

Das Dropdown-Menü rechts neben der Schaltfläche bietet schnellen Zugriff auf eine größere Anzahl von Farbverläufen und Optionen:

- *Farbverlauf invertieren*
- *Akutellen Farbverlauf zurücksetzten* um einen dem Widget zugewiesene Farbverlauf zurückzusetzen (nur in bestimmten Kontexten verfügbar)
- *Zufälliger Farbverlauf*: Nur in bestimmten Kontexten verfügbar (z. B. wenn ein Farbverlauf für eine Layer-Symbologie verwendet wird), wird durch Aktivieren dieses Eintrags ein Farbverlauf mit Zufallsfarben erstellt und angewendet. Außerdem kann mit dem Eintrag *Zufallsfarben mischen* ein neuer zufälliger Farbverlauf erzeugt werden, wenn der aktuelle nicht zufriedenstellend ist.
- eine Vorschau der als **Favoriten** gekennzeichneten Farbverläufe im *Stil-Manager*-Dialog
- *Alle Farbverläufe* zum Zugriff auf die Datenbank der kompatiblen Farbverläufe
- *Neuen Farbverlauf erstellen…* eines beliebigen unterstützten Typs, der im aktuellen Widget verwendet werden kann (beachten Sie, dass dieser Farbverlauf nur dann an anderer Stelle verfügbar ist, wenn Sie ihn in der Bibliothek speichern)
- *Farbverlauf bearbeiten…*, dasselbe wie ein Klick auf die Schaltfläche für den gesamten Farbverlauf
- *Farbverlauf speichern…*, um den aktuellen Farbverlauf mit seinen Anpassungen in der Stilbibliothek zu speichern

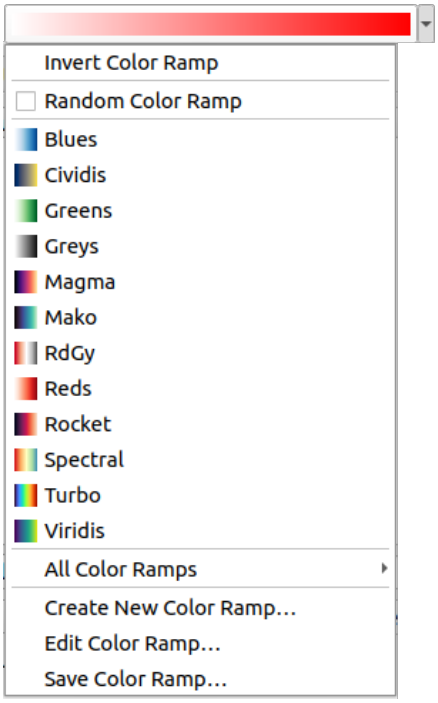


Abb. 8.21: Fenster zur schnellen Auswahl des Farbverlaufs

8.8.2 Symbol-Widget

Die Symbolbearbeitung bietet eine schnelle Möglichkeit, die Symboleigenschaften eines Objekts festzulegen. Sie finden die Symbolbearbeitung, wenn Sie im Kontextmenü eines Layers auf *Stile* gehen am unteren Ende des Widgets. Die folgenden Optionen stehen zur Verfügung:

- *Symbol bearbeiten…*: ist dasselbe wie das Drücken des Symbolauswahl-Widgets. Es öffnet sich ein Dialog zum Einstellen der *Symbolparameter*.
- *Symbol kopieren* des ausgewählten Objekts
- *Symbol einfügen* im ausgewählten Objekt
- *Symbol zurücksetzten* um jedes dem Widget zugewiesene Symbol wieder zu entfernen (nur in bestimmten Kontexten verfügbar)

Tipp

Bewegen Sie das Mausrad über ein Marker- oder Liniensymbol-Widget, um die Größe des zugehörigen Symbols schnell zu ändern.

8.8.3 Entfernte oder eingebettete Dateiauswahl

Zusammen mit dem Dateiselektor-Widget zeigt die Schaltfläche … manchmal einen Dropdown-Pfeil. Dieser ist in der Regel verfügbar, wenn Sie:

- eine SVG-Grafik in einem Symbol oder einer Beschriftung verwenden
- ein Rasterbild zum Anpassen von Symbolen, Beschriftungen, Texturen oder Dekorationen verwenden

Wenn Sie auf den Pfeil drücken, wird ein Menü angezeigt:

- *Datei auswählen…*, um die Datei aus dem Dateisystem zu laden. Die Datei wird durch den Dateipfad identifiziert und QGIS muss den Pfad auflösen, um das entsprechende Bild anzeigen zu können
- *Von URL…*, um die Datei von einer Internetadresse zu laden. Wie oben, wird das Bild nur bei erfolgreichem Abruf der Internet-Ressource geladen
- *Datei einbetten…*, um die Datei in das aktuelle Projekt, die Stildatenbank oder die Drucklayoutvorlage einzubetten. Die Datei wird dann immer als Teil des Objekts angezeigt. Dies ist ein bequemer Weg, um in sich geschlossene Projekte mit benutzerdefinierten Symbolen zu erstellen, die leicht von verschiedenen Benutzern und Installationen von QGIS gemeinsam genutzt werden können.
- *Eingebettete Datei extrahieren…*, um die eingebettete Datei aus dem Widget zu extrahieren und auf der Festplatte zu speichern.

8.8.4 Einstellen des Sichtbarkeitsmaßstabs

Mit der Auswahl des Sichtbarkeitsmaßstabs können Sie festlegen, in welchem Maßstab ein Element in der Kartenansicht sichtbar gemacht werden soll. Außerhalb des angegebenen Maßstabsbereichs werden die Elemente nicht angezeigt. Er kann z. B. auf Layer, Beschriftungen oder Diagramme über den Reiter *Darstellung* angepasst werden.

1. Aktivieren Sie das Kontrollkästchen *Maßstabsabhängige Sichtbarkeit*.
2. Füllen Sie das Feld *Minimum (exklusiv)* mit dem am weitesten herausgezoomten gewünschten Maßstab, indem Sie den Wert eingeben oder aus der Liste *Vordefinierte Maßstäbe* auswählen.
3. und/oder füllen Sie das Feld *Maximum (einschließlich)* mit dem am stärksten vergrößerten gewünschten Maßstab

   Die Schaltfläche Auf aktuellen Kartenmaßstab setzen neben den Maßstabsfeldern setzt den aktuellen Kartenansicht-Maßstab als Grenze des Sichtbarkeitsbereichs. Drücken Sie den Pfeil neben der Schaltfläche, um auf Maßstäbe aus den Karten der Layouts zuzugreifen und sie zum Ausfüllen des Feldes zu verwenden.

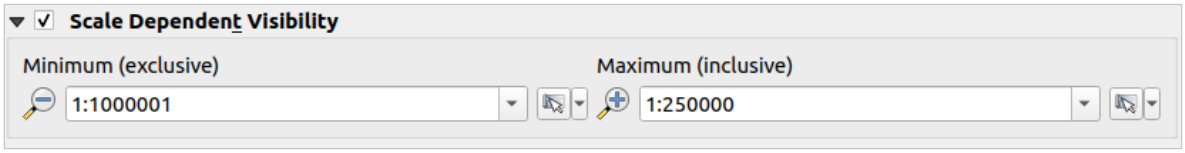


Abb. 8.22: Sichtbarkeitsmaßstab-Auswahl-Widget

8.8.5 Auswahl der räumlichen Ausdehnung

Das Auswahl-Werkzeug *Ausdehnung* ist eine bequeme Abkürzung, wenn Sie einem Layer eine räumliche Ausdehnung auswählen wollen oder den Bereich auszuführender Aktionen einschränken wollen. Je nach Kontext bietet es eine Auswahl zwischen:

- *Aktuelle Layer-Ausdehnung*: z.B. beim Exportieren eines Layers
- *Aus Layer berechnen* ►: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers
- Aktuelle *Kartenansicht Ausdehnung* verwenden
- *Zeichnen auf Kartenansicht*: ein Rechteck, dessen Koordinaten dann verwendet werden
- *Berechnen aus Lesezeichen*: verwendet die Ausdehnung eines gespeicherten *Lesezeichens*
- *Berechnen aus Layout*: verwendet den Bereich eines Drucklayouts
- Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein oder bearbeiten Sie diese

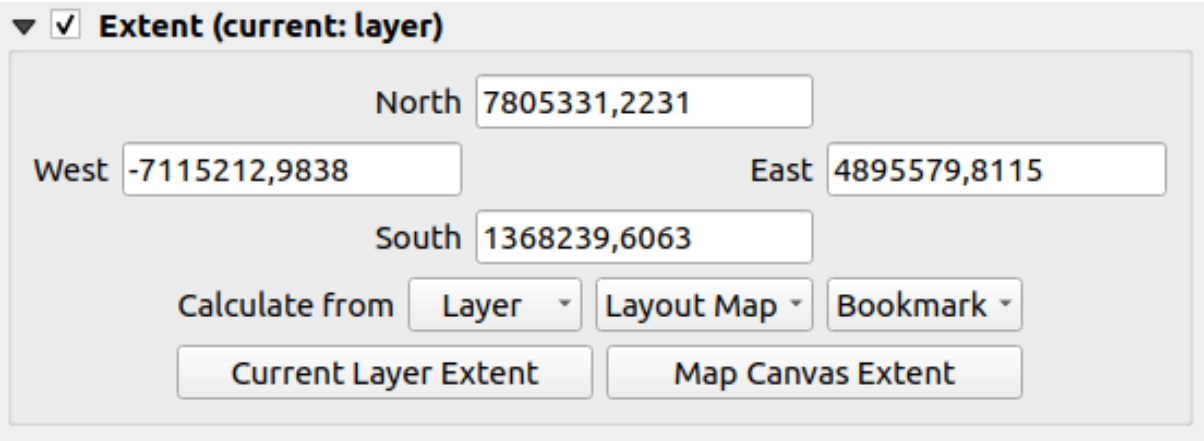


Abb. 8.23: Das Ausdehung-Auswahl Widget

8.8.6 Schrift Auswahl

Das *Schrift* Auswahl-Werkzeug bietet verschiedene Optionen für die Festlegung von Schrifteigenschaften in den verschiedensten Bereichen (Objektbeschriftungen, Dekorationsbeschriftungen, Kartenlegende, …). Wenn Sie auf den Dropdown-Pfeil klicken, werden, je nach Kontext, einige oder alle der folgenden Optionen angezeigt:

Abb. 8.24: Schriftauswahl-Werkzeug

- *Aktuelles Textformat zurücksetzen*, um das dem Widget zugewiesene Textformat zurückzusetzen (nur in bestimmten Kontexten verfügbar)
- *Schriftgröße* in der zugehörigen Einheit
- *Kürzlich verwendete Schriften* ► Menü mit der aktiven Schriftart markiert (ganz oben)
- *Format konfigurieren…*: dasselbe wie das Drücken des Schriftauswahl-Widgets. Es öffnet den Dialog *Textformat*, der erweiterte Formatierungsoptionen wie Farbe, Deckkraft, Ausrichtung, HTML-Notation, Puffer, Hintergrund, Schatten, … bietet.
- *Format kopieren* des aktuellen Textes
- *Format einfügen* in den aktuellen Text
- die *Farb-Auswahl* zum Einstellen der Schriftfarbe

Tipp

Bewegen Sie das Mausrad über ein Schriftauswahl-Widget, um die Schriftgröße des zugehörigen Textes schnell zu ändern.

8.8.7 Einheiten Auswahl

Die Größeneigenschaften der Elemente (Beschriftungen, Symbole, Layoutelemente, …) in QGIS sind nicht unbedingt an die Projekteinheiten oder die Einheiten eines bestimmten Layers gebunden. Für eine große Anzahl von Eigenschaften erlaubt das Dropdown-Menü *Einheit* die Anpassung der Werte an die gewünschte Darstellung (basierend auf der Bildschirmauflösung, der Blattgröße oder dem Terrain). Verfügbare Einheiten sind:

- *Millimeter*
- *Punkte*
- *Pixel*

- *Inch*
- *Prozentsatz*: ermöglicht es Ihnen, bestimmte Eigenschaften als Prozentsatz einer anderen Eigenschaft festzulegen. Dies ist zum Beispiel nützlich für die Erstellung von Textformaten, bei denen die einzelnen Bestandteile (Puffergröße, Schattenradius…) bei Änderung der Textgröße skaliert werden, anstatt konstante Größen zu haben. So müssen Sie diese Größen nicht anpassen, wenn sich die Textgröße ändert.
- *Meter im Maßstab*: Damit können Sie die Größe immer in Metern angeben, unabhängig davon, welche Karteneinheiten zugrunde liegen (z.B. Zoll, Fuß, geografische Grade, …). Die Größe in Metern wird auf der Grundlage der aktuellen Projekt-Ellipsoid-Einstellung und einer Projektion der Entfernungen in Metern in der Mitte der aktuellen Kartenausdehnung berechnet. Bei Karten in einem projizierten Koordinatensystem wird dies anhand der projizierten Einheiten berechnet. Bei Karten in einem geografischen (Breiten-/Längen-) System wird die Größe näherungsweise durch Berechnung der Metergrößen unter Verwendung von Ellipsoidberechnungen für den vertikalen Maßstab der Karte ermittelt.
- *Karteneinheiten*: Die Größe wird entsprechend dem Maßstab der Kartenansicht skaliert. Da dies zu sehr großen oder sehr kleinen Werten führen kann, können Sie mit der Schaltfläche **|Optionen|** neben dem Eintrag, die Größe auf einen Bereich von Werten beschränken:
  - *Minimaler Maßstab* und *Maximaler Maßstab*: Der Wert wird entsprechend dem Maßstab der Kartenansicht skaliert, bis eine dieser Maßstabsgrenzen erreicht wird. Außerhalb dieses Bereichs wird der Wert der nächstgelegenen Maßstabsgrenze beibehalten.
  - und/oder *Minimale Größe* und *Maximale Größe* in `mm`: Der Wert wird entsprechend dem Maßstab der Kartenansicht skaliert, bis er einen dieser Grenzwerte erreicht; dann wird die Grenzgröße beibehalten.

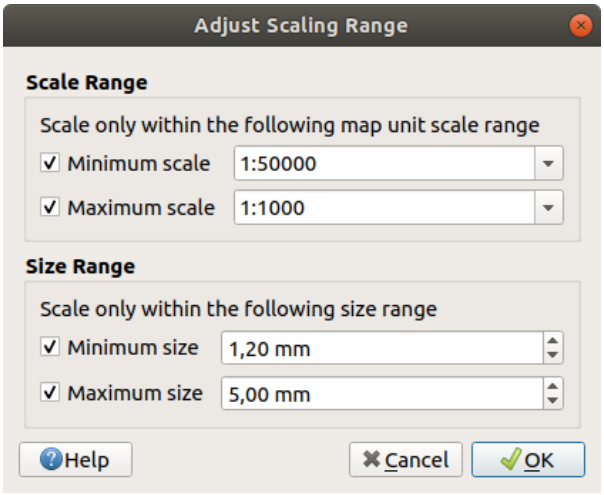


Abb. 8.25: Dialog zur Anpassung des Maßstabsbereichs

8.8.8 Zahlen Formatierung

Zahlenformate ermöglichen die Formatierung von Zahlenwerten für die Anzeige unter Verwendung einer Vielzahl verschiedener Formatierungstechniken (z. B. wissenschaftliche Notation, Währungswerte, Prozentwerte, benutzerdefinierte Formatierung usw.). Eine Anwendungsmöglichkeit hierfür ist die Darstellung von Text in einer Layout-Maßstabsleiste oder -Tabelle, einer Farbregisterlegende, einem Höhenprofil-Diagramm usw.

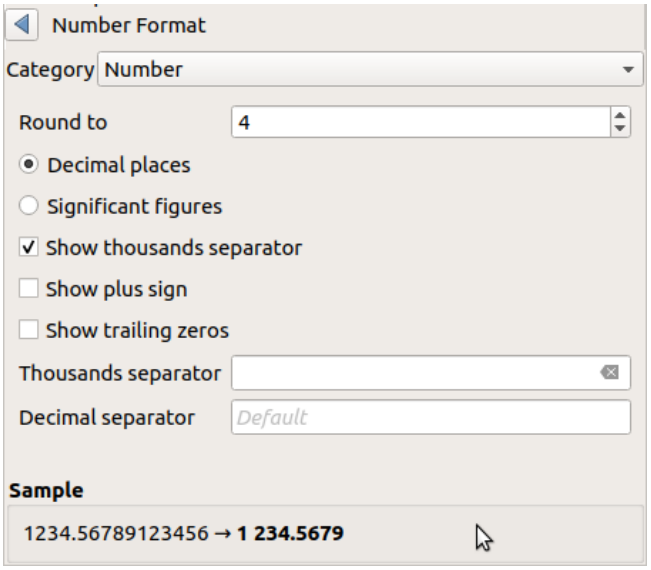


Abb. 8.26: Formatierung numerischer Werte

Es werden verschiedene Kategorien von Formaten unterstützt. Für die meisten von ihnen können Sie einen Teil oder alle der folgenden numerischen Optionen einstellen:

- *Tausender-Trennzeichen anzeigen*
- *Pluszeichen anzeigen*
- *Nullen am Ende anzeigen*

Sie können aber auch ihre eigenen Einstellungen haben. Bereitgestellte Kategorien sind:

- *Allgemein*, die Standardkategorie: hat keine Einstellung und zeigt die Werte an, die in den Eigenschaften des übergeordneten Widgets oder in den globalen Einstellungen festgelegt wurden.
- *Benutzerdefinierter Ausdruck*: Ermöglicht Ihnen die Erstellung eines benutzerdefinierten *QGIS-Ausdrucks* zur Formatierung von Zahlen. Der Ausdruck kann die Variable `@value` verwenden, um den zu formatierenden Wert abzurufen, und dann eine beliebige Standard-QGIS-Ausdrucksfunktion verwenden, um diesen nach Wunsch zu formatieren.
- *Zahl*
  - Der Wert kann *Gerundet auf* eine festgelegte Anzahl von *Dezimalstellen* oder *Signifikante Stellen* sein
  - Anpassen des *Tausender-Trennzeichen* und des *Dezimaltrennzeichen*
- *Kurs* für eine Textdarstellung einer Richtung/eines Kurses mit:
  - *Format*: mögliche Wertebereiche sind `0 bis 180°, mit Suffix O/W`, `−180 bis +180°` und `0 bis 360°`
  - Anzahl von *Dezimalstellen*
- *Währung* für eine Textdarstellung eines Währungswertes.
  - *Prefix*

  - *Suffix*
  - Anzahl von *Dezimalstellen*
- *Bruch* für eine Bruchdarstellung eines Dezimalwerts (z. B. *1/2* statt *0,5*)
  - ☐ *Unicode super/subscript-Darstellung* zum Beispiel $^{1}/_{2}$ anstelle von 1/2
  - ☐ *Spezielle Unicode-Zeichen verwenden*
  - *Tausender-Trennzeichen* anpassen
- *Prozentsatz* - hängt `%` an die Werte an, mit der Einstellung von:
  - Anzahl von *Dezimalstellen*
  - *Skalierung* um anzugeben, ob die aktuellen Werte bereits Prozente (dann werden sie beibehalten) oder Brüche (dann werden sie umgewandelt) darstellen
- *Wissenschaftliche* Notation in der Form `2.56e+03`. Die Anzahl der *Dezimalstellen* kann eingestellt werden.

Eine Live-Vorschau der Einstellungen wird unter dem Abschnitt *Beispiel* angezeigt.

8.8.9 Mischmodi

QGIS bietet mit diesen Werkzeugen verschiedene Möglichkeiten für spezielle Darstellungseffekte, die Sie vielleicht bisher nur aus Grafikprogrammen kennen. Mischmodi können auf Layer und Objekte, aber auch auf Drucklayoutelemente angewendet werden:

- *Normal*: Dies ist der Standardüberblendungsmodus, der den Alphakanal des obersten Pixels verwendet, um mit dem darunter liegenden Pixel zu mischen. Die Farben werden nicht gemischt.
- *Heller*: Dabei wird das Maximum jeder Komponente aus den Vordergrund- und Hintergrundpixeln ausgewählt. Beachten Sie, dass die Ergebnisse oft zackig und hart sind.
- *Bildschirm*: Helle Pixel aus der Quelle werden über das Ziel gemalt, dunkle Pixel dagegen nicht. Dieser Modus ist am nützlichsten, um die Textur eines Elements mit einem anderen Element zu mischen (z. B. Verwendung eines Hügelschattens zur Texturierung eines anderen Layers).
- *Abwedeln*: Hellt die darunter liegenden Pixel auf und sättigt sie, basierend auf der Helligkeit des oberen Pixels. Je heller die obersten Pixel sind, desto stärker werden Sättigung und Helligkeit der darunter liegenden Pixel. Dies funktioniert am besten, wenn die oberen Pixel nicht zu hell sind. Andernfalls ist der Effekt zu extrem.
- :guilabel:`Addition“: Addiert die Pixelwerte des einen Elements zum anderen. Bei Werten, die über dem Maximalwert liegen (im Falle von RGB), wird Weiß angezeigt. Dieser Modus eignet sich zum Hervorheben von Objekten.
- *Dunkler* Behält die niedrigsten Werte der einzelnen Komponenten der Vordergrund- und Hintergrundpixel bei. Wie beim Aufhellen sind die Ergebnisse eher zackig und hart.
- *Multiplizieren*: Die Pixelwerte des oberen Elements werden mit den entsprechenden Werten des unteren Elements multipliziert. Die Ergebnisse sind dunkler.
- *Einbrennen*: Dunklere Farben im obersten Element bewirken, dass die darunter liegenden Elemente dunkler werden. Mit „Einbrennen“ können Sie die darunter liegenden Layer optimieren und einfärben.
- *Overlay*: Combines multiply and screen blending modes. Light parts become lighter and dark parts become darker.
- *Weiches Licht*: Ähnlich wie Überlagern, aber anstelle von Multiplizieren/ Bildschirm verwendet die Funktion Einbrennen/Abwedeln. Dies soll den Schein eines weichen Lichts auf das Bild simulieren.
- *Hartes Licht*: Hartes Licht ist dem Überlagerungs-Modus sehr ähnlich. Er soll die Projektion eines sehr intensiven Lichts auf ein Bild simulieren.
- *Unterschied*: Subtrahiert das oberste Pixel vom untersten Pixel, oder umgekehrt, um immer einen positiven Wert zu erhalten. Das Überblenden mit Schwarz ergibt keine Veränderung, da die Differenz mit allen Farben gleich Null ist.

- *Abziehen*: Subtrahiert die Pixelwerte des einen Elements vom anderen. Im Falle negativer Werte wird Schwarz angezeigt.

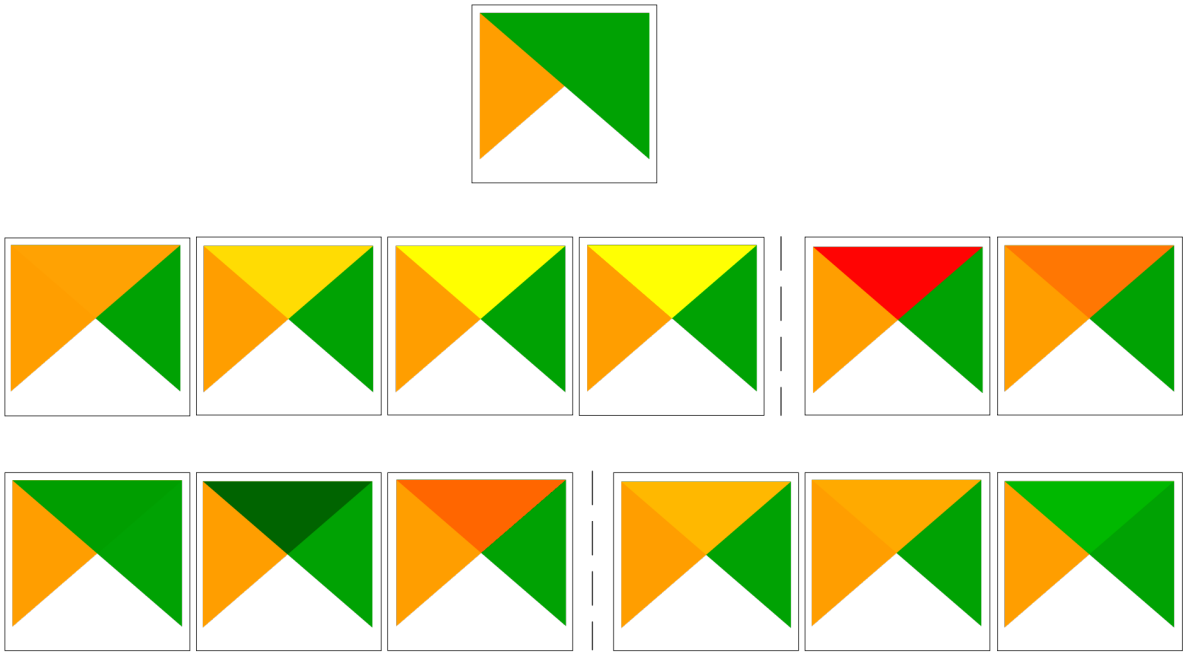


Abb. 8.27: Beispiele für Mischmodi, die auf ein grünes Objekt über einem orangefarbenen Objekt angewendet werden
Von oben nach unten, von links nach rechts: Normal – Heller, Bildschirm, Abwedeln, Addition – Unterschied, Abziehen–Dunkler, Multiplizieren, Einbrennen – Überlagern, Weiches Licht, Hartes Licht

Wenn ein Layer Teil einer Gruppe ist, für die *Layer als Gruppe zeichnen* aktiviert ist, stehen zusätzliche Blending-Modi für die Darstellung zur Verfügung. Sie bieten Methoden, um die Darstellung des Inhalts eines Layers durch den Inhalt eines zweiten „Masken“-Layers zu überlagern.

- *Durch darunter maskiert*: Die Ausgabe ist das obere Pixel, dessen Deckkraft um die des unteren Pixels reduziert ist.
- *Darunter maskieren*: Die Ausgabe ist das untere Pixel, dessen Deckkraft um die des oberen Pixels reduziert ist.
- *Invers durch darunter maskiert*: Die Ausgabe ist das oberste Pixel, bei dem die Deckkraft um den Kehrwert des unteren Pixels reduziert wird.
- *Darunter invers maskiere*: Die Ausgabe ist das untere Pixel, bei dem die Deckkraft um den Kehrwert des oberen Pixels reduziert ist.
- *In darunter zeichnen*: Das obere Pixel wird über das untere Pixel gemischt, wobei die Deckkraft des oberen Pixels um die Deckkraft des unteren Pixels verringert wird.
- *Darunter darin zeichnen*: Das untere Pixel wird über das obere Pixel gemischt, wobei die Deckkraft des unteren Pixels um die Deckkraft des oberen Pixels verringert wird.

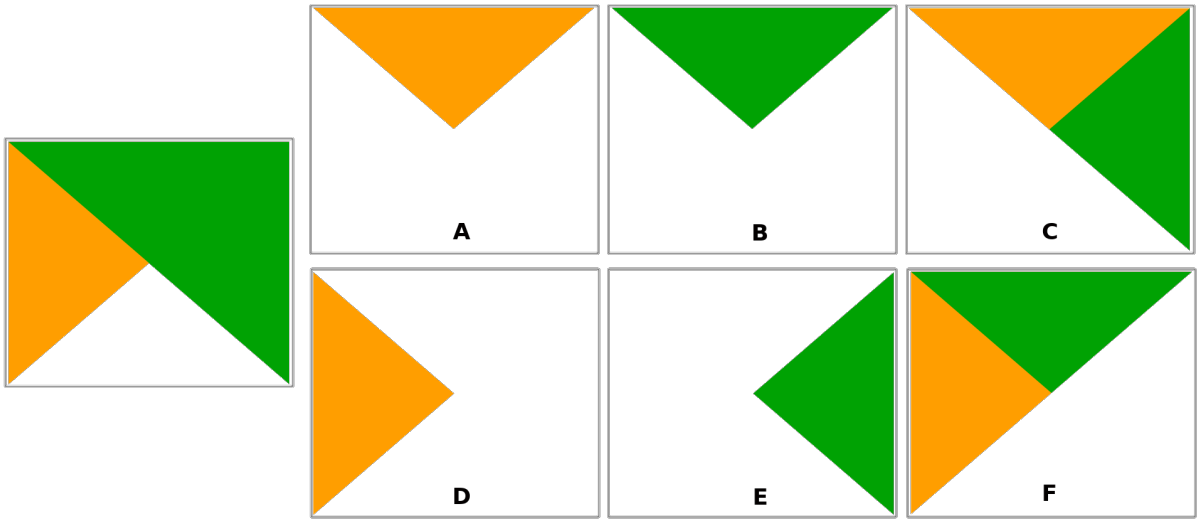


Abb. 8.28: Beispiele für Mischmodi, der auf den obersten grünen Layer in einer Gruppe angewendet wird
A: Darunter maskieren**B**: Durch darunter maskiert**C**: Darunter darin zeichnen**D**: Darunter invers maskiere**E**: Invers durch darunter maskiert**F**: In darunter zeichnen

8.8.10 Einrichten von datendefinierter Übersteuerung

Neben vielen Optionen im Layereigenschaften-Dialog oder Einstellungen im Drucklayout finden Sie das Symbol Datendefinierte Übersteuerung. Mithilfe von *Ausdrücken*, die auf Layer-Attributen oder Einstellungen basieren, vorgegebenen oder benutzerdefinierten Funktionen und *Variablen* können Sie mit diesem Werkzeug dynamische Werte für Parameter festlegen. Wenn die datendefinierte Übersteuerung aktiviert ist, wird der zurückgegebene Wert auf den Parameter angewendet, unabhängig von seinen eigentlichen Einstellungen (Kontrollkästchen, Textfeld, Schieberegler…).

Das Werkzeug für die datendefinierte Übersteuerung

Wenn Sie auf das Symbol Datendefinierte Übersteuerung klicken, werden die folgenden Einträge angezeigt:

- *Beschreibung…* die anzeigt, ob die Option aktiviert ist, welche Eingabe erwartet wird, den gültigen Eingabetyp und die aktuelle Definition. Wenn Sie den Mauszeiger über das Symbol bewegen, werden diese Informationen ebenfalls angezeigt.
- *Daten im Projekt speichern*: eine Option, mit der die Einstellungen unter Verwendung des Mechanismus *Hilfsspeicher* gespeichert werden kann.
- *Feldtyp*: ein Eintrag zur Auswahl aus den Feldern des Layers, die dem gültigen Eingabetyp entsprechen.
- *Farbe*: wenn das Widget mit einer Farbeigenschaft verknüpft ist, gibt dieses Menü Zugriff auf die Farben, die als Teil der aktuellen *Projektfarben* definiert sind.
- *Variable*: ein Menü für den Zugriff auf die verfügbaren benutzerdefinierten *Variablen*
- *Bearbeiten…* um einen Ausdruck zu erstellen oder zu bearbeiten, unter Verwendung des *Ausdrucks-Generator* Dialogs. Um Ihnen beim korrekten Erstellen des Ausdrucks zu helfen, wird im Dialogfeld eine Erinnerung an das erwartete Format der Ausgabe angezeigt.
- *Einfügen* und *Kopieren* Schaltflächen.
- *Löschen* Schaltfläche, um die Einstellung zu entfernen.

- Für numerische und farbliche Eigenschaften steht ein *Assistent…* zur Skalierung zur Verfügung (mehr Details *unten*)

> **Tipp**
>
> **Mit der rechten Maustaste können Sie die Datenüberschreibung (de)aktivieren**
>
> Wenn die datendefinierte Überschreibung korrekt eingerichtet ist, ist das Symbol gelb oder . Ist sie nicht korrekt, ist das Symbol rot oder .
>
> Sie können eine korrekt konfigurierte Datendefinierte Übersteuerung aktivieren oder deaktivieren, indem Sie einfach mit der rechten Maustaste auf das Symbol klicken.

> **Tipp**
>
> **Verwenden Sie die mittlere Maustaste, um den anzuwendenden Ausdruck zu erstellen oder zu bearbeiten**
>
> Sie können den Dialog *Ausdruckseditor* direkt öffnen, um den anzuwendenden Ausdruck zu erstellen oder zu bearbeiten, indem Sie einfach mit der mittleren Maustaste auf das Widget Datendefinierte Übersteuerung klicken.

Verwendung des Assistenten für die datendefinierte Übersteuerung

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.:
  - *Quelle*: das darzustellende Attribut, festgelegt durch ein Feld oder einen *Ausdruck*
  - 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.
- Definieren Sie die *Output*-Werte: Die Optionen variieren je nach dem zu definierenden Parameter. Sie können global festgelegt werden:
  - 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.

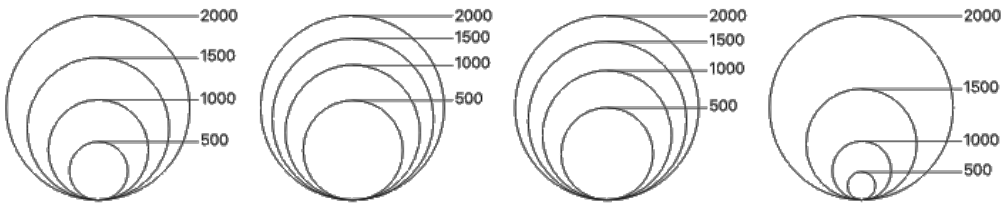


Abb. 8.29: Scaling values using different methods (from left to right): Radius - Surface - Flannery - Exponential of 2

Wenn es mit der Eigenschaft kompatibel ist, wird auf der rechten Seite des Dialogs eine Live-Update-Vorschau angezeigt, mit deren Hilfe Sie die Werteskalierung kontrollieren können.

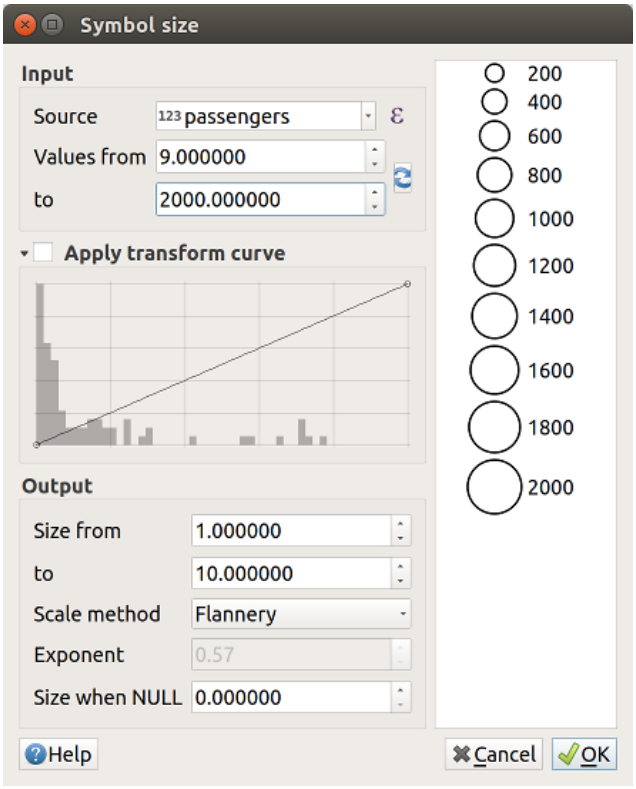


Abb. 8.30: Skalierung der Objektgröße basierend auf dem Wert des Passagier-Felds

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 Authentifizierung

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.

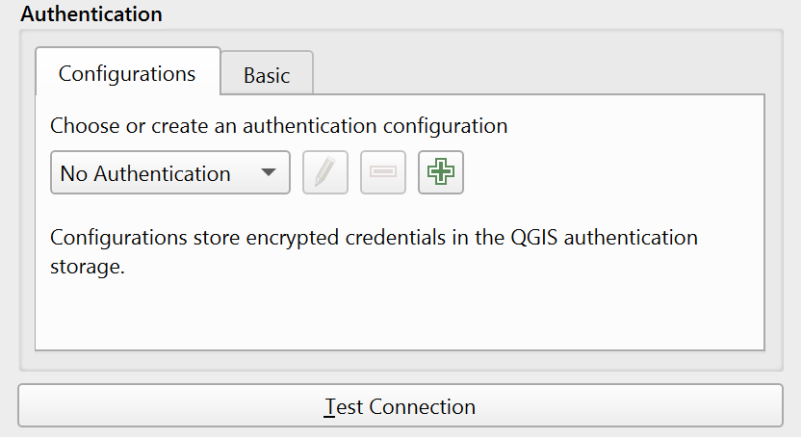


Abb. 8.31: Authentication widget

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.

> **Warnung**
>
> **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 *Authentifizierungssystem* 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 *Authentifizierungssystem*.

KAPITEL 9

Level up mit Ausdrücken

9.1 Ausdrücke

Basierend auf Layer-Daten und vorgegebenen oder benutzerdefinierten Funktionen bieten **Ausdrücke** eine leistungsstarke Möglichkeit zur Bearbeitung von Attributwerten, Geometrie und Variablen, um den Geometriestil, den Inhalt oder die Position der Beschriftung, den Wert für Diagramme, die Höhe eines Layout-Elements dynamisch zu ändern, einige Objekte auszuwählen, virtuelle Felder zu erstellen, …

Bemerkung

Eine Liste der Standardfunktionen und -variablen zum Erstellen von Ausdrücken, mit detaillierten Informationen und Beispielen, finden Sie unter *Liste der Funktionen*.

9.1.1 Der Ausdruckseditor

Der Hauptdialog zur Erstellung von Ausdrücken, der *Ausdruckseditor* ist an vielen Stellen in QGIS verfügbar. Er kann z.B. hier aufgerufen werden:

- durch Klick auf den ε Knopf;
- *Auswahl von Objekten* mit Hilfe des Tools Objekte über Ausdruck wählen;
- *Attribute bearbeiten* z.B. mit Hilfe des Tools Feldrechner;
- Änderung der Symbolisierung, der Beschriftung oder der Parameter von Layoutelementen mit Hilfe des Tools Datendefinierte Übersteuerung (see *Einrichten von datendefinierter Übersteuerung*);
- Erstellung eines Layers mit Symbolisierung mit Hilfe des *Geometriegenerators*;
- Ausführung einer *Geoprozessierung*.

Der Ausdruckseditor erlaubt den Zugang zu:

- Reiter *Ausdruck*, der dank einer Liste von *vordefinierten Funktionen* dabei hilft, den zu verwendenden Ausdruck zu schreiben und zu überprüfen;
- Reiter *Funktionseditor*, der bei der Erstellung angepasster Funktionen und damit der Erweiterung der Liste der Funktionen unterstützt.

Die Benutzeroberfläche

Der Reiter *Ausdruck* bietet die hauptsächliche Schnittstelle zum Schreiben von Ausdrücken mit Funktionen, Layer-Feldern und Werten. Er enthält die folgenden Elemente (Widgets):

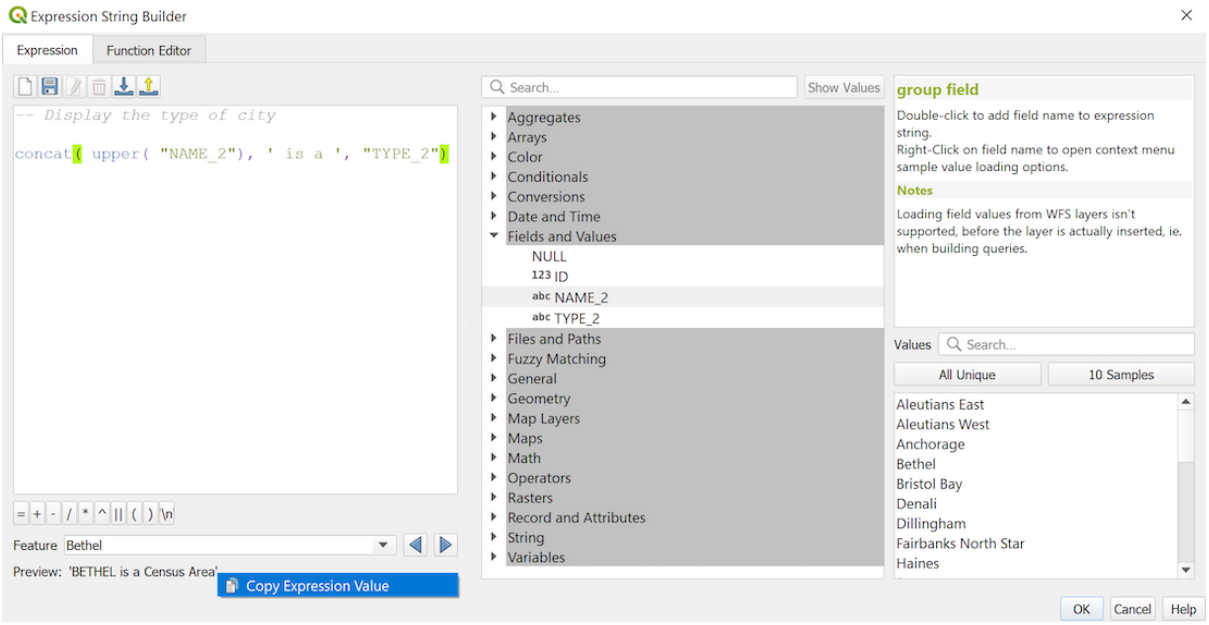


Abb. 9.1: Der Ausdruck-Reiter

- Ein Editorbereich für Ausdrücke zum Eingeben oder Einfügen von Ausdrücken. Autovervollständigung ist verfügbar, um das Schreiben von Ausdrücken zu beschleunigen:
  - Die zum Eingabetext passenden Variablen, Funktionsnamen und Feldnamen werden unterhalb der Eingabe angezeigt. Nutzen Sie die `Up` und `Down` Tasten, um die Einträge durchzublättern und die Taste `Tab`, um einen Eintrag in den Ausdruck zu übernehmen oder klicken Sie einfach doppelt auf den gewünschten Eintrag.
  - Die Parameter von Funktionen werden beim Ausfüllen angezeigt.

  QGIS prüft die Richtigkeit des Ausdrucks und hinterlegt alle Fehler farbig:
  - *Unterstreichung*: unbekannte Funktionen, falsche oder ungültige Funktionsargumente;
  - *Markierung*: alle weiteren Fehler (z.B. fehlende Klammern, unerwartete Zeichen) an der jeweiligen Stelle.

> **Tipp**
>
> **Dokumentieren Sie Ihre Ausdrücke mit Kommentaren**
>
> Bei Nutzung komplexer Ausdrücke, ist es eine gute Praxis mehrzeilige Kommentare bzw. im Ausdruck eingeschlossene Kommentare als Erinnerungshilfen zu verwenden.

```
/*
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
)
```

- Oberhalb des Ausdrucksеditors finden Sie eine Reihe von Werkzeugen, die Ihnen helfen:
  - Ausdrucksbearbeitung leeren
  - Erstellen und Bearbeiten von *Benutzerausdrücken*
- Unter dem Ausdrucksеditor finden Sie:
  - Eine Reihe von grundlegenden Operatoren, die Ihnen bei der Erstellung des Ausdrucks helfen
  - Ein Hinweis auf das erwartete Ausgabeformat bei der Datendefinierten Übersteuerung von Objekteigenschaften
  - Eine Live *Vorschau der Ausgabe* des Ausdrucks (bis zu 60 Zeichen), die standardmäßig für das erste Objekt des Layers berechnet wird. Um eine Ausgabevorschau mit mehr als 60 Zeichen anzuzeigen, können Sie mit dem Mauszeiger über den Text fahren, um ein Tooltip-Popup mit der gesamten Ausgabevorschau anzuzeigen. Um den Text der Ausgabevorschau in die Zwischenablage zu kopieren, klicken Sie mit der rechten Maustaste auf den Text der Ausgabevorschau und wählen *Ausdruckswert kopieren*.

    Mit der Combobox *Objekt* können Sie weitere Objekte des Layers durchsuchen und berechnen (die Werte werden aus der Eigenschaft *Anzeigename* des Layers übernommen).

    Im Falle eines Fehlers wird dieser angezeigt und Sie können die Details über den angegebenen Hyperlink aufrufen.
- In einem Auswahlfeld sind die Funktionen, Variablen, Felder, … in Gruppen organisiert aufgelistet. Ein Suchfeld dient zur Filterung der Gesamtliste, um schnell eine bestimmte Funktion oder Feld zu finden. Mit Hilfe eines Doppelklicks wird das Element in den Ausdrucksеditor eingefügt.
- Ein Bedienfeld, das Hilfe für jedes ausgewählte Element aus der Funktionsauwahl anzeigt.

> **Tipp**
>
> Drücke `Ctrl+Click` beim Überfahren eines Funktionsnamens, um automatisch die Hilfe zu dieser Funktion im Hilfe-Dialog anzuzeigen.

Wenn ein Feld im Funktionsselektor ausgewählt wird, können Sie sich im unteren, rechten Bereiche die Werte des Feldes auf verschiedene Weise anzeigen lassen:

- im *Werte* Suchfeld nach einem bestimmten Feldwert suchen

– Zeigt die Liste der Werte von *Alle eindeutigen* oder *10 Stichproben* an; auch über die rechte Maustaste verfügbar.

  Wenn das Feld mit einem anderen Layer oder einer Gruppe von Werten verknüpft ist, d.h. wenn die *Feldeigenschaften* entsprechend gesetzt sind, ist es möglich, alle Werte des verknüpften Feldes aufzulisten (aus dem referenzierten Layer, der Tabelle oder der Liste). Darüber hinaus können Sie diese Liste nach *Nur verwendete Werte anzeigen* im aktuellen Feld filtern.

Ein Doppelklick auf einen Feldwert im Widget fügt ihn dem Ausdrucksеditor hinzu.

Tipp

Das rechte Bedienfeld mit der Hilfe zu Funktionen und den Feldwerten kann ausgeblendet werden (durch nach rechts ziehen der Bedienfeldbegrenzung). Durch Drücken des Knopfes *Werte anzeigen* oder *Hilfe anzeigen* wird er wieder angezeigt.

Einen Ausdruck schreiben

QGIS-Ausdrücke werden verwendet, um Objekte auszuwählen oder Werte zu setzen. Das Schreiben eines Ausdrucks in QGIS folgt einigen Regeln:

1. **Der Dialog definiert den Kontext**: wenn Sie mit SQL vertraut sind, kennen Sie wahrscheinlich Abfragen des Typs *select features from layer where condition* oder *update layer set field = new_value where condition*. Ein QGIS-Ausdruck benötigt ebenfalls alle diese Informationen, aber das Werkzeug, mit dem Sie den Ausdrucksdialog öffnen, liefert einen Teil davon. Zum Beispiel, wenn man einen Layer (`buildings`) mit einem Feld (`height`) hat:

   - Wenn Sie das Werkzeug Objekte nach Ausdruck wählen drücken, bedeutet das, dass Sie „bestimmte Objekte aus `buildings``auswählen" wollen. Die **Bedingung** ist die einzige Information, die Sie in das Ausdruckstextwidget eingeben müssen, z. B. geben Sie ``"height" > 20` ein, um Gebäude auszuwählen, die höher als 20 sind.

   - Wenn Sie diese Auswahl getroffen haben, die Schaltfläche Feldrechner drücken und „height“ als *Vorhandenes Feld erneuern* wählen, haben Sie bereits den Befehl „update buildings set height = ??? where height > 20“. Das einzige, was Sie in diesem Fall noch eingeben müssen, ist der **neue Wert**, z.B. geben Sie einfach `50` in das Textfeld des Ausdrucksеditors ein, um die Höhe der zuvor ausgewählten Gebäude zu setzen.

2. **Achten Sie auf Anführungszeichen**: Einfache Anführungszeichen geben ein Literal zurück, d.h. ein Text zwischen einfachen Anführungszeichen (`'145'`) wird als String interpretiert. Doppelte Anführungszeichen geben den Wert des Textes zurück, also verwenden Sie sie für Felder (`"myfield"`). Felder können auch ohne Anführungszeichen verwendet werden (`myfield`). Keine Anführungszeichen für Zahlen (`3.16`).

Bemerkung

Die Funktionen nehmen normalerweise als Argument eine Zeichenkette für den Feldnamen. Schreiben Sie also:

```
attribute( @atlas_feature, 'height' ) -- returns the value stored in the
↪"height" attribute of the current atlas feature
```

Und nicht:

```
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.
```

> **Tipp**
>
> **Verwenden Sie benannte Parameter, um das Lesen von Ausdrücken zu erleichtern.**
>
> Bei einigen Funktionen müssen viele Parameter angegeben werden. Die Ausdrucksmaschine unterstützt die Verwendung von benannten Parametern. Das bedeutet, dass man statt des kryptischen Ausdrucks `clamp( 1, 2, 9)` auch `clamp( min:=1, value:=2, max:=9)` verwenden kann. Auf diese Weise können die Argumente auch vertauscht werden, z.B. `Klammer( value:=2, max:=9, min:=1)`. Die Verwendung von benannten Parametern hilft zu klären, worauf sich die Argumente einer Ausdrucksfunktion beziehen, was hilfreich ist, wenn Sie später versuchen, einen Ausdruck zu interpretieren!

Anwendungsbeispiele für Ausdrücke

- Berechnen Sie mit dem Feldrechner ein „pop_density“-Feld unter Verwendung der bestehenden „total_pop“- und „area_km2“-Felder:

```
"total_pop" / "area_km2"
```

- Beschriften oder kategorisieren Sie Objekte auf der Grundlage ihrer Größe:

```
CASE WHEN $area > 10 000 THEN 'Larger' ELSE 'Smaller' END
```

- Aktualisieren Sie das Feld „density_level“ mit zugehörigen Kategorien zu den „pop_density“ Werten:

```
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
```

- Wenden Sie einen kategorisierten Stil auf alle Objekte an, je nachdem, ob der durchschnittliche Hauspreis kleiner oder größer als 10.000 € pro Quadratmeter ist:

```
"price_m2" > 10000
```

- Mit dem „Objekte über Ausdruck wählen…“ Werkzeug, wählen Sie alle Objekte, die eine Fläche mit „Hoher Bevölkerungsdichte“ und einem durchschnittlichen Hauspreis größer als 10000€ pro Quadratmeter darstellen:

```
"density_level" = 'High population density' and "price_m2" > 10000
```

  Der vorherige Ausdruck kann auch verwendet werden, um zu definieren, welche Objekte auf der Karte beschriftet oder angezeigt werden sollen.

- Wählen Sie Objekte aus dem Layer „lands“ aus, die sich mit dem Zonentyp („zone_typ“) natürlich („Natural“) überschneiden:

```
overlay_intersects( layer:='lands', filter:="zone_type"='Natural' )
```

- Zählen Sie für jedes Objekt die Anzahl der Gebäude, die es enthält:

```
array_length( overlay_contains( layer:='buildings', expression:=@id ) )
```

- Erstellen Sie ein anderes Symbol (Typ) für den Layer, indem Sie den Geometriegenerator verwenden:

```
point_on_surface( @geometry )
```

- Erzeugen Sie mit `make_line` eine geschlossene Linie um ein vorhandenes Punkt-Objekt:

```
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 )␣
↪)
  )
)
```

- Erzeugen Sie in einem Drucklayout Beschriftungen der Objekte „airports“, die sich innerhalb des Layout-Elements „Map 1“ befinden:

```
with_variable( 'extent',
               map_get( item_variables( 'Map 1' ), 'map_extent' ),
               aggregate( 'airports', 'concatenate', "NAME",
                          intersects( @geometry, @extent ), ' ,'
                        )
             )
```

Ausdrücke speichern

Mit der Schaltfläche Aktuellen Ausdruck zu Benutzerausdrücken hinzufügen oberhalb des Ausdrucksеditor-Rahmens können Sie wichtige Ausdrücke speichern, auf die Sie schnell zugreifen möchten. Diese sind über die Gruppe **Benutzerausdrücke** im mittleren Bedienfeld verfügbar. Sie werden im *Benutzerprofil* unter der Datei (`<userprofile>/QGIS/QGIS3.ini`) gespeichert und sind in allen Ausdrucksdialogen in allen Projekten des aktuellen Benutzerprofils verfügbar.

Eine Reihe von Werkzeugen, die oberhalb des Ausdrucksеditors verfügbar sind, hilft Ihnen bei der Verwaltung der Benutzerausdrücke:

- Aktuellen Ausdruck zu Benutzerausdrücken hinzufügen: den Ausdruck im Benutzerprofil speichern; zur einfachen Identifizierung können eine Beschreibung und ein Hilfetext hinzugefügt werden.
- Gewählten Ausdruck aus Benutzerausdrücken bearbeiten, sowie deren Hilfe und Beschreibung
- Gewählten Ausdruck aus Benutzerausdrücken löschen
- Benutzerausdrücke importieren aus einer `.json`-Datei in den Ordner des aktiven Benutzerprofils
- Benutzerausdrücke exportieren als `.json`-Datei; alle Benutzerausdrücke in der `QGIS3.ini`-Datei im Benutzerprofil werden gemeinsam genutzt

9.1.2 Funktions Editor

Mit dem Reiter *Funktionseditor* ist man in der Lage, eigene Funktionen mit Python zu schreiben. Dies bietet einen einfachen und komfortablen Weg, um besondere Anforderungen zu erfüllen, die nicht durch die enthaltenen Funktionen abgedeckt werden.

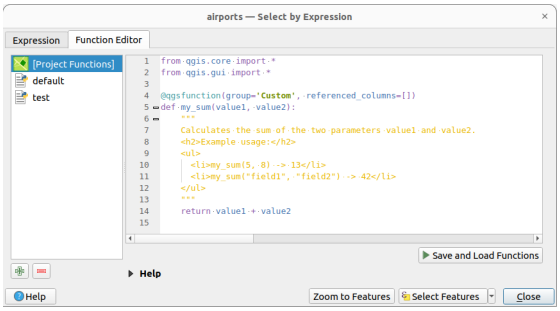


Abb. 9.2: Der Funktionseditor Reiter

Um eine neue Funtkion zu erstellen:

1. Drücken Sie auf die New File Schaltfläche.
2. Wählen Sie aus, wo die neue Funktion gespeichert werden soll. Sie können wählen, ob Sie sie entweder in einer *Funktionsdatei* oder in *Projektfunktionen* speichern möchten.

   - Wenn Sie *Funktionsdatei* auswählen, müssen Sie einen Namen für die Python-Datei `*.py` eingeben, die auf einer QGIS-Vorlagendatei basiert und im Ordner `/python/expressions` des aktiven *Benutzerprofil-Verzeichnisses* gespeichert wird und somit für alle Projekte verfügbar ist, die Sie mit diesem Benutzerprofil öffnen.

     Benutzerdefinierte Python-Funktionen, die als Funktionsdateien gespeichert sind, werden automatisch geladen, wenn Sie ein neues Projekt unter dem Benutzerprofil starten. Beachten Sie, dass Sie zum Freigeben solcher Python-Funktionen die Datei `*.py` im Ordner `/python/expressions` freigeben müssen.

   - Wenn Sie *Projektfunktionen* auswählen, wird die neue Funktion in der Projektdatei gespeichert und ist nur für dieses Projekt verfügbar.

     Benutzerdefinierte Python-Funktionen, die als Projektfunktionen gespeichert sind, können einfach durch Weitergabe der Projektdatei, in der sie gespeichert wurden, gemeinsam genutzt werden. Beim Öffnen eines Projekts mit Projektfunktionen kann QGIS diese laden oder ignorieren, je nachdem, ob *In Projekt eingebetteten Python-Code aktivieren* ausgewählt ist oder nicht.

   Im linken Bereich der Registerkarte *Funktionseditor* wird ein neuer Eintrag hinzugefügt.
3. Das rechte Fenster zeigt den Inhalt der Datei an, eine Python-Skriptvorlage.
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 *Python*. Online API documentation for Qt and PyQGIS objects is also accessible pressing the `F1` key.

   Weiter unten finden Sie allgemeine Informationen zum Einrichten des Dekorators @qgsfunction.
5. Drücken Sie auf die Schaltfläche *Lade- und Speicherfunktionen*. Die von ihnen geschriebene Funktion wird in der Liste der Funktionen im Reiter *Ausdruck* in der Voreinstellung in der Gruppe `Custom` hinzugefügt.
6. Haben Sie Spaß mit ihrer neuen Funktion.
7. Wenn die Funktion verbessert werden soll, geht man zum Reiter *Funktionseditor*, nimmt die Änderungen vor und drückt erneut auf die Schaltfläche *Lade- und Speicherfunktionen*. Damit werden die Änderungen in der Datei und somit auch im Reiter Ausdruck verfügbar.

Um eine benutzerdefinierte Funktion zu löschen, gehen Sie wie folgt vor:

1. Aktivieren Sie den Reiter *Funktionseditor*
2. Wählen Sie die Funktion in der Liste aus

3. Drücken Sie die Taste Gewählte Funktionsdatei löschen. Die Funktion wird aus der Liste entfernt und je nach Speicherort wird die entsprechende `*.py`-Datei aus dem Benutzerprofilordner gelöscht oder die Python-Funktionen werden aus der Projektdatei entfernt.

> **Bemerkung**
>
> Wenn eine Projektdatei mit Projektfunktionen geschlossen wird, werden die entsprechenden Projektfunktionen entladen und sind in der QGIS-Sitzung nicht mehr verfügbar.

Beispiel

Hier ist ein kurzes Beispiel, wie Sie Ihre eigene Funktion `my_sum` erstellen, die mit zwei Werten arbeitet.

```
from qgis.core import *
from qgis.gui import *

@qgsfunction(group='Custom', referenced_columns=[])
def my_sum(value1, value2):
    """
    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
```

Der Dekorator `@qgsfunction` akzeptiert die folgenden Argumente:

- Das Argument `Gruppe` gibt die Gruppe an, in der die Funktion im Ausdrucks-Dialog aufgelistet werden soll.
- „usesgeometry“: Setzen Sie diesen Wert auf „True“, wenn der Ausdruck Zugriff auf die Geometrie der Features erfordert. Standardmäßig ist der Wert „False“.
- `handlesnull`: Setzen Sie diesen Wert auf `True`, wenn der Ausdruck eine benutzerdefinierte Behandlung für NULL-Werte hat. Bei `False` (Standard) ist das Ergebnis immer NULL, sobald ein Parameter NULL ist.
- `referenced_columns`: Ein Array von Attributnamen, die zum Ausführen der Funktion erforderlich sind. Der Standardwert ist `[QgsFeatureRequest.ALL_ATTRIBUTES]`. Durch Angabe einer Teilmenge von Feldern oder einer leeren Liste wird die Ausführung beschleunigt.
- `params_as_list`: Setzen Sie diesen Wert auf `True`, um die Funktionsparameter als Liste zu übergeben. Bei `False` (Standard) erhält die Funktion die Parameter als einzelne Argumente.

Die Funktion selbst akzeptiert eine beliebige Anzahl und Art von Parametern, die Sie an Ihre Funktion übergeben möchten. Diese sollten vor den folgenden optionalen Argumenten festgelegt werden:

- `feature`: das aktuelle Objekt
- `parent`: das Objekt `QgsExpression`
- `context`: Wenn ein Argument namens `context` an der letzten Position gefunden wird, wird diese Variable ein `QgsExpressionContext` Objekt enthalten, das Zugang zu verschiedenen zusätzlichen Informationen wie Ausdrucksvariablen gibt. Z.B. `context.variable( 'layer_id' )`

Wenn diese Argumente in der Funktionssignatur vorhanden sind, werden sie automatisch an die Funktion übergeben, ohne dass sie im Ausdruck angegeben werden müssen.

Die vorherige Beispielfunktion kann dann in Ausdrücken verwendet werden:

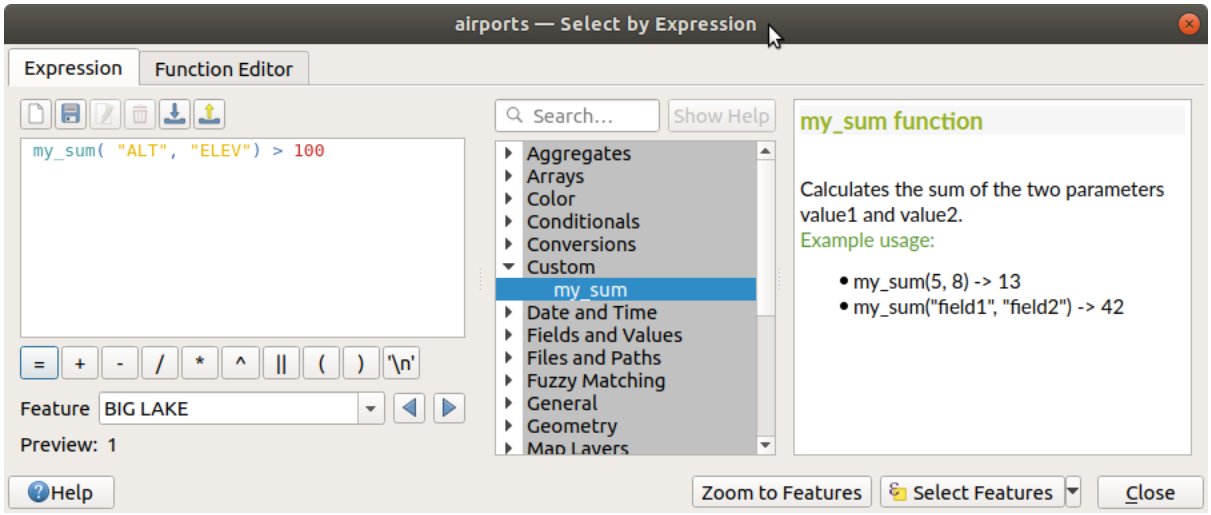


Abb. 9.3: Benutzerdefinierte Funktion im Reiter Ausdruck

Weitere Beispiele und zusätzliche Argumente für benutzerdefinierte Funktionen finden Sie unter `qgsfunction`. Weitere Informationen zum Erstellen von Python-Code finden Sie im PyQGIS-Developer-Cookbook.

9.2 Liste der Funktionen

Die in QGIS verfügbaren Funktionen, Operatoren und Variablen sind im Folgenden nach Kategorien gruppiert aufgeführt.

9.2.1 Aggregatsfunktionen

Diese Gruppe enthält Funktionen, die Werte über Layern und Feldern aggregieren.

aggregate

Gibt einen Gesamtwert zurück, der mit Objekten aus einem anderen Layer berechnet wurde.

Syntax	aggregate(layer, aggregate, expression, [filter], [concatenator=''], [order_by]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - eine Zeichenkette, die entweder einen Layernamen oder eine Layer-ID darstellt • **aggregate** - eine Zeichenkette, die dem zu berechnenden Aggregat entspricht. Gültige Optionen sind: – count – count_distinct – count_missing – minimum oder min – maximum oder max – sum – mean – median – stdev – stdevsample – range – minority – majority – q1: erstes Viertel – q3: drittes Viertel – iqr: inter quartile range – min_length: Mindestlänge der Zeichenkette – max_length: maximale Länge der Zeichenkette – concatenate: Strings mit einem Verknüpfungszeichen verbinden – concatenate_unique: eindeutige Zeichenketten mit einem Verknüpfungszeichen verbinden – collect: eine mehrteilige, aggregierte Geometrie erstellen – array_agg: ein Array der aggregierten Werte erstellen • **expression** - Unterausdruck oder zu aggregierender Feldname • **filter** - optionaler Filterausdruck zur Einschränkung der Objekte, die für die Berechnung des Aggregats verwendet werden. Felder und Geometrie stammen von den Objekten auf dem verbundenen Layer. Auf das Quellobjekt kann mit der Variablen @parent zugegriffen werden. • **concatenator** - optionale Zeichenfolge für die Verknüpfung von Werten für die Aggregate „concatenate“ und „concatenate_unique • **order_by** - optionaler Filterausdruck zur Sortierung der Objekte, die für die Berechnung des Aggregats verwendet werden. Felder und Geometrie stammen von den Objekten des verbundenen Layers. Standardmäßig werden die Objekte in einer nicht spezifizierten Reihenfolge zurückgegeben.
Beispiele	• `aggregate(layer:='rail_stations',aggregate:='sum', expression:="passengers")` → Summe aller Werte aus dem Feld „passengers“ im Layer „rail_stations“ • `aggregate('rail_stations','sum', "passengers"/7)` → errechnet einen Tagesdurchschnitt von „passengers“ durch Teilung der „passengers“ durch 7, bevor die Werte addiert werden • `aggregate(layer:='rail_stations',aggregate:='sum', expression:="passengers",filter:="class">3)` → summiert alle Werte aus dem Feld „passengers“ nur von Objekten, bei denen das Attribut „class“ größer als 3 ist • `aggregate(layer:='rail_stations',aggregate:='concatenate', expression:="name", concatenator:=',')` → kommagetrennte Liste des Namensfeldes für alle Objekte im Layer „rail_stations“ • `aggregate(layer:='countries', aggregate:='max', expression:="code", filter:=intersects( @geometry, geometry(@parent) ) )` → Der Ländercode eines sich überschneidenden Landes aus dem Layer „countries“ • `aggregate(layer:='rail_stations',aggregate:='sum', expression:="passengers",filter:=intersects( @atlas_geometry, @geometry ) )` → Summe aller Werte aus dem Feld „passengers“ im Feld „rail_stations“ im aktuellen Atlas. • `aggregate(layer:='rail_stations', aggregate:='collect',`

array_agg

Gibt ein Array von aggregierten Werten zurück, die aus einem Feld oder Ausdruck stammen.

Syntax	array_agg(expression, [group_by], [filter], [order_by]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden • **order_by** - optionaler Ausdruck für die Reihenfolge der Objekte, die für die Berechnung des Aggregats verwendet werden. Standardmäßig werden die Objekte in einer nicht spezifizierten Reihenfolge zurückgegeben.
Beispiele	• `array_agg("name",group_by:="state")` → Liste der Namenswerte, gruppiert nach dem „state“-Feld

collect

Liefert eine Multipart-Geometrie der aggregierten Geometrien eines Ausdrucks

Syntax	collect(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Geometrieausdruck zum Aggregieren • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `collect( @geometry )` → Multipart-Geometrie von aggregierten Geometrien • `collect( centroid(@geometry), group_by:="region", filter:= "use" = 'civilian' )` → aggregierte Zentroiden der „civilian“-Objekte auf der Grundlage der „region“

concatenate

Gibt alle aggregierten Zeichenketten aus einem Feld oder Ausdruck zurück, die durch ein Trennzeichen verbunden sind.

Syntax	concatenate(expression, [group_by], [filter], [concatenator], [order_by]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden • **concatenator** - optionale Zeichenkette für die Verknüpfung von Werten. Standardmäßig leer. • **order_by** - optionaler Ausdruck für die Reihenfolge der Objekte, die für die Berechnung des Aggregats verwendet werden. Standardmäßig werden die Objekte in einer nicht spezifizierten Reihenfolge zurückgegeben.
Beispiele	• `concatenate("town_name",group_by:="state",concatenator:=', ')` → kommagetrennte Liste von „town-name“, gruppiert nach dem Feld „state“

concatenate_unique

Gibt alle eindeutigen Zeichenfolgen aus einem Feld oder Ausdruck zurück, die durch ein Trennzeichen verbunden sind.

Syntax	concatenate_unique(expression, [group_by], [filter], [concatenator], [order_by]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden • **concatenator** - optionale Zeichenkette für die Verknüpfung von Werten. Standardmäßig leer. • **order_by** - optionaler Ausdruck für die Reihenfolge der Objekte, die für die Berechnung des Aggregats verwendet werden. Standardmäßig werden die Objekte in einer nicht spezifizierten Reihenfolge zurückgegeben.
Beispiele	• `concatenate_unique("town_name",group_by:="state", concatenator:=',')` → kommagetrennte Liste von eindeutigen „town-name“, gruppiert nach dem Feld „state“

count

Gibt die Anzahl der übereinstimmenden Objekte zurück.

Syntax	count(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `count("stations",group_by:="state")` → Anzahl der „stations“, gruppiert nach Feld „state“

count_distinct

Gibt die Anzahl der eindeutigen Werte zurück.

Syntax	count_distinct(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `count_distinct("stations",group_by:="state")` → Anzahl der eindeutigen „stations“, gruppiert nach dem Feld „state“

count_missing

Gibt die Anzahl der fehlenden Werte (NULL) zurück.

Syntax	count_missing(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `count_missing("stations",group_by:="state")` → Anzahl der fehlenden Werte (NULL) in den „stations“, gruppiert nach dem Feld „state“

iqr

Gibt den berechneten Interquartilsbereich aus einem Feld oder Ausdruck zurück.

Syntax	iqr(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `iqr("population",group_by:="state")` → Interquartilsbereich des Wertes „population“, gruppiert nach dem Feld „state“

majority

Gibt die aggregierte Mehrheit der Werte (am häufigsten vorkommender Wert) aus einem Feld oder Ausdruck zurück.

Syntax	majority(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `majority("class",group_by:="state")` → am häufigsten vorkommender „class“-Wert, gruppiert nach dem Feld „state“

max_length

Gibt die maximale Länge von Zeichenketten aus einem Feld oder Ausdruck zurück.

Syntax	max_length(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `max_length("town_name",group_by:="state")` → maximale Länge von „town_name“, gruppiert nach dem Feld „state“

maximum

Gibt den aggregierten Maximalwert aus einem Feld oder Ausdruck zurück.

Syntax	maximum(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `maximum("population",group_by:="state")` → maximaler Wert der „population“, gruppiert nach dem Feld „state“

mean

Gibt den aggregierten Mittelwert aus einem Feld oder Ausdruck zurück.

Syntax	mean(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `mean("population",group_by:="state")` → mittlerer Wert der „population“, gruppiert nach dem Feld „state“

median

Gibt den aggregierten Medianwert aus einem Feld oder Ausdruck zurück.

Syntax	median(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `median("population",group_by:="state")` → medianer Wert der „population“, gruppiert nach dem Feld „state“

min_length

Gibt die Mindestlänge von Zeichenketten aus einem Feld oder Ausdruck zurück.

Syntax	min_length(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `min_length("town_name",group_by:="state")` → Mindestlänge von „town_name“, gruppiert nach dem Feld „state“

minimum

Gibt den aggregierten Mindestwert aus einem Feld oder Ausdruck zurück.

Syntax	minimum(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `minimum("population",group_by:="state")` → minimaler Wert der „population“, gruppiert nach dem Feld „state“

minority

Gibt die aggregierte Minderheit von Werten (am wenigsten vorkommender Wert) aus einem Feld oder Ausdruck zurück.

Syntax	minority(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `minority("class",group_by:="state")` → am wenigsten vorkommender „class“-Wert, gruppiert nach dem Feld „state“

q1

Gibt das berechnete erste Quartil aus einem Feld oder Ausdruck zurück.

Syntax	q1(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `q1("population",group_by:="state")` → erstes Quartil der „population“, gruppiert nach dem Feld „state“

q3

Gibt das berechnete dritte Quartil aus einem Feld oder Ausdruck zurück.

Syntax	q3(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `q3("population",group_by:="state")` → drittes Quartil der „population“, gruppiert nach dem Feld „state“

range

Gibt den aggregierten Wertebereich (Maximum - Minimum) aus einem Feld oder Ausdruck zurück.

Syntax	range(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `range("population",group_by:="state")` → Wertebereich der „population“, gruppiert nach dem Feld „state“

relation_aggregate

Gibt einen Gesamtwert zurück, der anhand aller übereinstimmenden untergeordneten Objekte einer Layer-Beziehung berechnet wird.

Syntax	relation_aggregate(relation, aggregate, expression, [concatenator=''], [order_by]) [] kennzeichnet optionale Argumente
Argumente	• **relation** - eine Zeichenfolge, die eine Beziehungs-ID angibt • **aggregate** - eine Zeichenkette, die dem zu berechnenden Aggregat entspricht. Gültige Optionen sind: – count – count_distinct – count_missing – minimum oder min – maximum oder max – sum – mean – median – stdev – stdevsample – range – minority – majority – q1: erstes Viertel – q3: drittes Viertel – iqr: inter quartile range – min_length: Mindestlänge der Zeichenkette – max_length: maximale Länge der Zeichenkette – concatenate: Strings mit einem Verknüpfungszeichen verbinden – concatenate_unique: eindeutige Zeichenketten mit einem Verknüpfungszeichen verbinden – collect: eine mehrteilige, aggregierte Geometrie erstellen – array_agg: ein Array der aggregierten Werte erstellen • **expression** - Unterausdruck oder zu aggregierender Feldname • **concatenator** - optionale Zeichenkette zum Verbinden von Werten für das ‚concatenate'-Aggregat • **order_by** - optionaler Ausdruck für die Reihenfolge der Objekte, die für die Berechnung des Aggregats verwendet werden. Felder und Geometrie stammen von den Objekten auf dem verbundenen Layer. Standardmäßig werden die Objekte in einer nicht spezifizierten Reihenfolge zurückgegeben.
Beispiele	• `relation_aggregate(relation:='my_relation', aggregate:='mean',expression:="passengers")` → Mittelwert aller übereinstimmender, untergeordneter Objekte unter Verwendung der Beziehung „my_relation". • `relation_aggregate('my_relation','sum', "passengers"/7)` → Summe des „passengers"-Feldes geteilt durch 7 für alle übereinstimmenden, untergeordneten Objekte unter Verwendung der Beziehung „my_relation". • `relation_aggregate('my_relation','concatenate', "towns", concatenator:=',')` → durch Kommata getrennte Liste der Werte des „towns"-Feldes für alle übereinstimmenden, untergeordneten Objekte, die die Beziehung „my_relation" verwenden • `relation_aggregate('my_relation','array_agg', "id")` → Array des id-Feldes aus allen übereinstimmenden, untergeordneten Objekten unter Verwendung der Beziehung „my_relation"

Siehe auch: *Festlegung von Beziehungen zwischen mehreren Layern*

stdev

Gibt den aggregierten Wert der Standardabweichung aus einem Feld oder Ausdruck zurück.

Syntax	stdev(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `stdev("population",group_by:="state")` → Standardabweichung der „population“, gruppiert nach dem Feld „state“

sum

Gibt den aggregierten Summenwert aus einem Feld oder Ausdruck zurück.

Syntax	sum(expression, [group_by], [filter]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - Unterausdruck des zu aggregierenden Feldes • **group_by** - optionaler Ausdruck für die Gruppierung von Aggregatberechnungen • **filter** - optionaler Ausdruck zum Filtern von Objekten, die zur Berechnung des Aggregats verwendet werden
Beispiele	• `sum("population",group_by:="state")` → summierter Wert der „population“, gruppiert nach dem Feld „state“

9.2.2 Array-Funktionen

Diese Gruppe enthält Funktionen zur Erzeugung und Manipulation von Arrays (auch als „Listen“ oder „Felddaten“ bekannt). Die Reihenfolge der Werte im Array ist wichtig. Eine Ausnahme bildet die ‚map‘ Datenstruktur, bei der die Reihenfolge der Schlüssel-Werte-Paare unwichtig ist. Die Werte werden in diesem Fall durch den zugehörigen Schlüssel identifiziert.

array

Gibt ein Array zurück, das alle als Parameter übergebenen Werte enthält.

Syntax	array(value1, value2, …)
Argumente	• **value** - ein Wert
Beispiele	• `array(2,10)` → [2, 10] • `array(2,10)[0]` → 2

array_all

Gibt TRUE zurück, wenn ein Array alle Werte eines gegebenen Arrays enthält.

Syntax	array_all(array_a, array_b)
Argumente	• **array_a** - ein Array • **array_b** - das Array der zu durchsuchenden Werte
Beispiele	• `array_all(array(1,2,3),array(2,3))` → TRUE • `array_all(array(1,2,3),array(1,2,4))` → FALSE

array_append

Gibt ein Array zurück, dem der angegebene Wert am Ende hinzugefügt wurde.

Syntax	array_append(array, value)
Argumente	• **array** - Ein Array • **value** - der hinzuzufügende Wert
Beispiele	• `array_append(array(1,2,3),4)` → [1, 2, 3, 4]

array_cat

Gibt ein Array zurück, das alle angegebenen Arrays verkettet enthält.

Syntax	array_cat(array1, array2, …)
Argumente	• **array** - Ein Array
Beispiele	• `array_cat(array(1,2),array(2,3))` → [1, 2, 2, 3]

array_contains

Gibt TRUE zurück, wenn ein Array den angegebenen Wert enthält.

Syntax	array_contains(array, value)
Argumente	• **array** - Ein Array • **value** - der zu suchende Wert
Beispiele	• `array_contains(array(1,2,3),2)` → TRUE

array_count

Zählt die Anzahl der Vorkommen eines bestimmten Wertes in einem Array.

Syntax	array_count(array, value)
Argumente	• **array** - Ein Array • **value** - der zu zählende Wert
Beispiele	• `array_count(array('a', 'b', 'c', 'b'), 'b')` → 2

array_distinct

Gibt ein Array zurück, die einzelnen Werte des angegebenen Arrays enthält.

Syntax	array_distinct(array)
Argumente	• **array** - Ein Array
Beispiele	• `array_distinct(array(1,2,3,2,1))` → [1, 2, 3]

array_filter

Gibt ein Array zurück, das nur die Elemente enthält, für die der Ausdruck als wahr ausgewertet wird.

Syntax	array_filter(array, expression, [limit=0]) [] kennzeichnet optionale Argumente
Argumente	• **array** - Ein Array • **expression** - ein Ausdruck, der für jedes Element ausgewertet wird. Die Variable *@element* wird durch den aktuellen Wert ersetzt. • **limit** - maximale Anzahl der zurückzugebenden Elemente. Verwenden Sie 0, um alle Werte zurückzugeben.
Beispiele	• `array_filter(array(1,2,3),@element < 3)` → [1, 2] • `array_filter(array(1,2,3),@element < 3, 1)` → [1]

array_find

Gibt den niedrigsten Index (0 für den ersten) eines Wertes innerhalb eines Arrays zurück. Gibt -1 zurück, wenn der Wert nicht gefunden wird.

Syntax	array_find(array, value)
Argumente	• **array** - Ein Array • **value** - der zu suchende Wert
Beispiele	• `array_find(array('a', 'b', 'c'), 'b')` → 1 • `array_find(array('a', 'b', 'c', 'b'), 'b')` → 1

array_first

Gibt den ersten Wert eines Arrays zurück.

Syntax	array_first(array)
Argumente	• **array** - Ein Array
Beispiele	• `array_first(array('a','b','c'))` → ‚a'

array_foreach

Gibt ein Array zurück, in dem der angegebene Ausdruck für jedes Element ausgewertet wird.

Syntax	array_foreach(array, expression)
Argumente	• **array** - Ein Array • **expression** - Ein Ausdruck, der für jedes Element ausgewertet wird. Die Variable *@element* wird durch den aktuellen Wert ersetzt, die Variable *@counter* durch den aktuellen Index (beginnend mit 0).
Beispiele	• `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

Gibt den N-ten Wert (0 für den ersten) oder den letzten -N-ten Wert (-1 für den letzten) eines Arrays zurück.

Syntax	array_get(array, pos)
Argumente	• **array** - Ein Array • **pos** - der zu ermittelnde Index (auf 0 basierend)
Beispiele	• `array_get(array('a','b','c'),1)` → ‚b' • `array_get(array('a','b','c'),-1)` → ‚c'

Hinweis

Sie können auch den *Index-Operator ([]) 1* verwenden, um einen Wert aus einem Array zu erhalten.

array_insert

Gibt ein Array zurück, dem der angegebene Wert an der angegebenen Position hinzugefügt wurde.

Syntax	array_insert(array, pos, value)
Argumente	• **array** - Ein Array • **pos** - die Position, an der hinzugefügt werden soll (0-basiert) • **value** - der hinzuzufügende Wert
Beispiele	• `array_insert(array(1,2,3),1,100)` → [1, 100, 2, 3]

array_intersect

Gibt TRUE zurück, wenn mindestens ein Element von array1 in array2 vorhanden ist.

Syntax	array_intersect(array1, array2)
Argumente	• **array1** - ein Array • **array2** - ein anderer Array
Beispiele	• `array_intersect(array(1,2,3,4),array(4,0,2,5))` → TRUE

array_last

Gibt den letzten Wert eines Arrays zurück.

Syntax	array_last(array)
Argumente	• **array** - Ein Array
Beispiele	• `array_last(array('a','b','c'))` → ‚c‘

array_length

Gibt die Anzahl der Elemente eines Arrays zurück.

Syntax	array_length(array)
Argumente	• **array** - Ein Array
Beispiele	• `array_length(array(1,2,3))` → 3

array_majority

Gibt die häufigsten Werte in einem Array zurück.

Syntax	array_majority(array, [option='all']) [] kennzeichnet optionale Argumente
Argumente	• **array** - Ein Array • **option='all'** - eine Zeichenkette, die die Behandlung der Rückgabewerte angibt. Gültige Optionen sind: – all: Standard, die häufigsten Werte werden in einem Array zurückgegeben. – any: Gibt einen der häufigsten Werte zurück. – median: Gibt den Median der häufigsten Werte zurück. Nicht arithmetische Werte werden ignoriert. – real_majority: Gibt den Wert zurück, der mehr als die Hälfte der Größe des Arrays ausmacht.
Beispiele	• `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

Gibt den Maximalwert eines Arrays zurück.

Syntax	array_max(array)
Argumente	• **array** - Ein Array
Beispiele	• `array_max(array(0,42,4,2))` → 42

array_mean

Gibt den Mittelwert der arithmetischen Werte in einem Array zurück. Nicht numerische Werte im Array werden ignoriert.

Syntax	array_mean(array)
Argumente	• **array** - Ein Array
Beispiele	• `array_mean(array(0,1,7,66.6,135.4))` → 42 • `array_mean(array(0,84,'a','b','c'))` → 42

array_median

Gibt den Median der arithmetischen Werte in einem Array zurück. Nicht arithmetische Werte im Array werden ignoriert..

Syntax	array_median(array)
Argumente	• **array** - Ein Array
Beispiele	• `array_median(array(0,1,42,42,43))` → 42 • `array_median(array(0,1,2,42,'a','b'))` → 1.5

array_min

Gibt den Mindestwert eines Arrays zurück.

Syntax	array_min(array)
Argumente	• **array** - Ein Array
Beispiele	• `array_min(array(43,42,54))` → 42

array_minority

Gibt die seltensten Werte in einem Array zurück.

Syntax	array_minority(array, [option='all']) [] kennzeichnet optionale Argumente
Argumente	• **array** - Ein Array • **option='all'** - eine Zeichenkette, die die Behandlung der Rückgabewerte angibt. Gültige Optionen sind: – all: Standard, alle seltenen Werte werden in einem Array zurückgegeben. – any: Gibt einen der seltensten Werte zurück. – median: Gibt den Median der seltensten Werte zurück. Nicht arithmetische Werte werden ignoriert. – real_minority: Gibt Werte zurück, die weniger als die Hälfte der Größe des Arrays ausmachen.
Beispiele	• `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

Gibt ein Array zurück, dem der angegebene Wert am Anfang hinzugefügt wurde..

Syntax	array_prepend(array, value)
Argumente	• **array** - Ein Array • **value** - der hinzuzufügende Wert
Beispiele	• `array_prepend(array(1,2,3),0)` → [0, 1, 2, 3]

array_prioritize

Gibt ein Array zurück, das nach der in einem anderen Array angegebenen Reihenfolge sortiert ist. Werte, die im ersten Array vorhanden sind, aber im zweiten Array fehlen, werden an das Ende des Ergebnisses angehängt.

Syntax	array_prioritize(array, array_prioritize)
Argumente	• **array** - Ein Array • **array_prioritize** - ein Array mit nach Priorität geordneten Werten
Beispiele	• `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

Gibt ein Array zurück, aus dem alle Einträge mit dem angegebenen Wert entfernt wurden.

Syntax	array_remove_all(array, value)
Argumente	• **array** - Ein Array • **value** - die zu entfernenden Werte
Beispiele	• `array_remove_all(array('a','b','c','b'),'b')` → [‚a', ‚c']

array_remove_at

Gibt ein Array zurück, aus dem das Element mit dem angegebenen Index entfernt wurde. Unterstützt positiven (0 für das erste Element) und negativen (der letzte -N-te Wert, -1 für das letzte Element) Index.

Syntax	array_remove_at(array, pos)
Argumente	• **array** - Ein Array • **pos** - die zu entfernende Position (0-basiert)
Beispiele	• `array_remove_at(array(1, 2, 3), 1)` → [1, 3] • `array_remove_at(array(1, 2, 3), -1)` → [1, 2]

array_replace

Gibt ein Array zurück, in dem der übergebene Wert, das Array oder die Zuordnung von Werten ersetzt wird..

Wert & Array-Variante

Gibt ein Array zurück, in dem der übergebene Wert oder das Array von Werten durch einen anderen Wert oder ein Array von Werten ersetzt wurde.

Syntax	array_replace(array, before, after)
Argumente	• **array** - das Eingabe-Array • **before** - der zu ersetzende Wert oder das zu ersetzende Array von Werten • **after** - der Wert oder das Array von Werten, die als Ersatz verwendet werden sollen
Beispiele	• `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-Variante

Gibt ein Array zurück, in dem die übergebenen Zuordnungsschlüssel durch ihre gepaarten Werte ersetzt sind.

Syntax	array_replace(array, map)
Argumente	• **array** - das Eingabe-Array • **map** - die Zuordnung von Schlüsseln und Werten
Beispiele	• `array_replace(array('APP', 'SHOULD', 'ROCK'),map('APP', 'QGIS','SHOULD','DOES'))` → [‚QGIS', ‚DOES', ‚ROCK']

Siehe auch: *replace*, *regexp_replace*

array_reverse

Gibt das angegebene Array mit den Array-Werten in umgekehrter Reihenfolge zurück.

Syntax	array_reverse(array)
Argumente	• **array** - Ein Array
Beispiele	• `array_reverse(array(2,4,0,10))` → [10, 0, 4, 2]

array_slice

Gibt einen Teil des Arrays zurück. Das Segment wird durch die Argumente start_pos und end_pos definiert.

Syntax	array_slice(array, start_pos, end_pos)
Argumente	• **array** - Ein Array • **start_pos** - der Index der Startposition des Teilstücks (0-basiert). Der Index start_pos wird in das Segment aufgenommen. Wenn Sie einen negativen start_pos verwenden, wird der Index vom Ende der Liste aus gezählt (-1-basiert). • **end_pos** - der Index der Endposition des Teilstücks (0-basiert). Der end_pos-Index wird in das Segment aufgenommen. Wenn Sie einen negativen end_pos verwenden, wird der Index vom Ende der Liste aus gezählt (-1-basiert).
Beispiele	• `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

Gibt das angegebene Array mit sortierten Elementen zurück.

Syntax	array_sort(array, [ascending=true]) [] kennzeichnet optionale Argumente
Argumente	• **array** - Ein Array • **ascending** - Setzen Sie diesen Parameter auf false, um das Array in absteigender Reihenfolge zu sortieren
Beispiele	• `array_sort(array(3,2,1))` → [1, 2, 3]

array_sum

Gibt die Summe der arithmetischen Werte in einem Array zurück. Nicht numerische Werte im Array werden ignoriert.

Syntax	array_sum(array)
Argumente	• **array** - Ein Array
Beispiele	• `array_sum(array(0,1,39.4,1.6,'a'))` → 42.0

array_to_string

Verkettet Array-Elemente zu einem String, getrennt durch ein Trennzeichen und unter Verwendung eines optionalen Strings für leere Werte.

Syntax	array_to_string(array, [delimiter=‘,‘], [empty_value=“]) [] kennzeichnet optionale Argumente
Argumente	• **array** - das Eingabe-Array • **delimiter** - das Trennzeichen, das verwendet wird, um verkettete Array-Elemente zu verbinden • **empty_value** - die optionale Zeichenkette, die als Ersatz für leere („Null-Länge“) Treffer verwendet wird
Beispiele	• `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‘

Siehe auch: *string_to_array*

generate_series

Erzeugt ein Array mit einer Zahlenfolge.

Syntax	generate_series(start, stop, [step=1]) [] kennzeichnet optionale Argumente
Argumente	• **start** - erster Wert der Zahlenfolge • **stop** - letzter Wert der Zahlenfolge • **step** - Wert, der als Schrittweite zwischen den Werten verwendet wird
Beispiele	• `generate_series(1,5)` → [1, 2, 3, 4, 5] • `generate_series(5,1,-1)` → [5, 4, 3, 2, 1]

geometries_to_array

Zerlegt eine Geometrie in einfachere Geometrien in einem Array.

Syntax	geometries_to_array(geometry)
Argumente	• **geometry** - die Eingabe-Geometrie
Beispiele	• `geometries_to_array(geom_from_wkt('MultiPoint (1 2, 5 21'))` → Ein Array, das ‚Point (1 2)' and ‚Point (5 21)' enthält • `geometries_to_array(geom_from_wkt('GeometryCollection (Polygon ((5 8, 4 1, 3 2, 5 8)),LineString (3 2, 4 2))'))` → ein Array von einer Polygon- und einer Liniengeometrien • `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))'))` → ein Array mit zwei Polygongeometrien

regexp_matches

Gibt ein Array aller Zeichenketten zurück, die von Erfassungsgruppen erfasst werden, und zwar in der Reihenfolge, in der die Gruppen selbst im übergebenen regulären Ausdruck gegen eine Zeichenkette erscheinen.

Syntax	regexp_matches(string, regex, [empty_value='']) [] kennzeichnet optionale Argumente
Argumente	• **string** - die Zeichenkette, aus der Gruppen anhand des regulären Ausdrucks erfasst werden sollen • **regex** - der reguläre Ausdruck, der zur Erfassung von Gruppen verwendet wird • **empty_value** - die optionale Zeichenkette, die als Ersatz für leere („Null-Länge") Treffer verwendet wird
Beispiele	• `regexp_matches('QGIS=>rocks','(.*)=>(.*)')` → [‚QGIS', ‚rocks'] • `regexp_matches('key=>','(.*)=>(.*)','empty value')` → [‚key', ‚empty value']

Siehe auch: *substr*, *regexp_substr*

string_to_array

Zerlegt eine Zeichenkette in ein Array unter Verwendung des angegebenen Trennzeichens und einer optionalen Zeichenkette für leere Werte.

Syntax	string_to_array(string, [delimiter=‘,‘], [empty_value=“]) [] kennzeichnet optionale Argumente
Argumente	• **string** - die Eingabezeichenkette • **delimiter** - das Trennzeichen, das zur Aufteilung der Eingabezeichenfolge verwendet wird • **empty_value** - die optionale Zeichenkette, die als Ersatz für leere („Null-Länge“) Treffer verwendet wird
Beispiele	• `string_to_array('1,2,3',',')` → [‚1‘, ‚2‘, ‚3‘] • `string_to_array('1,,3',',','0')` → [‚1‘, ‚0‘, ‚3‘]

Siehe auch: *array_to_string*

9.2.3 Farbfunktionen

Diese Gruppe enthält Funktionen zur Bearbeitung von Farben.

color_cmyk

Liefert die Zeichenkettendarstellung einer Farbe nach ihrem Cyan-, Magenta-, Gelb- und Schwarzkomponenten

Syntax	color_cmyk(cyan, magenta, yellow, black)
Argumente	• **cyan** - Cyan-Anteil der Farbe, als prozentualer ganzzahliger Wert von 0 bis 100 • **magenta** - Magenta-Anteil der Farbe, als prozentualer ganzzahliger Wert von 0 bis 100 • **yellow** - Gelb-Anteil der Farbe, als prozentualer ganzzahliger Wert von 0 bis 100 • **black** - Schwarz-Anteil der Farbe, als prozentualer ganzzahliger Wert von 0 bis 100
Beispiele	• `color_cmyk(100,50,0,10)` → ‚0,115,230‘

color_cmyka

Liefert die Zeichenkettendarstellung einer Farbe nach ihrem Cyan-, Magenta-, Gelb-, Schwarz- und Alpha-(Transparenz-)komponenten

Syntax	color_cmyka(cyan, magenta, yellow, black, alpha)
Argumente	• **cyan** - Cyan-Anteil der Farbe, als prozentualer ganzzahliger Wert von 0 bis 100 • **magenta** - Magenta-Anteil der Farbe, als prozentualer ganzzahliger Wert von 0 bis 100 • **yellow** - Gelb-Anteil der Farbe, als prozentualer ganzzahliger Wert von 0 bis 100 • **black** - Schwarz-Anteil der Farbe, als prozentualer ganzzahliger Wert von 0 bis 100 • **alpha** - Alphakomponente als Ganzzahlwert von 0 (völlig transparent) bis 255 (undurchsichtig).
Beispiele	• `color_cmyka(100,50,0,10,200)` → ‚0,115,230,200‘

color_cmykf

Gibt ein Farbobjekt basierend auf seinen Cyan-, Magenta-, Gelb-, Schwarz- und Alpha-Komponenten zurück.

Syntax	color_cmykf(cyan, magenta, yellow, black, [alpha=1.0]) [] kennzeichnet optionale Argumente
Argumente	• **cyan** - Cyan-Komponente als Fließkommawert von 0,0 bis 1,0 • **magenta** - Magenta-Komponente als Fließkommawert von 0,0 bis 1,0 • **yellow** - Gelb-Komponente als Fließkommawert von 0,0 bis 1,0 • **black** - Schwarz- Komponente als Fließkommawert von 0,0 bis 1,0 • **alpha** - Alpha-Komponente als Fließkommawert zwischen 0,0 und 1,0
Beispiele	• `color_cmykf(1,0.9,0.81,0.62)` → CMYKA: 1.00,0.90,0.81,0.62,1.00

color_grayscale_average

Wendet einen Graustufenfilter auf eine Farbe an und gibt sie zurück. Der Rückgabetyp entspricht dem Farbargument, d. h. einer Farbzeichenfolge oder einem Farbobjekt.

Syntax	color_grayscale_average(color)
Argumente	• **color –** eine Farbrepräsentation als Farbzeichenfolge oder ein Farbobjekt
Beispiele	• `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

Gibt eine String-Darstellung einer Farbe basierend auf den Attributen Farbton, Sättigung und Helligkeit zurück.

Syntax	color_hsl(hue, saturation, lightness)
Argumente	• **hue** - Farbton der Farbe, als ganzzahliger Wert von 0 bis 360 • **saturation** - Sättigungsprozentsatz der Farbe als ganzzahliger Wert von 0 bis 100 • **lightness** - prozentuale Helligkeit der Farbe als ganzzahliger Wert von 0 bis 100
Beispiele	• `color_hsl(100,50,70)` → ‚166,217,140‘

color_hsla

Liefert die Zeichenkettendarstellung einer Farbe nach ihrem Farb-, Sättigungs-, Helligkeits- und Alpha-(Transparenz-)Attributen

Syntax	color_hsla(hue, saturation, lightness, alpha)
Argumente	• **hue** - Farbton der Farbe, als ganzzahliger Wert von 0 bis 360 • **saturation** - Sättigungsprozentsatz der Farbe als ganzzahliger Wert von 0 bis 100 • **lightness** - prozentuale Helligkeit der Farbe als ganzzahliger Wert von 0 bis 100 • **alpha** - Alphakomponente als Ganzzahlwert von 0 (völlig transparent) bis 255 (undurchsichtig).
Beispiele	• `color_hsla(100,50,70,200)` → ‚166,217,140,200‘

color_hslf

Gibt ein Farbobjekt basierend auf seinen Attributen für Farbton, Sättigung und Helligkeit zurück.

Syntax	color_hslf(hue, saturation, lightness, [alpha=1.0]) [] kennzeichnet optionale Argumente
Argumente	• **hue** - Farbton als Fließkommawert zwischen 0,0 und 1,0 • **saturation** - Farbsättigung als Fließkommawert von 0,0 bis 1,0 • **lightness** - Helligkeit der Farbe als Fließkommawert von 0,0 bis 1,0 • **alpha** - Alpha-Komponente als Fließkommawert zwischen 0,0 und 1,0
Beispiele	• `color_hslf(0.3,0.52,0.7)` → HSLA: 0.30,0.52,0.70,1.00

color_hsv

Gibt eine String-Darstellung einer Farbe basierend auf den Attributen Farbton, Sättigung und Wert zurück.

Syntax	color_hsv(hue, saturation, value)
Argumente	• **hue** - Farbton der Farbe, als ganzzahliger Wert von 0 bis 360 • **saturation** - Sättigungsprozentsatz der Farbe als ganzzahliger Wert von 0 bis 100 • **value** - Wert in Prozent der Farbe als Ganzzahl von 0 bis 100
Beispiele	• `color_hsv(40,100,100)` → ‚255,170,0‘

color_hsva

Gibt eine String-Darstellung einer Farbe basierend auf den Attributen Farbton, Sättigung, Wert und Alpha (Transparenz) zurück.

Syntax	color_hsva(hue, saturation, value, alpha)
Argumente	• **hue** - Farbton der Farbe, als ganzzahliger Wert von 0 bis 360 • **saturation** - Sättigungsprozentsatz der Farbe als ganzzahliger Wert von 0 bis 100 • **value** - Wert in Prozent der Farbe als Ganzzahl von 0 bis 100 • **alpha** - Alpha-Komponente als Ganzzahlwert von 0 (völlig transparent) bis 255 (undurchsichtig)
Beispiele	• `color_hsva(40,100,100,200)` → ‚255,170,0,200‘

color_hsvf

Gibt ein Farbobjekt basierend auf seinen Attributen für Farbton, Sättigung und Wert zurück.

Syntax	color_hsvf(hue, saturation, value, [alpha=1.0]) [] kennzeichnet optionale Argumente
Argumente	• **hue** - Farbton als Fließkommawert zwischen 0,0 und 1,0 • **saturation** - Farbsättigung als Fließkommawert von 0,0 bis 1,0 • **value** - Wert der Farbe als Fließkommawert zwischen 0,0 und 1,0 • **alpha** - Alpha-Komponente als Fließkommawert zwischen 0,0 und 1,0
Beispiele	• `color_hsvf(0.4,1,0.6)` → HSVA: 0.40,1.00,0.60,1.00

color_mix

Gibt eine Farbe zurück, die die Rot-, Grün-, Blau- und Alpha-Werte zweier angegebener Farben basierend auf einem vorgegebenen Verhältnis mischt. Der zurückgegebene Typ entspricht den Farbargumenten, d. h. einer Farbzeichenfolge oder einem Farbobjekt.

Syntax	color_mix(color1, color2, ratio)
Argumente	• **color1** - eine Farbzeichenfolge oder ein Farbobjekt • **color2** - eine Farbzeichenfolge oder ein Farbobjekt • **ratio** - ein Verhältnis
Beispiele	• `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

Gibt eine Zeichenkette zurück, die eine Farbe darstellt, die die Rot-, Grün-, Blau- und Alphawerte von zwei angegebenen Farben in einem bestimmten Verhältnis mischt.

Syntax	color_mix_rgb(color1, color2, ratio)
Argumente	• **color1** - eine Farbzeichenfolge • **color2** - eine Farbzeichenfolge • **ratio** - ein Verhältnis
Beispiele	• `color_mix_rgb('0,0,0','255,255,255',0.5)` → ‚127,127,127,255‘

color_part

Gibt eine bestimmte Komponente aus einer Farbzeichenfolge oder einem Farbobjekt zurück, z. B. die rote Komponente oder die Alpha-Komponente.

Syntax	color_part(color, component)
Argumente	• **color** - eine Farbzeichenfolge oder ein Farbobjekt • **component** - eine Zeichenkette, die der zurückzugebenden Farbkomponente entspricht. Gültige Optionen sind: – red: RGB-Rot-Komponente (0-255) – green: RGB-Grün-Komponente (0-255) – blue: RGB-Blau-Komponente (0-255) – alpha: Alpha-Wert (Transparenz) (0-255) – hue: HSV-Farbton (0-360) – saturation: HSV-Sättigung (0-100) – value: HSV-Wert (0-100) – hsl_hue: HSL-Farbton (0-360) – hsl_saturation: HSL-Sättigung (0-100) – lightness: HSL-Helligkeit (0-100) – cyan: CMYK Cyan-Anteil (0-100) – magenta: CMYK Magenta-Anteil (0-100) – yellow: CMYK Gelb-Anteil (0-100) – black: CMYK Schwarz-Anteil(0-100)
Beispiele	• `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

Gibt eine String-Darstellung einer Farbe auf der Grundlage ihrer Rot-, Grün- und Blaukomponenten zurück.

Syntax	color_rgb(red, green, blue)
Argumente	• **red** - Rotanteil als ganzzahliger Wert von 0 bis 255 • **green** - Grünanteil als ganzzahliger Wert von 0 bis 255 • **blue** - Blauanteil als ganzzahliger Wert von 0 bis 255
Beispiele	• `color_rgb(255,127,0)` → ‚255,127,0‘

color_rgba

Gibt eine String-Darstellung einer Farbe basierend auf ihren Rot-, Grün-, Blau- und Alpha-Komponenten (Transparenz) zurück.

Syntax	color_rgba(red, green, blue, alpha)
Argumente	• **red** - Rotanteil als ganzzahliger Wert von 0 bis 255 • **green** - Grünanteil als ganzzahliger Wert von 0 bis 255 • **blue** - Blauanteil als ganzzahliger Wert von 0 bis 255 • **alpha** - Alphakomponente als Ganzzahlwert von 0 (völlig transparent) bis 255 (undurchsichtig).
Beispiele	• `color_rgba(255,127,0,200)` → ‚255,127,0,200‘

color_rgbf

Gibt ein Farbobjekt basierend auf seinen Rot-, Grün-, Blau- und Alpha-Komponenten zurück.

Syntax	color_rgbf(red, green, blue, [alpha=1.0]) [] kennzeichnet optionale Argumente
Argumente	• **rot** – Rotanteil als Fließkommawert zwischen 0,0 und 1,0 • **grün** – Grünanteil als Fließkommawert zwischen 0,0 und 1,0 • **blau** – Blauanteil als Fließkommawert zwischen 0,0 und 1,0 • **alpha** - Alpha-Komponente als Fließkommawert zwischen 0,0 und 1,0
Beispiele	• `color_rgbf(1.0,0.5,0)` → RGBA: 1.00,0.50,0.00,1.00

create_ramp

Gibt eine Farbverlaufsrampe aus einer Zuordnung von Zeichenfolgen und Stufen zurück.

Syntax	create_ramp(map, [discrete=false]) [] kennzeichnet optionale Argumente
Argumente	• **map** - eine Zuordnung von Zeichenfolgen und Schritten • **discrete** - Setzen Sie diesen Parameter auf true, um einen diskrete Farbverlauf zu erstellen
Beispiele	• `ramp_color(create_ramp(map(0,'0,0,0',1,'255,0,0')),1)` → ‚255,0,0,255‘

darker

Gibt eine dunklere (oder hellere) Farbe zurück. Der zurückgegebene Typ entspricht den Farbargumenten, d. h. einer Farbzeichenfolge oder einem Farbobjekt.

Syntax	darker(color, factor)
Argumente	• **color** - eine Farbzeichenfolge oder ein Farbobjekt • **factor** - eine ganze Zahl, die dem Verdunkelungsfaktor entspricht: – Wenn der Faktor größer als 100 ist, gibt diese Funktion eine dunklere Farbe zurück (z. B. gibt die Einstellung des Faktors auf 200 eine Farbe zurück, die halb so hell ist); – Wenn der Faktor kleiner als 100 ist, wird eine hellere Farbe zurückgegeben, aber es wird empfohlen, zu diesem Zweck die Funktion lighter() zu verwenden; – wenn der Faktor 0 oder negativ ist, ist der Rückgabewert nicht spezifiziert.
Beispiele	• `darker('200,10,30', 200)` → ‚100,5,15,255‘

Siehe auch: *lighter*

lighter

Gibt eine hellere (oder dunklere) Farbe zurück. Der zurückgegebene Typ entspricht den Farbargumenten, d. h. einer Farbzeichenfolge oder einem Farbobjekt.

Syntax	lighter(color, factor)
Argumente	• **color** - eine Farbzeichenfolge oder ein Farbobjekt • **factor** - eine ganze Zahl, die dem Aufhellungsfaktor entspricht: – Wenn der Faktor größer als 100 ist, gibt diese Funktion eine hellere Farbe zurück (z. B. gibt die Einstellung des Faktors auf 150 eine um 50 % hellere Farbe zurück); – Wenn der Faktor kleiner als 100 ist, ist die zurückgegebene Farbe dunkler, aber es wird empfohlen, zu diesem Zweck die Funktion darker() zu verwenden; – wenn der Faktor 0 oder negativ ist, ist der Rückgabewert nicht spezifiziert.
Beispiele	• `lighter('200,10,30', 200)` → ‚255,158,168,255‘

Siehe auch: *darker*

project_color

Gibt eine Farbe aus dem Projekt-Farbschema zurück.

Syntax	project_color(name)
Argumente	• **name** - einen Farbnamen
Beispiele	• `project_color('Logo color')` → ‚20,140,50‘

Siehe auch: *setting project colors*

project_color_object

Gibt eine Farbe aus dem Farbschema des Projekts zurück. Im Gegensatz zu project_color, das eine Farbzeichenfolge zurückgibt, gibt project_color_object ein Farbobjekt zurück.

Syntax	project_color_object(name)
Argumente	• **name** - einen Farbnamen
Beispiele	• `project_color_object('Logo color')` → RGBA: 0.08,0.55,0.20,1.00

ramp_color

Gibt eine Zeichenkette zurück, die eine Farbe aus einer Farbrampe darstellt.

Gespeicherte Rampenvariante

Gibt einen String zurück, der eine Farbe aus einer gespeicherten Rampe darstellt

Syntax	ramp_color(ramp_name, value)
Argumente	• **ramp_name** - der Name der Farbrampe als Zeichenkette, zum Beispiel ‚Spectral‘ • **value** - die Position auf der Rampe, von der aus die Farbe ausgewählt werden soll, als reale Zahl zwischen 0 und 1
Beispiele	• `ramp_color('Spectral',0.3)` → ‚253,190,115,255‘

Bemerkung

Die verfügbaren Farbrampen variieren je nach QGIS-Installation. Diese Funktion liefert möglicherweise nicht die erwarteten Ergebnisse, wenn Sie Ihr QGIS-Projekt zwischen Installationen wechseln.

Durch Expression erzeugte Rampenvariante

Gibt eine Zeichenkette zurück, die eine Farbe aus einer durch einen Ausdruck erstellten Rampe darstellt

Syntax	ramp_color(ramp, value)
Argumente	• **ramp** - die Farbrampe • **value** - die Position auf der Rampe, von der aus die Farbe ausgewählt werden soll, als reale Zahl zwischen 0 und 1
Beispiele	• `ramp_color(create_ramp(map(0,'0,0,0',1,'255,0,0')),1)` → ‚255,0,0,255‘

Siehe auch: *Einstellen einer Farbrampe*, *Die Dropdown- Schaltfläche für Farbverläufe*

ramp_color_object

Gibt ein Farbobjekt aus einer Farbverlaufsrampe zurück. Im Gegensatz zu ramp_color, das eine Zeichenkettendarstellung der Farbe zurückgibt, gibt ramp_color_object ein Farbobjekt zurück.

Gespeicherte Rampenvariante

Gibt ein Farbobjekt aus einer gespeicherten Farbskala zurück.

Syntax	ramp_color_object(ramp_name, value)
Argumente	• **ramp_name** - der Name der Farbrampe als Zeichenkette, zum Beispiel ‚Spectral‘ • **value** - die Position auf der Rampe, von der aus die Farbe ausgewählt werden soll, als reale Zahl zwischen 0 und 1
Beispiele	• `ramp_color_object('Spectral',0.3)` → RGBA: 0.99,0.75,0.45,1.00

Bemerkung

Die verfügbaren Farbrampen variieren je nach QGIS-Installation. Diese Funktion liefert möglicherweise nicht die erwarteten Ergebnisse, wenn Sie Ihr QGIS-Projekt zwischen Installationen wechseln.

Durch Expression erzeugte Rampenvariante

Gibt ein Farbobjekt aus einer durch einen Ausdruck erstellten Farbskala zurück.

Syntax	ramp_color_object(ramp, value)
Argumente	• **ramp** - die Farbrampe • **value** - die Position auf der Rampe, von der aus die Farbe ausgewählt werden soll, als reale Zahl zwischen 0 und 1
Beispiele	• `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

Legt eine bestimmte Farbkomponente für eine Farbkette oder ein Farbobjekt fest, z. B. die Rot-Komponente oder die Alpha-Komponente.

Syntax	set_color_part(color, component, value)
Argumente	• **color** - eine Farbzeichenfolge oder ein Farbobjekt • **component** - eine Zeichenkette, die der zu setzenden Farbkomponente entspricht. Gültige Optionen sind: – red: RGB-Rot-Komponente (0-255) – green: RGB-Grün-Komponente (0-255) – blue: RGB-Blau-Komponente (0-255) – alpha: Alpha-Wert (Transparenz) (0-255) – hue: HSV-Farbton (0-360) – saturation: HSV-Sättigung (0-100) – value: HSV-Wert (0-100) – hsl_hue: HSL-Farbton (0-360) – hsl_saturation: HSL-Sättigung (0-100) – lightness: HSL-Helligkeit (0-100) – cyan: CMYK Cyan-Anteil (0-100) – magenta: CMYK Magenta-Anteil (0-100) – yellow: CMYK Gelb-Anteil (0-100) – black: CMYK Schwarz-Anteil(0-100) • **value** - neuer Wert für die Farbkomponente unter Berücksichtigung der oben aufgeführten Wertebereiche
Beispiele	• `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 Funktionen für Bedingungen

Diese Gruppe enthält Funktionen, um in Ausdrücken mit Bedingen zu arbeiten.

CASE

CASE wird verwendet, um eine Reihe von Bedingungen auszuwerten und ein Ergebnis für die erste erfüllte Bedingung zurückzugeben. Die Bedingungen werden nacheinander abgefragt, und wenn eine Bedingung erfüllt ist, wird die Abfrage beendet und das entsprechende Ergebnis zurückgegeben. Wenn keine der Bedingungen erfüllt ist, wird der Wert in der ELSE-Klausel zurückgegeben. Wenn keine ELSE-Klausel gesetzt ist und keine der Bedingungen erfüllt ist, wird NULL zurückgegeben.

CASE

WHEN *Bedingung* THEN *Ergebnis*

[…n]

[ELSE *Ergebnis*]

END

[ ] markiert optionale Komponenten

Argumente	• **WHEN Bedingung** - ein zu prüfender Ausdruck • **THEN Ergebnis** - wenn die *Bedingung* als Wahr ausgewertet wird, wird das *THEN-Ergebnis* zurückgegeben • **ELSE Ergebnis** - wenn keine der oben genannten *Bedingungen* als wahr bewertet wird, wird das *ELSE-Ergebnis* zurückgegeben
Beispiele	• `CASE WHEN "name" IS NULL THEN 'None' END` → Gibt die Zeichenkette ‚None' zurück, wenn das Feld „name" NULL ist • `CASE WHEN $area > 10000 THEN 'Big property' WHEN $area > 5000 THEN 'Medium property' ELSE 'Small property' END` → Gibt die Zeichenfolge „ Big property" zurück, wenn die Fläche größer als 10000 ist, „Medium property", wenn die Fläche zwischen 5000 und 10000 liegt, und „Small property" für alle anderen

coalesce

Gibt den ersten Nicht-NULL-Wert aus der Ausdrucksliste zurück.

Diese Funktion kann eine beliebige Anzahl von Argumenten annehmen.

Syntax	coalesce(expression1, expression2, …)
Argumente	• **expression** - jeder gültige Ausdruck oder Wert, unabhängig vom Typ.
Beispiele	• `coalesce(NULL, 2)` → 2 • `coalesce(NULL, 2, 3)` → 2 • `coalesce(7, NULL, 3*2)` → 7 • `coalesce("fieldA", "fallbackField", 'ERROR')` → Wert von FeldA, wenn es nicht NULL ist, sonst der Wert von „fallbackField" oder die Zeichenkette ‚ERROR', wenn beide NULL sind

if

Prüft eine Bedingung und gibt je nach der Bedingungsprüfung ein anderes Ergebnis zurück.

Syntax	if(condition, result_when_true, result_when_false)
Argumente	• **condition** - die Bedingung, die überprüft werden soll • **result_when_true** - das Ergebnis, das zurückgegeben wird, wenn die Bedingung erfüllt ist, oder ein anderer Wert, der nicht in „false“ umgewandelt werden kann. • **result_when_false** - das Ergebnis, das zurückgegeben wird, wenn die Bedingung falsch ist, oder ein anderer Wert, der in falsch umgewandelt wird, wie 0 oder ‚‘. NULL wird ebenfalls in „false“ umgewandelt.
Beispiele	• `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)' )` → ‚It is false (empty)‘ • `if( ' ', 'It is true (not empty)', 'It is false (empty)' )` → ‚It is true (not empty)‘ • `if( 0, 'One', 'Zero' )` → ‚Zero‘ • `if( 10, 'One', 'Zero' )` → ‚One‘

nullif

Gibt einen NULL-Wert zurück, wenn value1 gleich value2 ist; andernfalls gibt es value1 zurück. Dies kann zur bedingten Ersetzung von Werten durch NULL verwendet werden.

Syntax	nullif(value1, value2)
Argumente	• **value1** - Der Wert, der entweder verwendet oder durch NULL ersetzt werden soll. • **value2** - Der Kontrollwert, der die NULL-Ersetzung auslösen wird.
Beispiele	• `nullif('(none)', '(none)')` → NULL • `nullif('text', '(none)')` → ‚text‘ • `nullif("name", '')` → NULL, wenn „name“ ein leerer String ist (oder bereits NULL), „name“ in jedem anderen Fall.

regexp_match

Gibt die erste übereinstimmende Position eines regulären Ausdrucks innerhalb einer Unicode-Zeichenkette zurück, oder 0, wenn die Teilzeichenkette nicht gefunden wird.

Syntax	regexp_match(input_string, regex)
Argumente	• **input_string** - die mit dem regulären Ausdruck zu prüfende Zeichenfolge • **regex** - Der reguläre Ausdruck, gegen den getestet werden soll. Backslash-Zeichen müssen doppelt escaped werden (z. B. „\\s“ für ein Leerzeichen oder „\\b“ für eine Wortgrenze).
Beispiele	• `regexp_match('QGIS ROCKS','\\sROCKS')` → 5 • `regexp_match('Budač','udač\\b')` → 2

try

Testet einen Ausdruck und gibt seinen Wert zurück, wenn er fehlerfrei ist. Liefert der Ausdruck einen Fehler, wird ein alternativer Wert zurückgegeben, andernfalls gibt die Funktion NULL zurück.

Syntax	try(expression, [alternative]) [] kennzeichnet optionale Argumente
Argumente	• **expression** - der Ausdruck, der ausgeführt werden soll • **alternative** - das Ergebnis, das zurückgegeben wird, wenn der Ausdruck einen Fehler liefert.
Beispiele	• `try( to_int( '1' ), 0 )` → 1 • `try( to_int( 'a' ), 0 )` → 0 • `try( to_date( 'invalid_date' ) )` → NULL

9.2.5 Umwandlungs-Funktionen

Diese Gruppe enthält Funktionen zur Umwandlung eines Datentyps in einen anderen (z. B. String von/nach Integer, Binär von/nach String, String nach Datum, …).

from_base64

Dekodiert eine Zeichenkette in der Base64-Kodierung in einen Binärwert.

Syntax	from_base64(string)
Argumente	• **string** - die zu dekodierende Zeichenkette
Beispiele	• `from_base64('UUdJUw==')` → ‚QGIS‘

hash

Erzeugt einen Hash aus einer Zeichenkette mit einer festgelegten Methode. Ein Byte (8 Bits) wird durch zwei Hexadezimalziffern dargestellt, sodass „md4“ (16 Bytes) einen 16 * 2 = 32 Zeichen langen Hex-String und „keccak_512“ (64 Bytes) einen 64 * 2 = 128 Zeichen langen Hex-String erzeugt.

Syntax	hash(string, method)
Argumente	• **string** - die Zeichenkette, die einen Hash umgewandelt werden soll • **method** - Die Hash-Methode; gültig sind: ‚md4‘, ‚md5‘, ‚sha1‘, ‚sha224‘, ‚sha384‘, ‚sha512‘, ‚sha3_224‘, ‚sha3_256‘, ‚sha3_384‘, ‚sha3_512‘, ‚keccak_224‘, ‚keccak_256‘, ‚keccak_384‘, ‚keccak_512‘
Beispiele	• `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')` → ‚467f49a5039e7280d5d42fd433e80d203439e338eaabd701f0d6c17d • `hash('QGIS', 'sha3_256')` → ‚540f7354b6b8a6e735f2845250f15f4f3ba4f666c55574d9e9354575 • `hash('QGIS', 'sha3_384')` → ‚96052da1e77679e9a65f60d7ead961b28797782314478638 6eb4364 • `hash('QGIS', 'sha3_512')` → ‚900d079dc69761da113980253aa8ac0414a8bd6d09879a916228f874 • `hash('QGIS', 'keccak_224')` → ‚5b0ce6acef8b0a121d4ac4f3eaa8503c799ad4e26a3392d1fb2014 • `hash('QGIS', 'keccak_256')` → ‚991c520aa6815392de24087f61b2ae0fd56abbfeee4a8ca019c101 • `hash('QGIS', 'keccak_384')` → ‚c57a3aed9d856fa04e5eeee9b62b6e027cca81ba574116d3cc1f0d4

md5

Erzeugt einen md5-Hash aus einer Zeichenkette.

Syntax	md5(string)
Argumente	• **string** - die Zeichenkette, die einen Hash umgewandelt werden soll
Beispiele	• `md5('QGIS')` → ‚57470aaa9e22adaefac7f5f342f1c6da‘

sha256

Erzeugt einen sha256-Hash aus einer Zeichenkette.

Syntax	sha256(string)
Argumente	• **string** - die Zeichenkette, die einen Hash umgewandelt werden soll
Beispiele	• `sha256('QGIS')` → ‚eb045cba7a797aaa06ac58830846e40c8e8c780bc0676d3393605fae50c05309‘

to_base64

Kodiert einen Binärwert in eine Zeichenkette, unter Verwendung der Base64-Kodierung.

Syntax	to_base64(value)
Argumente	• **value** - der zu kodierende Binärwert
Beispiele	• `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.

Syntax	to_bool(value)
Argumente	• **value** - value to convert to boolean
Beispiele	• `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

Konvertiert eine Zeichenkette in ein Datumsobjekt. Ein optionaler Format-String kann angegeben werden, um den String zu parsen; siehe QDate::fromString oder die Dokumentation der Funktion format_date für zusätzliche Dokumentation zum Format. Standardmäßig wird das aktuelle Gebietsschema des QGIS-Benutzers verwendet.

Syntax	to_date(string, [format], [language]) [] kennzeichnet optionale Argumente
Argumente	• **string** - String, der einen Datumswert repräsentiert • **format** - Format für die Umwandlung der Zeichenkette in ein Datum • **language** - Sprache (Kleinbuchstaben, zwei- oder dreibuchstabig, ISO 639 Sprachcode), die für die Umwandlung der Zeichenkette in ein Datum verwendet wird. Standardmäßig wird das aktuelle Gebietsschema des QGIS-Benutzers verwendet.
Beispiele	• `to_date('2012-05-04')` → 2012-05-04 • `to_date('June 29, 2019','MMMM d, yyyy')` → 2019-06-29, wenn das aktuelle Gebietsschema den Namen ‚June“ für den sechsten Monat verwendet, andernfalls tritt ein Fehler auf • `to_date('29 juin, 2019','d MMMM, yyyy','fr')` → 2019-06-29

Siehe auch: *format_date*

to_datetime

Konvertiert eine Zeichenkette in einen DatumZeit-Wert. Optional kann ein gewünschtes Ausgabeformat angegeben werden; siehe QDate::fromString, QTime::fromString oder die Dokumentation der Funktion format_date für zusätzliche Dokumentation zum Format. Standardmäßig wird das aktuell für QGIS eingestellte Gebietsschema verwendet.

Syntax	to_datetime(string, [format], [language]) [] kennzeichnet optionale Argumente
Argumente	• **string** - Zeichenkette, die einen Datumswert darstellt • **format** - Format, in dem der DatumZeit-Wert ausgegeben werden soll • **language** - Sprache (Kleinbuchstaben, zwei- oder dreibuchstabig, ISO 639 Sprachcode), die zur Umwandlung der Zeichenkette in einen DatumZeit-Wert verwendet wird. Standardmäßig wird das aktuell für QGIS eingestellte Gebietsschema verwendet.
Beispiele	• `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, wenn das aktuelle Gebietsschema den Namen ‚June" für den sechsten Monat verwendet, andernfalls wird ein Fehler ausgegeben • `to_datetime('29 juin, 2019 @ 12:34','d MMMM, yyyy @ HH:mm', 'fr')` → 2019-06-29T12:34

Siehe auch: *format_date*

to_decimal

Konvertiert eine Grad-, Minuten- und Sekundenkoordinate in ihr dezimales Äquivalent

Syntax	to_decimal(value)
Argumente	• **value** - Eine Grad-, Minuten- und Sekunden-Zeichenkette
Beispiele	• `to_decimal('6°21\'16.445')` → 6.3545680555

to_dm

Konvertiert eine Koordinate in Grad, Minute

Syntax	to_dm(coordinate, axis, precision, [formatting=]) [] kennzeichnet optionale Argumente
Argumente	• **coordinate** - Ein Wert für den Breitengrad oder Längengrad • **axis** - Die Achse der Koordinate. Entweder ‚x' oder ‚y'. • **precision** - Anzahl der Dezimalen. • **formatting** - Bezeichnet den Formatierungstyp. Zulässige Werte sind NULL (Standard), ‚aligned' oder ‚suffix'.
Beispiele	• `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

Konvertiert eine Koordinate in Grad, Minute, Sekunde.

Syntax	to_dms(coordinate, axis, precision, [formatting=]) [] kennzeichnet optionale Argumente
Argumente	• **coordinate** - Ein Wert für den Breitengrad oder Längengrad • **axis** - Die Achse der Koordinate. Entweder ‚x' oder ‚y'. • **precision** - Anzahl der Dezimalen. • **formatting** - Bezeichnet den Formatierungstyp. Zulässige Werte sind NULL (Standard), ‚aligned' oder ‚suffix'.
Beispiele	• `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

Konvertiert eine Zeichenfolge in eine Ganzzahl. Wenn ein Wert nicht in eine Ganzzahl konvertiert werden kann, ist der Ausdruck ungültig (z. B. ist „123asd" ungültig).

Syntax	to_int(string)
Argumente	• **string** - Zeichenkette, die in eine ganze Zahl umgewandelt werden soll
Beispiele	• `to_int('123')` → 123

to_interval

Konvertiert eine Zeichenkette in einen Intervalltyp. Kann verwendet werden, um Tage, Stunden, Monate usw. eines Datums zu ermitteln

Syntax	to_interval(string)
Argumente	• **string** - eine Zeichenkette, die ein Intervall darstellt. Erlaubte Formate sind {n} Tage {n} Stunden {n} Monate
Beispiele	• `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

Konvertiert eine Zeichenfolge in eine reale Zahl. Wenn ein Wert nicht in eine reale Zahl konvertiert werden kann, ist der Ausdruck ungültig (z. B. ist „123,56asd" ungültig). Zahlen werden nach dem Speichern der Änderungen gerundet, wenn die Genauigkeit geringer ist als das Ergebnis der Konvertierung.

Syntax	to_real(string)
Argumente	• **string** - String zur Umwandlung in eine reelle Zahl
Beispiele	• `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.

Syntax	to_string(number)
Argumente	• **number** - Ganzzahl oder reeller Wert, der in eine Zeichenkette konvertiert werden soll
Beispiele	• `to_string(1.23)` → ‚1.23'

Siehe auch: *format_number*

to_time

Konvertiert eine Zeichenkette in ein Zeitobjekt. Optional kann ein gewünschtes Format für die Ausgabe angegeben werden; siehe QTime::fromString für zusätzliche Dokumentation über das Format

Syntax	to_time(string, [format], [language]) [] kennzeichnet optionale Argumente
Argumente	• **string** - Zeichenkette, die einen Zeitwert darstellt • **format** - Format für die Umwandlung der Zeichenkette in eine Zeitangabe • **language** - Sprache (Kleinbuchstaben, zwei- oder dreibuchstabiger ISO 639-Sprachcode), die zur Umwandlung der Zeichenfolge in eine Zeitangabe verwendet wird
Beispiele	• `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

Siehe auch: *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.

Syntax	crs_from_text(definition)
Argumente	• **definition** - CRS definition
Beispiele	• `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.

Syntax	crs_to_authid(crs)
Argumente	• **crs** - crs value
Beispiele	• `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 Benutzerdefinierte Funktionen

Diese Gruppe enthält vom Nutzer erstellte Funktionen. Unter *Funktions Editor* finden Sie weitere Details.

9.2.8 Datums- und Zeit-Funktionen

Diese Gruppe enthält Funktionen zur Verarbeitung von Datums- und Zeitdaten. Einige Funktionen hat überschneiden sich mit den Gruppen *Umwandlungs-Funktionen* (to_date, to_time, to_datetime, to_interval) und *Zeichenketten-Funktionen* (format_date).

Bemerkung

Speichern von Datum, DatumZeit und Intervallen in Feldern

Die Möglichkeit, *Datum*-, *Uhrzeit*- und *DatumZeit*-Werte direkt in Feldern zu speichern, hängt vom Format der Datenquelle ab (z. B. akzeptiert Shapefile das Format *Datum*, aber nicht das Format *Uhrzeit* oder *DatumZeit*). Im Folgenden finden Sie Vorschläge zur Umgehung dieser Einschränkung:

- *Datum*, *DatumZeit* und *Uhrzeit* können mit der Funktion *format_date()* konvertiert und in Textfeldern gespeichert werden.
- *Intervalle* können in ganzzahligen oder dezimalen Feldern gespeichert werden, nachdem eine der Datumsextraktions-Funktionen verwendet wurde (z. B. *day()*, um das Intervall in Tagen ausgedrückt zu erhalten).

age

Gibt die Differenz zwischen zwei Daten oder DatumZeit-Werten zurück.

Die Differenz wird als `Intervall` zurückgegeben und muss mit einer der folgenden Funktionen verwendet werden, um nützliche Informationen zu liefern:

- `year`
- `month`
- `week`
- `day`
- `hour`
- `minute`
- `second`

Syntax	age(datetime1, datetime2)
Argumente	• **datetime1** - eine Zeichenkette, ein Datum oder eine Zeitangabe, die das spätere Datum darstellt • **datetime2** - eine Zeichenkette, ein Datum oder eine Zeitangabe, die das frühere Datum darstellt
Beispiele	• `day(age('2012-05-12','2012-05-02'))` → 10 • `hour(age('2012-05-12','2012-05-02'))` → 240

datetime_from_epoch

Gibt einen DatumZeit-Wert zurück, deren Datum und Zeit die Anzahl der Millisekunden sind, die seit 1970-01-01T00:00:00.000, Koordinierte Weltzeit (Qt.UTC), vergangen sind und in Qt.LocalTime umgerechnet wurden.

Syntax	datetime_from_epoch(int)
Argumente	• **int** - Anzahl (Millisekunden)
Beispiele	• `datetime_from_epoch(1483225200000)` → 2017-01-01T00:00:00

day

Extrahiert den Tag aus einem Datum, oder die Anzahl der Tage aus einem Intervall.

Datumsvariante

Extrahiert den Tag aus einem Datum oder einem Datumzeit-Wert.

Syntax	day(date)
Argumente	• **date** - ein Datum oder ein DatumZeit-Wert
Beispiele	• `day('2012-05-12')` → 12

Intervall-Variante

Berechnet die Länge eines Intervalls in Tagen.

Syntax	day(interval)
Argumente	• **interval** - Intervallwert, von welchem die Anzahl der Tage zurückgegeben werden soll
Beispiele	• `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

Gibt den Wochentag für ein angegebenes Datum oder einen DatumZeit-Wert zurück. Der zurückgegebene Wert reicht von 0 bis 6, wobei 0 einem Sonntag und 6 einem Samstag entspricht.

Syntax	day_of_week(date)
Argumente	• **date** - Datum oder DatumZeit-Wert
Beispiele	• `day_of_week(to_date('2015-09-21'))` → 1

epoch

Gibt das Intervall in Millisekunden zwischen der Unix-Epoche und einem gegebenen Datumswert zurück.

Syntax	epoch(date)
Argumente	• **date** - ein Datum oder ein DatumZeit-Wert
Beispiele	• `epoch(to_date('2017-01-01'))` → 1483203600000

format_date

Formatiert einen Datumstyp oder einen String in ein benutzerdefiniertes Stringformat. Verwendet Qt Datum/Zeit Format Strings. Siehe QDateTime::toString.

Syntax	format_date(datetime, format, [language]) [] kennzeichnet optionale Argumente
Argumente	• **datetime** - Datum, Uhrzeit oder DatumZeit-Wert • **format** - Formatvorlage, die zur Formatierung der Zeichenkette verwendet werden soll.

Ausdruck	gibt zurück
d	den Tag als Zahl ohne führende Null (1 bis 31)
dd	den Tag als Zahl mit führender Null (01 bis 31)
ddd	den abgekürzten lokalisierten Tagesnamen (z. B. ‚Mon' bis ‚Sun')
dddd	den langen lokalisierten Tagesnamen (z. B. ‚Monday' bis ‚Sunday')
M	den Monat als Zahl ohne führende Null (1-12)
MM	den Monat als Zahl mit führender Null (01-12)
MMM	den abgekürzten lokalisierten Monatsnamen (z. B. ‚Jan' bis ‚Dec')
MMMM	den langen lokalisierten Monatsnamen (z. B. ‚January' bis ‚December')
yy	das Jahr als zweistellige Zahl (00-99)
yyyy	das Jahr als vierstellige Zahl

Diese Ausdrücke können für den Zeitteil des Formatstrings verwendet werden:

Ausdruck	gibt zurück
h	die Stunde ohne führende Null (0 bis 23 oder 1 bis 12 bei AM/PM-Anzeige)
hh	die Stunde mit führender Null (00 bis 23 oder 01 bis 12 bei AM/PM-Anzeige)
H	die Stunde ohne führende Null (0 bis 23, auch bei AM/PM-Anzeige)
HH	die Stunde mit einer führenden Null (00 bis 23, auch bei AM/PM-Anzeige)
m	die Minute ohne führende Null (0 bis 59)
mm	die Minute mit führender Null (00 bis 59))
s	die Sekunde ohne führende Null (0 bis 59)
ss	die Sekunde mit führender Null (00 bis 59)
z	die Millisekunden ohne nachgestellte Nullen (0 bis 999)
zzz	die Millisekunden mit nachgestellten Nullen (000 bis 999)
AP oder A	als AM/PM-Zeit interpretieren. *AP* muss entweder ‚AM' oder ‚PM' sein.
ap oder a	Interpretation als AM/PM-Zeit. *ap* muss entweder ‚am' oder ‚pm' sein.

- **language** - Sprache (Kleinbuchstaben, zwei- oder dreibuchstabig, ISO 639 Sprachcode), die verwendet wird, um das Datum in eine benutzerdefinierte Zeichenfolge zu formatieren. Standardmäßig wird das aktuelle Gebietsschema des QGIS-Benutzers verwendet.

Beispiele

- `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', wenn das aktuelle Gebietsschema eine englische Variante ist
- `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

Extrahiert den Stundenanteil aus einem DatumZeit-Wert oder einer Uhrzeit oder die Anzahl der Stunden aus einem Intervall.

Zeit-Variante

Extrahiert den Stundenteil aus einer Zeit oder einem DatumZeit-Wert.

Syntax	hour(datetime)
Argumente	• **datetime** - einen Zeit- oder DatumZeit-Wert
Beispiele	• `hour( to_datetime('2012-07-22 13:24:57') )` → 13

Intervall-Variante

Berechnet die Länge eines Intervalls in Stunden.

Syntax	hour(interval)
Argumente	• **interval** - Intervallwert, von welchem die Anzahl der Stunden zurückgegeben werden soll
Beispiele	• `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

Erzeugt einen Datumswert aus Jahr, Monat und Tag.

Syntax	make_date(year, month, day)
Argumente	• **year** - Jahreszahl. Die Jahre 1 bis 99 werden als solche interpretiert. Jahr 0 ist ungültig. • **month** - Monatsnummer, wobei 1=Januar entspricht • **day** - Tagesnummer, beginnend mit 1 für den ersten Tag des Monats
Beispiele	• `make_date(2020,5,4)` → 2020-05-04

make_datetime

Erzeugt einen DatumZeit-Wert aus den Zahlen Jahr, Monat, Tag, Stunde, Minute und Sekunde.

Syntax	make_datetime(year, month, day, hour, minute, second)
Argumente	• **year** - Jahreszahl. Die Jahre 1 bis 99 werden als solche interpretiert. Jahr 0 ist ungültig. • **month** - Monatsnummer, wobei 1=Januar entspricht • **day** - Tagesnummer, beginnend mit 1 für den ersten Tag des Monats • **hour** - Stunden • **minute** - Minuten • **second** - Sekunden (Nachkommastellen für Millisekunden können angegeben werden)
Beispiele	• `make_datetime(2020,5,4,13,45,30.5)` → 2020-05-04 13:45:30.500

make_interval

Erstellt einen Intervallwert aus den Werten Jahr, Monat, Wochen, Tage, Stunden, Minuten und Sekunden.

Syntax	make_interval([years=0], [months=0], [weeks=0], [days=0], [hours=0], [minutes=0], [seconds=0]) [] kennzeichnet optionale Argumente
Argumente	• **years** - Anzahl der Jahre (erwartet eine Jahreslänge von 365,25 Tagen). • **months** - Anzahl der Monate (erwartet eine Monatslänge von 30 Tagen) • **weeks** - Anzahl der Wochen • **days** - Anzahl der Tage • **hours** - Anzahl der Stunden • **minutes** - Anzahl der Minuten • **seconds** - Anzahl der Sekunden
Beispiele	• `make_interval(hours:=3)` → 3 Stunden • `make_interval(days:=2, hours:=3)` → 2,125 Tage • `make_interval(minutes:=0.5, seconds:=5)` → 35 Sekunden

make_time

Erzeugt einen Zeitwert aus Stunden-, Minuten- und Sekundenzahlen.

Syntax	make_time(hour, minute, second)
Argumente	• **hour** - Stunden • **minute** - Minuten • **second** - Sekunden (Nachkommastellen für Millisekunden können angegeben werden)
Beispiele	• `make_time(13,45,30.5)` → 13:45:30.500

minute

Extrahiert den Minutenanteil aus einem DatumZeit-Wert oder einer Uhrzeit oder die Anzahl der Minuten aus einem Intervall.

Zeit-Variante

Extrahiert den Minutenteil aus einer Zeit oder einem DatumZeit-Wert.

Syntax	minute(datetime)
Argumente	• **datetime** - einen Zeit- oder DatumZeit-Wert
Beispiele	• `minute( to_datetime('2012-07-22 13:24:57') )` → 24

Intervall-Variante

Berechnet die Länge eines Intervalls in Minuten.

Syntax	minute(interval)
Argumente	• **interval** - Intervallwert, von welchem die Anzahl der Minuten zurückgegeben werden soll
Beispiele	• `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

Extrahiert den Monatsteil aus einem Datum oder die Anzahl der Monate aus einem Intervall.

Datumsvariante

Extrahiert den Monatsteil aus einem Datum oder DatumZeit-Wert.

Syntax	month(date)
Argumente	• **date** - ein Datum oder ein DatumZeit-Wert
Beispiele	• `month('2012-05-12')` → 05

Intervall-Variante

Berechnet die Länge eines Intervalls in Monaten.

Syntax	month(interval)
Argumente	• **interval** - Intervallwert, von welchem die Anzahl der Monate zurückgegeben werden soll
Beispiele	• `month(to_interval('3 months'))` → 3 • `month(age('2012-01-01','2010-01-01'))` → 4.03333

now

Gibt das aktuelle Datum und die Uhrzeit zurück. Die Funktion ist statisch und liefert bei der Auswertung konsistente Ergebnisse. Die zurückgegebene Zeit ist die Zeit, zu der der Ausdruck erstellt wurde.

Syntax	now()
Beispiele	• `now()` → 2012-07-22T13:24:57

second

Extrahiert den Sekundenanteil aus einem DatumZeit-Wert oder einer Uhrzeit bzw. die Anzahl der Sekunden aus einem Intervall.

Zeit-Variante

Extrahiert den Sekundenanteil aus einer Zeit oder einem DatumZeit-Wert.

Syntax	second(datetime)
Argumente	• **datetime** - einen Zeit- oder DatumZeit-Wert
Beispiele	• `second( to_datetime('2012-07-22 13:24:57') )` → 57

Intervall-Variante

Berechnet die Länge eines Intervalls in Sekunden.

Syntax	second(interval)
Argumente	• **interval** - Intervallwert, von welchem die Anzahl der Sekunden zurückgegeben werden soll
Beispiele	• `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

Konvertiert eine Zeichenkette in ein Datumsobjekt. Ein optionaler Format-String kann angegeben werden, um den String zu parsen; siehe QDate::fromString oder die Dokumentation der Funktion format_date für zusätzliche Dokumentation zum Format. Standardmäßig wird das aktuelle Gebietsschema des QGIS-Benutzers verwendet.

Syntax	to_date(string, [format], [language]) [] kennzeichnet optionale Argumente
Argumente	• **string** - String, der einen Datumswert repräsentiert • **format** - Format für die Umwandlung der Zeichenkette in ein Datum • **language** - Sprache (Kleinbuchstaben, zwei- oder dreibuchstabig, ISO 639 Sprachcode), die für die Umwandlung der Zeichenkette in ein Datum verwendet wird. Standardmäßig wird das aktuelle Gebietsschema des QGIS-Benutzers verwendet.
Beispiele	• `to_date('2012-05-04')` → 2012-05-04 • `to_date('June 29, 2019','MMMM d, yyyy')` → 2019-06-29, wenn das aktuelle Gebietsschema den Namen ‚June" für den sechsten Monat verwendet, andernfalls tritt ein Fehler auf • `to_date('29 juin, 2019','d MMMM, yyyy','fr')` → 2019-06-29

Siehe auch: *format_date*

to_datetime

Konvertiert eine Zeichenkette in einen DatumZeit-Wert. Optional kann ein gewünschtes Ausgabeformat angegeben werden; siehe QDate::fromString, QTime::fromString oder die Dokumentation der Funktion format_date für zusätzliche Dokumentation zum Format. Standardmäßig wird das aktuell für QGIS eingestellte Gebietsschema verwendet.

Syntax	to_datetime(string, [format], [language]) [] kennzeichnet optionale Argumente
Argumente	• **string** - Zeichenkette, die einen Datumswert darstellt • **format** - Format, in dem der DatumZeit-Wert ausgegeben werden soll • **language** - Sprache (Kleinbuchstaben, zwei- oder dreibuchstabig, ISO 639 Sprachcode), die zur Umwandlung der Zeichenkette in einen DatumZeit-Wert verwendet wird. Standardmäßig wird das aktuell für QGIS eingestellte Gebietsschema verwendet.
Beispiele	• `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, wenn das aktuelle Gebietsschema den Namen ‚June" für den sechsten Monat verwendet, andernfalls wird ein Fehler ausgegeben • `to_datetime('29 juin, 2019 @ 12:34','d MMMM, yyyy @ HH:mm', 'fr')` → 2019-06-29T12:34

Siehe auch: *format_date*

to_interval

Konvertiert eine Zeichenkette in einen Intervalltyp. Kann verwendet werden, um Tage, Stunden, Monate usw. eines Datums zu ermitteln

Syntax	to_interval(string)
Argumente	• **string** - eine Zeichenkette, die ein Intervall darstellt. Erlaubte Formate sind {n} Tage {n} Stunden {n} Monate
Beispiele	• `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

Konvertiert eine Zeichenkette in ein Zeitobjekt. Optional kann ein gewünschtes Format für die Ausgabe angegeben werden; siehe QTime::fromString für zusätzliche Dokumentation über das Format

Syntax	to_time(string, [format], [language]) [] kennzeichnet optionale Argumente
Argumente	• **string** - Zeichenkette, die einen Zeitwert darstellt • **format** - Format für die Umwandlung der Zeichenkette in eine Zeitangabe • **language** - Sprache (Kleinbuchstaben, zwei- oder dreibuchstabiger ISO 639-Sprachcode), die zur Umwandlung der Zeichenfolge in eine Zeitangabe verwendet wird
Beispiele	• `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

Siehe auch: *format_date*

week

Extrahiert die Wochennummer aus einem Datum oder die Anzahl der Wochen aus einem Intervall.

Datumsvariante

Extrahiert die Wochennummer aus einem Datum oder DatumZeit-Wert.

Syntax	week(date)
Argumente	• **date** - ein Datum oder ein DatumZeit-Wert
Beispiele	• `week('2012-05-12')` → 19

Intervall-Variante

Berechnet die Länge eines Intervalls in Wochen.

Syntax	week(interval)
Argumente	• **interval** - Intervallwert, von welchem die Anzahl der Monate zurückgegeben werden soll
Beispiele	• `week(to_interval('3 weeks'))` → 3 • `week(age('2012-01-01','2010-01-01'))` → 104.285

year

Extrahiert den Jahresanteil aus einem Datum oder die Anzahl der Jahre aus einem Intervall.

Datumsvariante

Extrahiert den Jahresanteil aus einem Datum oder DatumZeit-Wert.

Syntax	year(date)
Argumente	• **date** - ein Datum oder ein DatumZeit-Wert
Beispiele	• `year('2012-05-12')` → 2012

Intervall-Variante

Berechnet die Länge eines Intervalls in Jahren.

Syntax	year(interval)
Argumente	• **interval** - Intervallwert, von welchem die Anzahl der Jahre zurückgegeben werden soll
Beispiele	• `year(to_interval('3 years'))` → 3 • `year(age('2012-01-01','2010-01-01'))` → 1.9986

Beispiele:

Mit diesen Funktionen können Sie auch Daten, DatumZeit-Werte oder Zeiten mit dem Operator − (Minus) subtrahieren, um ein Intervall zu erhalten.

Die Addition oder Subtraktion von Zeitintervallen zu Datums- oder Zeitangaben mit Hilfe des + (plus) oder − (minus) Operators gibt eine Datums- oder Zeitangabe zurück.

- Ausgabe der Anzahl der Tage bis zur Herausgabe von QGIS 3.0:

```
to_date('2017-09-29') - to_date(now())
-- Returns <interval: 203 days>
```

- Dasselbe gilt für die Zeit:

```
to_datetime('2017-09-29 12:00:00') - now()
-- Returns <interval: 202.49 days>
```

- Ausgabe der Datums- und Zeitangabe für 100 Tage in der Zukunft:

```
now() + to_interval('100 days')
-- Returns <datetime: 2017-06-18 01:00:00>
```

9.2.9 Felder und Werte

Enthält eine Liste von Feldern des aktiven Layers und spezielle Werte. Die Feldliste enthält die im Datensatz gespeicherten virtuellen Felder und die Felder aus *Verknüpfungen*.

Doppelklick auf einen Feldnamen fügt das Feld im Ausdruck ein. Man kann auch den Feldnamen (vorzugsweise in doppelten Anführungszeichen) oder seinen *Alias* verwenden.

Um sich die Feldwerte, die man in einem Ausdruck verwenden möchte, anzuzeigen zu lassen, kann man im Vorschau-Widget auf *10 Stichproben* oder *Alle eindeutigen* klicken. Die angeforderten Werte werden dann angezeigt und man kann das Ergebnis mithilfe des *Suche* Kastens über der Liste filtern. Beispielwerte können außerdem durch einen Klick mit der rechten Maustaste auf ein Feld angezeigt werden.

Um einen Wert in den erstellten Ausdruck zu übernehmen, klickt man doppelt auf den Wert in der Liste. Wenn der Wert eine Zeichenkette ist, sollte er in einfachen Anführungszeichen geschrieben werden, ansonsten sind keine Anführungszeichen erforderlich.

NULL

Entspricht einem NULL-Wert.

Syntax	NULL
Beispiele	• `NULL` → einen NULL-Wert

Bemerkung

Um auf NULL zu testen, verwenden Sie einen *IS NULL* oder *IS NOT NULL* Ausdruck.

9.2.10 Funktionen für Dateien und Pfade

Diese Gruppe enthält Funktionen zur Bearbeitung von Datei- und Pfadnamen.

base_file_name

Gibt den Basisnamen der Datei ohne Verzeichnis oder Dateisuffix zurück.

Syntax	base_file_name(path)
Argumente	• **path** - ein Dateipfad oder ein Karten-Layer. Wird ein Karten-Layer angegeben, wird die Dateiquelle des Layers verwendet.
Beispiele	• `base_file_name('/home/qgis/data/country_boundaries.shp')` → ‚country_boundaries‘

exif

Ruft die Exif-Tag-Werte einer Bilddatei ab.

Syntax	exif(path, [tag]) [] kennzeichnet optionale Argumente
Argumente	• **path** - Pfad einer Bilddatei oder ein Karten-Layer. Wird ein Karten-Layer angegeben, wird die Dateiquelle des Layers verwendet. • **tag** - Das zurückzugebende Tag. Wenn leer, wird eine Tabelle mit allen Exif-Tag-Werten zurückgegeben.
Beispiele	• `exif('/my/photo.jpg','Exif.Image.Orientation')` → 0

file_exists

Gibt TRUE zurück, wenn der Dateipfad existiert.

Syntax	file_exists(path)
Argumente	• **path** - ein Dateipfad oder ein Karten-Layer. Wird ein Karten-Layer angegeben, wird die Dateiquelle des Layers verwendet.
Beispiele	• `file_exists('/home/qgis/data/country_boundaries.shp')` → TRUE

file_name

Liefert den Namen einer Datei (einschließlich der Dateierweiterung), ohne das Verzeichnis.

Syntax	file_name(path)
Argumente	• **path** - ein Dateipfad oder ein Karten-Layer. Wird ein Karten-Layer angegeben, wird die Dateiquelle des Layers verwendet.
Beispiele	• `file_name('/home/qgis/data/country_boundaries.shp')` → ‚country_boundaries.shp‘

file_path

Gibt die Verzeichniskomponente eines Dateipfads zurück. Dies beinhaltet nicht den Dateinamen.

Syntax	file_path(path)
Argumente	• **path** - ein Dateipfad oder ein Karten-Layer. Wird ein Karten-Layer angegeben, wird die Dateiquelle des Layers verwendet.
Beispiele	• `file_path('/home/qgis/data/country_boundaries.shp')` → ‚/home/qgis/data‘

file_size

Gibt die Größe (in Bytes) einer Datei zurück.

Syntax	file_size(path)
Argumente	• **path** - ein Dateipfad oder ein Karten-Layer. Wird ein Karten-Layer angegeben, wird die Dateiquelle des Layers verwendet.
Beispiele	• `file_size('/home/qgis/data/country_boundaries.geojson')` → 5674

file_suffix

Gibt das Dateisuffix (Erweiterung) eines Dateipfads zurück.

Syntax	file_suffix(path)
Argumente	• **path** - ein Dateipfad oder ein Karten-Layer. Wird ein Karten-Layer angegeben, wird die Dateiquelle des Layers verwendet.
Beispiele	• `file_suffix('/home/qgis/data/country_boundaries.shp')` → ‚shp‘

is_directory

Gibt TRUE zurück, wenn ein Pfad einem Verzeichnis entspricht.

Syntax	is_directory(path)
Argumente	• **path** - ein Dateipfad oder ein Karten-Layer. Wird ein Karten-Layer angegeben, wird die Dateiquelle des Layers verwendet.
Beispiele	• `is_directory('/home/qgis/data/country_boundaries.shp')` → FALSE • `is_directory('/home/qgis/data/')` → TRUE

is_file

Gibt TRUE zurück, wenn ein Pfad mit einer Datei übereinstimmt.

Syntax	is_file(path)
Argumente	• **path** - ein Dateipfad oder ein Karten-Layer. Wird ein Karten-Layer angegeben, wird die Dateiquelle des Layers verwendet.
Beispiele	• `is_file('/home/qgis/data/country_boundaries.shp')` → TRUE • `is_file('/home/qgis/data/')` → FALSE

9.2.11 Formular-Funktionen

Diese Gruppe enthält Funktionen, die ausschließlich im Kontext des Attributformulars funktionieren. Zum Beispiel in den *Feld-Einstellungen*.

current_parent_value

Diese Funktion ist nur im Kontext eines eingebetteten Formulars verwendbar und gibt den aktuellen, nicht gespeicherten Wert eines Feldes im übergeordneten Formular zurück, das gerade bearbeitet wird. Dieser Wert unterscheidet sich von den tatsächlichen Attributwerten des übergeordneten Objekts, die gerade bearbeitet werden oder noch nicht zu einem übergeordneten Layer hinzugefügt wurden. Wenn diese Funktion in einem Filterausdruck eines Wert-Beziehungs-Widgets verwendet wird, sollte sie in ein ‚coalesce()' eingeschlossen werden, das das tatsächliche übergeordnete Objekt vom Layer abrufen kann, wenn das Formular nicht in einem eingebetteten Kontext verwendet wird.

Syntax	current_parent_value(field_name)
Argumente	• **field_name** - ein Feldname im aktuellen übergeordneten Formular
Beispiele	• `current_parent_value( 'FIELD_NAME' )` → Der aktuelle Wert des Feldes ‚FIELD_NAME' im übergeordneten Formular.

current_value

Gibt den aktuellen, nicht gespeicherten Wert eines Feldes im Formular oder in der Tabellenzeile zurück, die gerade bearbeitet wird. Dieser Wert unterscheidet sich von den tatsächlichen Attributwerten des Objekts bei Objekten, die gerade bearbeitet werden oder noch nicht zu einem Layer hinzugefügt wurden.

Syntax	current_value(field_name)
Argumente	• **field_name** - ein Feldname in der aktuellen Formular- oder Tabellenzeile
Beispiele	• `current_value( 'FIELD_NAME' )` → Der aktuelle Wert des Feldes ‚FIELD_NAME'.

9.2.12 Unscharfer Vergleich Funktion

Diese Gruppe enthält Funktionen für unscharfe Vergleiche (Fuzzy) zwischen Werten.

hamming_distance

Gibt die Hamming-Distanz zwischen zwei Zeichenketten zurück. Dies entspricht der Anzahl der Zeichen an den entsprechenden Positionen in den Eingabezeichenketten, bei denen die Zeichen unterschiedlich sind. Die Eingabestrings müssen gleich lang sein, und beim Vergleich wird die Groß- und Kleinschreibung beachtet.

Syntax	hamming_distance(string1, string2)
Argumente	• **string1** - eine Zeichenkette • **string2** - eine zweite Zeichenkette
Beispiele	• `hamming_distance('abc','xec')` → 2 • `hamming_distance('abc','ABc')` → 2 • `hamming_distance(upper('abc'),upper('ABC'))` → 0 • `hamming_distance('abc','abcd')` → NULL

levenshtein

Gibt die Levenshtein-Edit-Distanz zwischen zwei Zeichenketten zurück. Dies entspricht der Mindestanzahl von Zeichenänderungen (Einfügungen, Löschungen oder Ersetzungen), die erforderlich sind, um eine Zeichenkette in eine andere zu ändern.

Der Levenshtein-Abstand ist ein Maß für die Ähnlichkeit zwischen zwei Zeichenketten. Kleinere Abstände bedeuten, dass sich die Zeichenfolgen ähnlicher sind, und größere Abstände bedeuten, dass sich die Zeichenfolgen stärker unterscheiden. Der Abstand ist abhängig von der Groß- und Kleinschreibung.

Syntax	levenshtein(string1, string2)
Argumente	• **string1** - eine Zeichenkette • **string2** - eine zweite Zeichenkette
Beispiele	• `levenshtein('kittens','mitten')` → 2 • `levenshtein('Kitten','kitten')` → 1 • `levenshtein(upper('Kitten'),upper('kitten'))` → 0

longest_common_substring

Gibt die längste gemeinsame Teilzeichenkette zwischen zwei Zeichenketten zurück. Diese Teilzeichenkette ist die längste Zeichenkette, die eine Teilzeichenkette der beiden Eingabezeichenketten ist. Zum Beispiel ist die längste gemeinsame Teilzeichenkette von „ABABC“ und „BABCA“ „BABC“. Bei der Teilzeichenkette wird zwischen Groß- und Kleinschreibung unterschieden.

Syntax	longest_common_substring(string1, string2)
Argumente	• **string1** - eine Zeichenkette • **string2** - eine zweite Zeichenkette
Beispiele	• `longest_common_substring('ABABC','BABCA')` → ‚BABC‘ • `longest_common_substring('abcDeF','abcdef')` → ‚abc‘ • `longest_common_substring(upper('abcDeF'),upper('abcdex'))` → ‚ABCDE‘

soundex

Gibt die Soundex-Darstellung einer Zeichenkette zurück. Soundex ist ein phonetischer Zuordnungsalgorithmus, so dass Zeichenfolgen mit ähnlichen Klängen durch denselben Soundex-Code dargestellt werden sollten.

Syntax	soundex(string)
Argumente	• **string** - eine Zeichenkette
Beispiele	• `soundex('robert')` → ‚R163‘ • `soundex('rupert')` → ‚R163‘ • `soundex('rubin')` → ‚R150‘

9.2.13 Allgemeine Funktionen

Diese Gruppe enthält allgemeine Funktionen.

env

Ruft eine Umgebungsvariable ab und gibt ihren Inhalt als String zurück. Wenn die Variable nicht gefunden wird, wird NULL zurückgegeben. Dies ist praktisch, um systemspezifische Konfigurationen wie Laufwerksbuchstaben oder Pfadpräfixe einzubringen. Die Definition von Umgebungsvariablen hängt vom Betriebssystem ab. Bitte erkundigen Sie sich bei Ihrem Systemadministrator oder in der Dokumentation des Betriebssystems, wie dies eingestellt werden kann.

Syntax	env(name)
Argumente	• **name** - Der Name der Umgebungsvariable, die abgerufen werden soll.
Beispiele	• `env( 'LANG' )` → ‚en_US.UTF-8‘ • `env( 'MY_OWN_PREFIX_VAR' )` → ‚Z:‘ • `env( 'I_DO_NOT_EXIST' )` → NULL

eval

Wertet einen Ausdruck aus, der in einer Zeichenkette übergeben wird. Nützlich zur Erweiterung dynamischer Parameter, die als Kontextvariablen oder Felder übergeben werden.

Syntax	eval(expression)
Argumente	• **expression** - ein Ausdrucks-String
Beispiele	• `eval('\'nice\'')` → ‚nice‘ • `eval(@expression_var)` → [was auch immer das Ergebnis der Auswertung von @expression_var sein mag…]

eval_template

Wertet eine Vorlage aus, die in einer Zeichenkette übergeben wird. Nützlich zur Erweiterung dynamischer Parameter, die als Kontextvariablen oder Felder übergeben werden.

Syntax	eval_template(template)
Argumente	• **template** - eine Zeichenkette als Vorlage
Beispiele	• `eval_template('QGIS [% upper(\'rocks\') %]')` → QGIS ROCKS

is_layer_visible

Gibt TRUE zurück, wenn ein angegebener Layer sichtbar ist.

Syntax	is_layer_visible(layer)
Argumente	• **layer** - eine Zeichenkette, die entweder einen Layernamen oder eine Layer-ID darstellt
Beispiele	• `is_layer_visible('baseraster')` → TRUE

mime_type

Gibt den Mime-Typ der Binärdaten zurück.

Syntax	mime_type(bytes)
Argumente	• **bytes** - die binären Daten
Beispiele	• `mime_type('<html><body></body></html>')` → text/html • `mime_type(from_base64('R0lGODlhAQABAAAAACH5BAEKAAEALAAAAAABAAEAAAIAOw==` → image/gif

var

Gibt den in einer bestimmten Variablen gespeicherten Wert zurück.

Syntax	var(name)
Argumente	• **name** - einen Variablennamen
Beispiele	• `var('qgis_version')` → ‚2.12‘

Siehe auch: Liste der standardmäßigen *Variablen*

with_variable

Diese Funktion setzt eine Variable für jeden Ausdruckscode, der als drittes Argument angegeben wird. Dies ist nur für komplizierte Ausdrücke nützlich, bei denen derselbe berechnete Wert an verschiedenen Stellen verwendet werden muss.

Syntax	with_variable(name, value, expression)
Argumente	• **name** - der Name der zu setzenden Variablen • **value** - der zu setzende Wert • **expression** - der Ausdruck, für den die Variable verfügbar sein wird
Beispiele	• `with_variable('my_sum', 1 + 2 + 3, @my_sum * 2 + @my_sum * 5)` → 42

9.2.14 Geometriefunktionen

Diese Gruppe enthält Funktionen, die mit Geometrieobjekten arbeiten (z.B. buffer, transform, $area).

affine_transform

Gibt die Geometrie nach einer affinen Transformation zurück. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie. Die Operationen werden in der Reihenfolge Skalierung, Rotation, Translation durchgeführt. Wenn es einen Z- oder M-Offset gibt, die Koordinate aber nicht in der Geometrie vorhanden ist, wird sie hinzugefügt.

Syntax	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]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **delta_x** - x-Achsen Translation • **delta_y** - y-Achsen Translation • **rotation_z** - Drehung um die z-Achse in Grad gegen den Uhrzeigersinn • **scale_x** - Skalierungsfaktor der x-Achse • **scale_y** - Skalierungsfaktor der y-Achse • **delta_z** - z-Achsen Translation • **delta_m** - m-Achsen Translation • **scale_z** - Skalierungsfaktor der z-Achse • **scale_m** - Skalierungsfaktor der m-Achse
Beispiele	• `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)‘

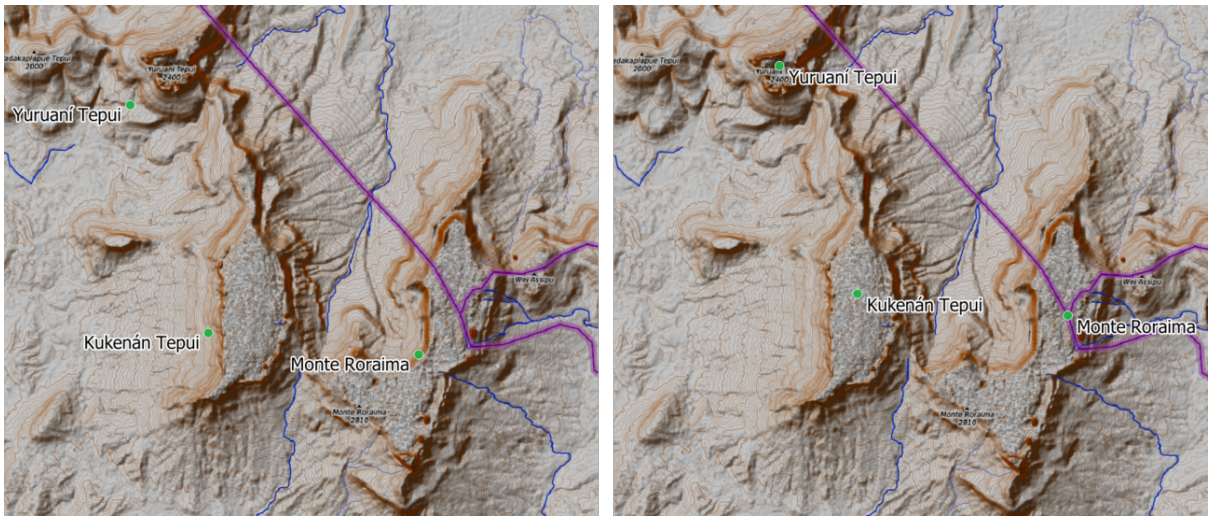


Abb. 9.4: Vektor-Punkt-Layer (grüne Punkte) vor (links) und nach (rechts) einer affinen Transformation (Translation).

angle_at_vertex

Gibt den Winkel der Winkelhalbierenden (durchschnittlicher Winkel) zur Geometrie für einen angegebenen Scheitelpunkt in einer Liniengeometrie zurück. Die Winkel sind in Grad im Uhrzeigersinn von Norden aus angegeben..

Syntax	angle_at_vertex(geometry, vertex)
Argumente	• **geometry** - eine Linienstring-Geometrie • **vertex** - Knotenpunktindex, beginnend bei 0; wenn der Wert negativ ist, entspricht der ausgewählte Knotenpunktindex der Gesamtzahl abzüglich des absoluten Wertes
Beispiele	• `angle_at_vertex(geometry:=geom_from_wkt('LineString(0 0, 10 0, 10 10)'),vertex:=1)` → 45.0

apply_dash_pattern

Wendet ein Strichmuster auf eine Geometrie an und gibt eine MultiLineString-Geometrie zurück, die die Eingabegeometrie ist, die entlang jeder Linie/jedes Rings mit dem angegebenen Muster gestrichen ist.

Syntax	apply_dash_pattern(geometry, pattern, [start_rule=no_rule], [end_rule=no_rule], [adjustment=both], [pattern_offset=0]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie (akzeptiert (mehrere) Linienstrings oder (mehrere) Polygone). • **pattern** - Bindestrichmuster als Array von Zahlen, die die Länge von Bindestrichen und Lücken darstellen. Es muss eine gerade Anzahl von Elementen enthalten. • **start_rule** - optionale Regel zur Einschränkung des Beginns des Musters. Gültige Werte sind ‚no_rule', ‚full_dash', ‚half_dash', ‚full_gap', ‚half_gap'. • **end_rule** - optionale Regel zur Einschränkung des Endes des Musters. Gültige Werte sind ‚no_rule', ‚full_dash', ‚half_dash', ‚full_gap', ‚half_gap'. • **adjustment** - optionale Regel zur Angabe, welcher Teil von Mustern an die gewünschten Musterregeln angepasst wird. Gültige Werte sind ‚both', ‚dash', ‚gap'. • **pattern_offset** - Optionaler Abstand, der einen bestimmten Abstand entlang des Musters angibt, mit dem begonnen werden soll.
Beispiele	• `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

Gibt die Fläche des aktuellen Objekts zurück. Die von dieser Funktion berechnete Fläche berücksichtigt sowohl die Ellipsoid- als auch die Flächeneinheit-Einstellungen des aktuellen Projekts. Wenn zum Beispiel ein Ellipsoid für das Projekt festgelegt wurde, ist die berechnete Fläche ellipsoidisch, und wenn kein Ellipsoid festgelegt wurde, ist die berechnete Fläche planimetrisch.

Syntax	$area
Beispiele	• `$area` → 42

area

Gibt die Fläche eines Geometrie-Polygon-Objekts zurück. Die Berechnungen erfolgen immer planimetrisch im räumlichen Bezugssystem (SRS) dieser Geometrie, und die Einheiten der zurückgegebenen Fläche entsprechen den Einheiten für das SRS. Dies unterscheidet sich von den Berechnungen, die von der Funktion $area durchgeführt werden, die ellipsoidische Berechnungen auf der Grundlage der Einstellungen für Ellipsoid und Flächeneinheit des Projekts vornimmt.

Syntax	area(geometry)
Argumente	• **geometry** - Polygon-Geometrie-Objekt
Beispiele	• `area(geom_from_wkt('POLYGON((0 0, 4 0, 4 2, 0 2, 0 0))'))` → 8.0

azimuth

Liefert den Azimut nach Norden als Winkel in Bogenmaß, gemessen im Uhrzeigersinn von der Senkrechten auf Punkt_a bis Punkt_b.

Syntax	azimuth(point_a, point_b)
Argumente	• **point_a** - Punktgeometrie • **point_b** - Punktgeometrie
Beispiele	• `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

Gibt die nördliche Peilung als Winkel im Bogenmaß zurück, gemessen im Uhrzeigersinn auf dem Ellipsoid von der Senkrechten auf Punkt_a bis Punkt_b.

Syntax	bearing(point_a, point_b, [source_crs], [ellipsoid]) [] kennzeichnet optionale Argumente
Argumente	• **point_a** - Punktgeometrie • **point_b** - Punktgeometrie • **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.
Beispiele	• `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

Gibt die Schließung der kombinatorischen Begrenzung der Geometrie zurück (d. h. die topologische Begrenzung der Geometrie). Zum Beispiel hat eine Polygongeometrie eine Begrenzung, die aus den Linienzügen für jeden Ring im Polygon besteht. Einige Geometrietypen haben keine definierte Begrenzung, z. B. Punkte oder Geometriesammlungen, und geben NULL zurück.

Syntax	boundary(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `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))'

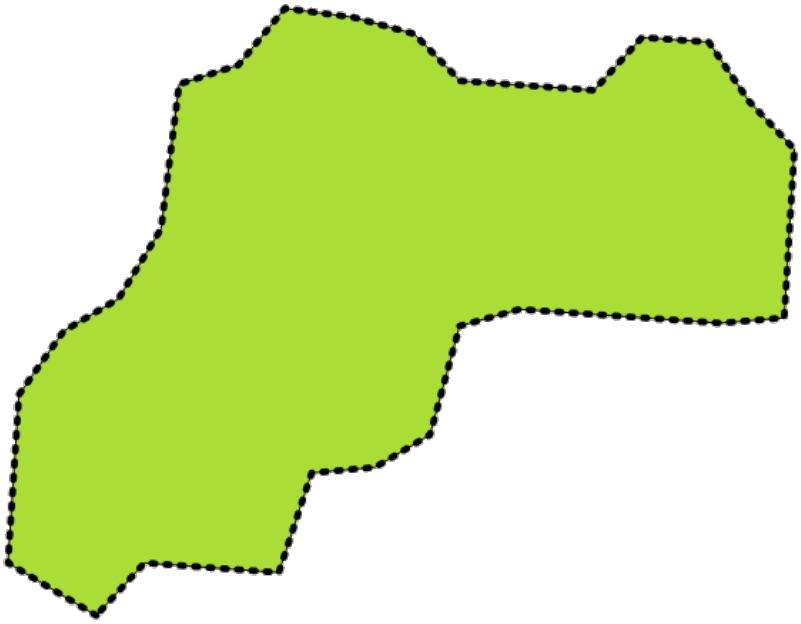


Abb. 9.5: Grenze (schwarze gestrichelte Linie) des Quell-Polygon-Layers

Siehe auch: *Boundary*-Algorithmus

bounds

Gibt eine Geometrie zurück, die den Begrenzungsrahmen einer Eingabegeometrie darstellt. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	bounds(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `bounds(@geometry)` → Begrenzungsrahmen der Geometrie des aktuellen Objekts • `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))‘

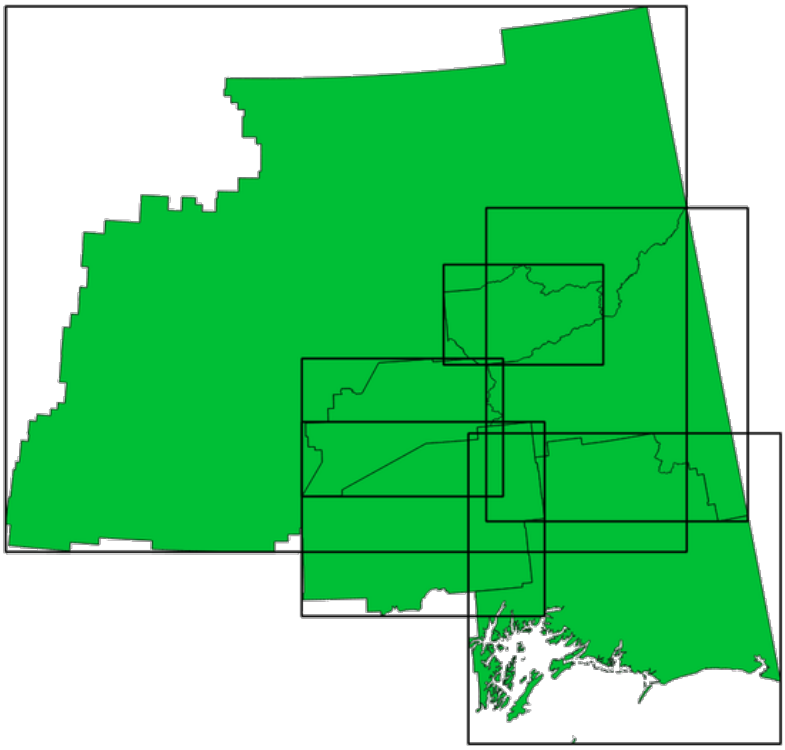


Abb. 9.6: Schwarze Linien stellen die Begrenzungslinien der einzelnen Polygonobjekte dar

Siehe auch: *Bounding boxes*-Algorithmus

bounds_height

Gibt die Höhe des Begrenzungsrahmens einer Geometrie zurück. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	bounds_height(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `bounds_height(@geometry)` → Höhe des Begrenzungsrahmens der Geometrie des aktuellen Objekts • `bounds_height(geom_from_wkt('Polygon((1 1, 0 0, -1 1, 1 1))'))` → 1

bounds_width

Gibt die Breite des Begrenzungsrahmens einer Geometrie zurück. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	bounds_width(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `bounds_width(@geometry)` → Breite des Begrenzungsrahmens der Geometrie des aktuellen Objekts • `bounds_width(geom_from_wkt('Polygon((1 1, 0 0, -1 1, 1 1))'))` → 2

buffer

Gibt eine Geometrie zurück, die alle Punkte repräsentiert, deren Abstand zu dieser Geometrie kleiner oder gleich dem Abstand ist. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	buffer(geometry, distance, [segments=8], [cap='round'], [join='round'], [miter_limit=2]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **distance** - Pufferabstand in Layer-Einheiten • **segments** - Anzahl der Segmente, die zur Darstellung eines Viertelkreises verwendet werden, wenn ein runder Verbindungsstil verwendet wird. Eine größere Anzahl führt zu einem glatteren Puffer mit mehr Knotenpunkten. • **cap** - Endkappenstil für Puffer. Gültige Werte sind ‚round', ‚flat' oder ‚square' • **join** - Verbindungsstil für Puffer. Gültige Werte sind ‚round', ‚bevel' oder ‚miter'. • **miter_limit** - Grenzwert für den Gehrungsabstand, wenn der Verbindungsstil auf ‚'Miter" eingestellt ist
Beispiele	• `buffer(@geometry, 10.5)` → Polygon der Geometrie des aktuellen Objekts, gepuffert um 10,5 Einheiten

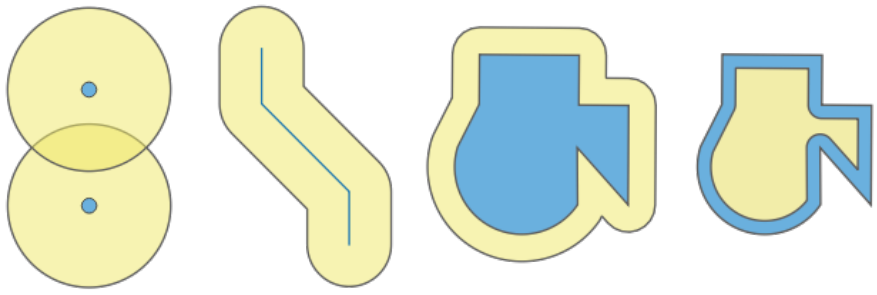


Abb. 9.7: Puffer (in gelb) von Punkten, Linie, Polygon mit positivem Puffer und Polygon mit negativem Puffer

Siehe auch: *Puffer*-Algorithmus

buffer_by_m

Erzeugt einen Puffer entlang einer Liniengeometrie, wobei der Pufferdurchmesser entsprechend den m-Werten an den Knotenpunkten der Linie variiert.

Syntax	buffer_by_m(geometry, [segments=8]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - Eingabegeometrie. Muss eine ((Multi-)Liniengeometrie mit m Werten sein. • **segments** - Anzahl der Segmente zur Annäherung von Viertelkreiskurven im Puffer.
Beispiele	• `buffer_by_m(geometry:=geom_from_wkt('LINESTRINGM(1 2 0.5, 4 2 0.2)'),segments:=8)` → Ein Puffer mit variabler Breite, beginnend mit einem Durchmesser von 0,5 und endend mit einem Durchmesser von 0,2 entlang der Liniengeometrie.

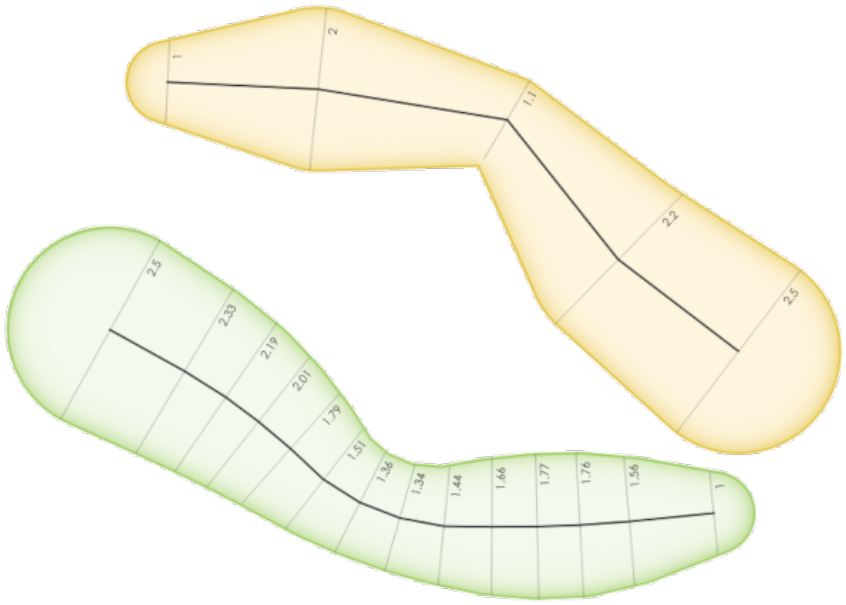


Abb. 9.8: Pufferung von Linienobjekten unter Verwendung des m-Wertes an den Knotenpunkten

Siehe auch: *Variable width buffer (by M value)*-Algorithmus

centroid

Gibt den geometrischen Mittelpunkt einer Geometrie zurück.

Syntax	centroid(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `centroid(@geometry)` → eine Punktgeometrie

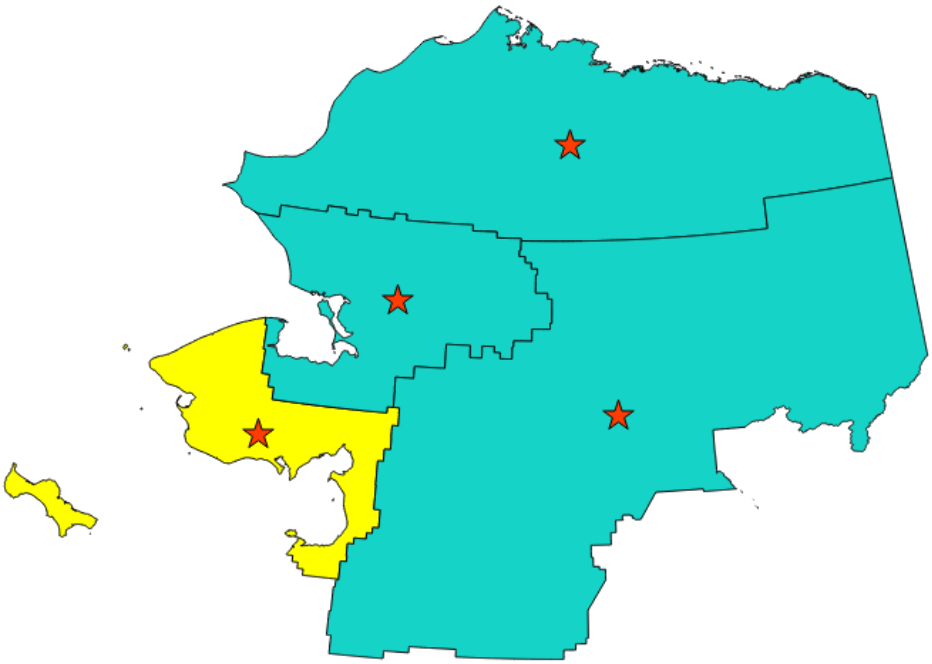


Abb. 9.9: Die roten Sterne stellen die Zentren der Objekte des Input Layers dar.

Siehe auch: *Centroids* Algorithmus

close_line

Gibt einen geschlossenen Linien-String des eingegebenen Linien-String zurück, indem der erste Punkt an das Ende der Linie angehängt wird, sofern sie nicht bereits geschlossen ist. Handelt es sich bei der Geometrie nicht um einen Linien-String oder einen Multi-Linien-String, ist das Ergebnis NULL.

Syntax	close_line(geometry)
Argumente	• **geometry** - eine Linien-String-Geometrie
Beispiele	• `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

Gibt den Punkt von geometry1 zurück, der geometry2 am nächsten ist.

Syntax	closest_point(geometry1, geometry2)
Argumente	• **geometry1** - Geometrie, in welcher der nächstgelegene Punkt gesucht werden soll • **geometry2** - Geometrie, zu welcher der nächstgelegene Punkt gefunden werden soll
Beispiele	• `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

Sammelt eine Reihe von Geometrien in einem mehrteiligen Geometrieobjekt.

Liste von Argumenten Variante

Geometrieteile werden als separate Argumente für die Funktion angegeben.

Syntax	collect_geometries(geometry1, geometry2, …)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `geom_to_wkt(collect_geometries(make_point(1,2), make_point(3,4), make_point(5,6)))` → ‚MultiPoint ((1 2),(3 4),(5 6))‘

Array-Variante

Geometrieteile werden als ein Array von Geometrieteilen angegeben.

Syntax	collect_geometries(array)
Argumente	• **array** - ein Array von Geometrieobjekten
Beispiele	• `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))‘

Siehe auch: *Collect geometries*-Algorithmus

combine

Gibt die Kombination von zwei Geometrien zurück.

Syntax	combine(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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

Gibt ein möglichst konkaves Polygon zurück, das alle Punkte der Geometrie enthält

Syntax	concave_hull(geometry, target_percent, [allow_holes=False]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **target_percent** - der prozentuale Anteil der Fläche der konvexen Hülle, dem sich die Lösung anzunähern versucht. Ein target_percent von 1 ergibt das gleiche Ergebnis wie die konvexe Hülle. Ein target_percent zwischen 0 und 0,99 ergibt ein Ergebnis, das eine kleinere Fläche als die konvexe Hülle haben sollte. • **allow_holes** – optionales Argument, das angibt, ob Löcher innerhalb der Ausgabegeometrie zulässig sind. Der Standardwert ist FALSE. Setzen Sie den Wert auf TRUE, um Löcher in der Ausgabegeometrie zuzulassen.
Beispiele	• `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))‘

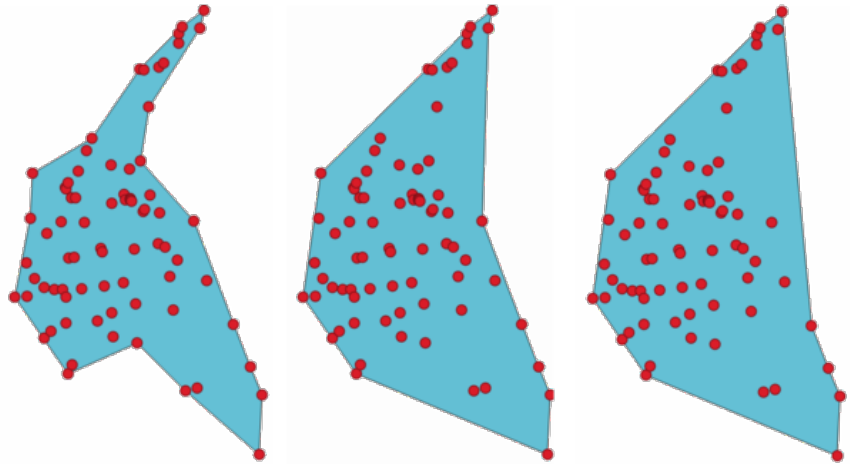


Abb. 9.10: Konkave Hüllen mit steigendem target_percent-Parameter

Siehe auch: *convex_hull*, *Concave hull* Algorithmus

contains

Prüft, ob eine Geometrie eine andere enthält. Gibt TRUE zurück, wenn keine Punkte von Geometrie2 im Äußeren von Geometrie1 liegen und mindestens ein Punkt des Inneren von Geometrie2 im Inneren von Geometrie1 liegt.

Syntax	contains(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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

Siehe auch: *overlay_contains*-Algorithmus

convex_hull

Gibt die konvexe Hülle einer Geometrie zurück. Diese stellt die minimale konvexe Geometrie dar, die alle Geometrien innerhalb der Menge umschließt.

Syntax	convex_hull(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `geom_to_wkt( convex_hull( geom_from_wkt( 'LINESTRING(3 3, 4 4, 4 10)' ) ) )` → ‚POLYGON((3 3, 4 10, 4 4, 3 3))‘

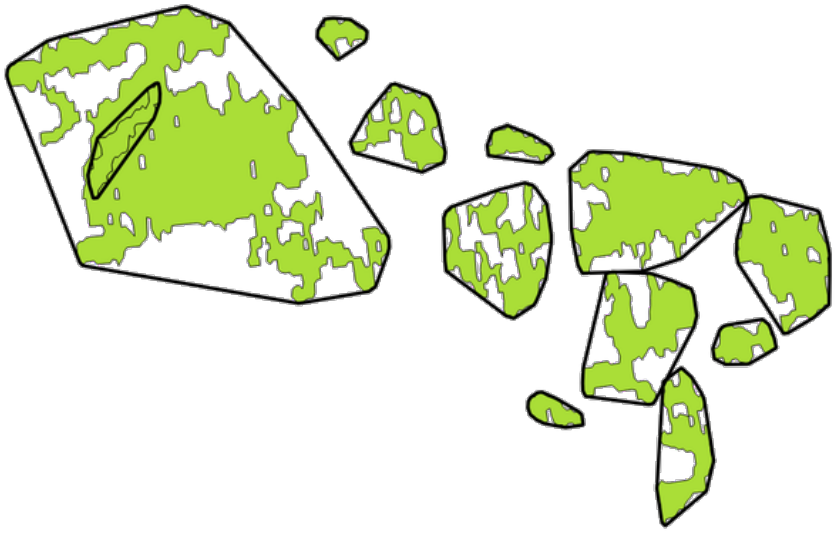


Abb. 9.11: Schwarze Linien kennzeichnen die konvexe Hülle für jedes Objekt

Siehe auch: *concave_hull*, *Convex hull*

crosses

Prüft, ob eine Geometrie eine andere kreuzt. Gibt TRUE zurück, wenn die angegebenen Geometrien einige, aber nicht alle inneren Punkte gemeinsam haben.

Syntax	crosses(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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

Siehe auch: *overlay_crosses*

densify_by_count

Nimmt eine Polygon- oder Linien-Layer-Geometrie und erzeugt eine neue, bei der die Geometrien eine größere Anzahl von Knotenpunkten haben als die ursprüngliche.

Syntax	densify_by_count(geometry, vertices)
Argumente	• **geometry** - eine Geometrie (akzeptiert (mehrere) Linienstrings oder (mehrere) Polygone). • **vertices** - Anzahl der hinzuzufügenden Knotenpunkte (pro Segment)
Beispiele	• `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)

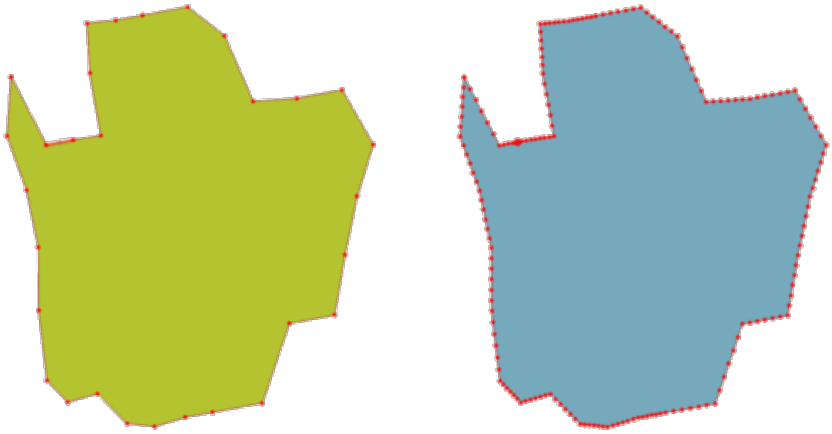


Abb. 9.12: Rote Punkte zeigen die Knotenpunkte vor und nach der Verdichtung

Siehe auch: *Densify by count*-Algorithmus

densify_by_distance

Nimmt eine Polygon- oder Linien-Layer-Geometrie und erzeugt eine neue, in der die Geometrien verdichtet werden, indem zusätzliche Knotenpunkte an Kanten hinzugefügt werden, die einen maximalen Abstand des angegebenen Intervalls haben.

Syntax	densify_by_distance(geometry, distance)
Argumente	• **geometry** - eine Geometrie (akzeptiert (mehrere) Linienstrings oder (mehrere) Polygone). • **distance** - maximaler Abstand zwischen Knotenpunkten in der Ausgangsgeometrie
Beispiele	• `geom_to_wkt(densify_by_distance(geom_from_wkt('LINESTRING(1 1, 10 1)'), 4))` → LineString (1 1, 4 1, 7 1, 10 1)

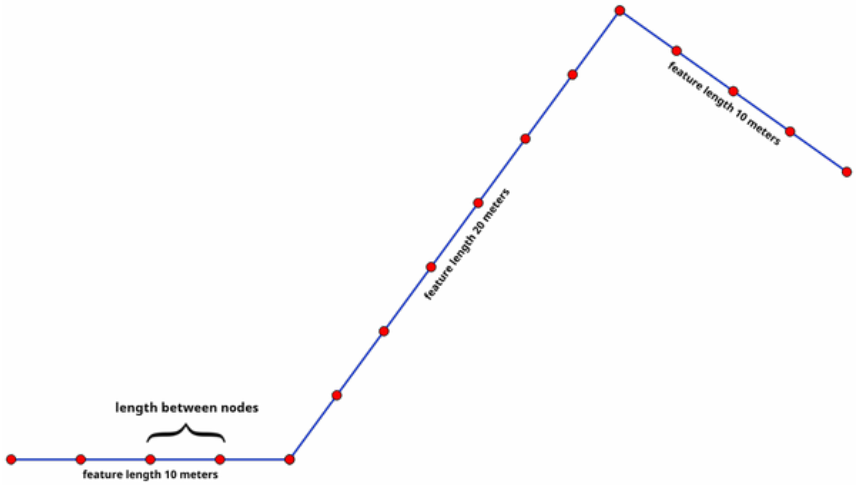


Abb. 9.13: Verdichtung der Geometrie in einem bestimmten Intervall

Siehe auch: *Densify by interval*-Algorithmus

difference

Gibt eine Geometrie zurück, die den Teil von geometry1 darstellt, der sich nicht mit geometry2 schneidet.

Syntax	difference(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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)‘

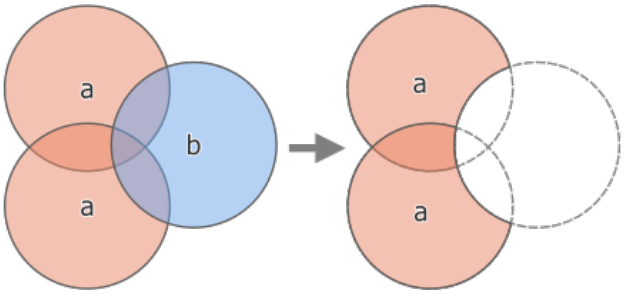


Abb. 9.14: Differenzoperation zwischen einem Eingabelayer „a“ mit zwei Objekten und einem Überlagerungslayer „b“ mit einem Objekt (links) – Ergebnis ist ein neuer Layer mit den modifizierten Objekten von „a“ (rechts)

Siehe auch: *Difference*-Algorithmus

disjoint

Prüft, ob sich die Geometrien nicht räumlich überschneiden. Gibt TRUE zurück, wenn die Geometrien keinen Raum miteinander teilen.

Syntax	disjoint(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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

Siehe auch: *overlay_disjoint*

distance

Gibt den Mindestabstand (basierend auf dem Raumbezug) zwischen zwei Geometrien in projizierten Einheiten zurück.

Syntax	distance(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `distance( geom_from_wkt( 'POINT(4 4)' ), geom_from_wkt( 'POINT(4 8)' ) )` → 4

distance_to_vertex

Gibt den Abstand entlang der Geometrie zu einem angegebenen Knotenpunkt zurück.

Syntax	distance_to_vertex(geometry, vertex)
Argumente	• **geometry** - eine Linienstring-Geometrie • **vertex** - Knotenpunktindex, beginnend bei 0; wenn der Wert negativ ist, entspricht der ausgewählte Knotenpunktindex der Gesamtzahl abzüglich des absoluten Wertes
Beispiele	• `distance_to_vertex(geometry:=geom_from_wkt('LineString(0 0, 10 0, 10 10)'),vertex:=1)` → 10.0

end_point

Gibt den letzten Knotenpunkt einer Geometrie zurück.

Syntax	end_point(geometry)
Argumente	• **geometry** - Geometrie-Objekt
Beispiele	• `geom_to_wkt(end_point(geom_from_wkt('LINESTRING(4 0, 4 2, 0 2)')))` → ‚Point (0 2)‘

Abb. 9.15: Endpunkt eines Linienobjekts

Siehe auch: *start_point*, *Extract specific vertices*-Algorithmen

exif_geotag

Erzeugt eine Punktgeometrie aus den Exif-Geotags einer Bilddatei.

Syntax	exif_geotag(path)
Argumente	• **path** - Pfad einer Bilddatei oder ein Karten-Layer. Wird ein Karten-Layer angegeben, wird die Dateiquelle des Layers verwendet.
Beispiele	• `geom_to_wkt(exif_geotag('/my/photo.jpg'))` → ‚Point (2 4)‘

extend

Verlängert den Anfang und das Ende einer Liniengeometrie um einen bestimmten Betrag. Linien werden anhand der Richtung des ersten und letzten Segments der Linie verlängert. Bei einer Multi-Linie werden alle Teile verlängert. Die Abstände werden im räumlichen Bezugssystem dieser Geometrie angegeben.

Syntax	extend(geometry, start_distance, end_distance)
Argumente	• **geometry** - eine (Multi-)Linien-Geometrie • **start_distance** - Entfernung, um den Anfang der Linie zu verlängern • **end_distance** - Entfernung, um das Ende der Linie zu verlängern.
Beispiele	• `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))‘

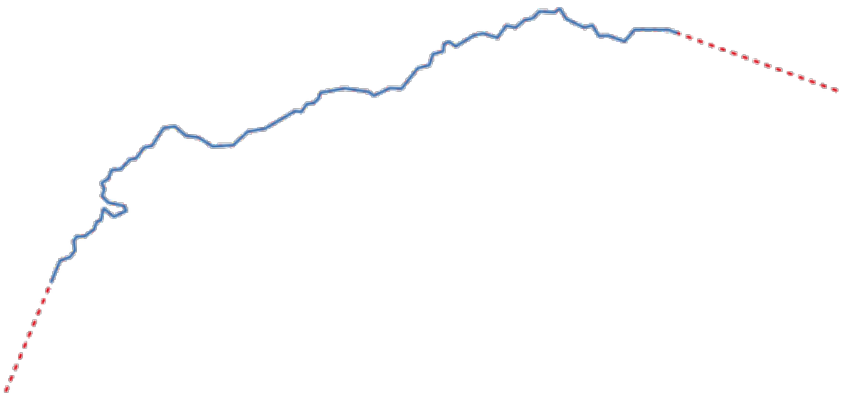


Abb. 9.16: Die roten Striche stellen die Verlängerung der ursprünglichen Geometrie dar

Siehe auch: *Extend lines*-Algorithmus

exterior_ring

Gibt eine Zeichenkette zurück, die den äußeren Ring einer Polygongeometrie darstellt. Wenn die Geometrie kein Polygon ist, ist das Ergebnis NULL.

Syntax	exterior_ring(geometry)
Argumente	• **geometry** - eine Polygon-Geometrie
Beispiele	• `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)‘

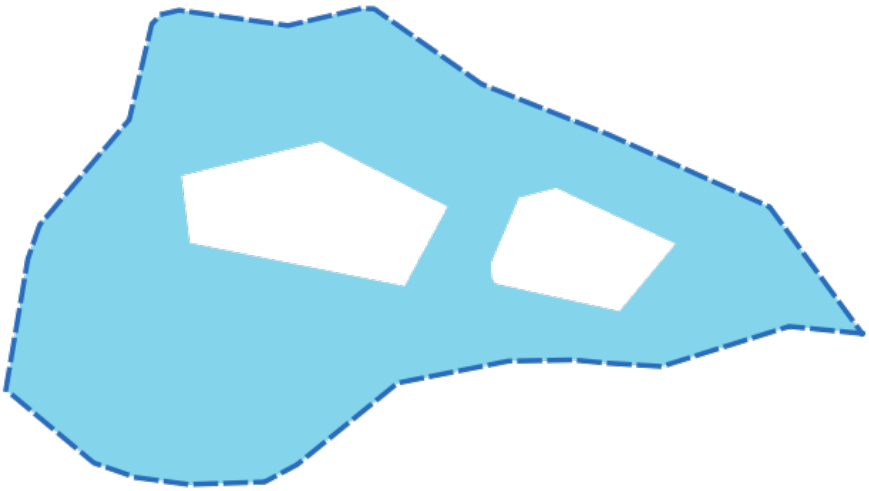


Abb. 9.17: Die gestrichelte Linie stellt den äußeren Ring des Polygons dar

Siehe auch: *Boundary* algorithm, *interior_ring_n*

extrude

Gibt eine extrudierte Version der eingegebenen (Multi-)Kurven- oder (Multi-)Liniengeometrie mit einer durch x und y angegebenen Erweiterung zurück.

Syntax	extrude(geometry, x, y)
Argumente	• **geometry** - eine Kurve oder eine Linien-Geometrie • **x** - x Erweiterung, numerischer Wert • **y** - y Erweiterung, numerischer Wert
Beispiele	• `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)))‘

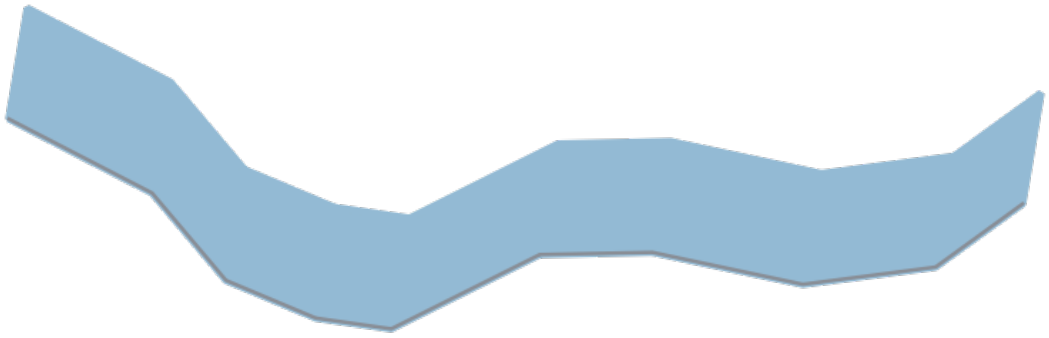


Abb. 9.18: Erzeugen eines Polygons durch Extrudieren einer Linie mit Versatz in x- und y-Richtung

flip_coordinates

Gibt eine Kopie der Geometrie zurück, bei der die x- und y-Koordinaten vertauscht sind. Nützlich für die Reparatur von Geometrien, deren Breiten- und Längengradwerte vertauscht wurden.

Syntax	flip_coordinates(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `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)‘

Siehe auch: *Swap X and Y coordinates*-Algorithmus

force_polygon_ccw

Erzwingt die Einhaltung der Konvention, dass äußere Ringe gegen den Uhrzeigersinn und innere Ringe im Uhrzeigersinn sind.

Syntax	force_polygon_ccw(geometry)
Argumente	• **geometry** - eine Geometrie. Alle Nicht-Polygon-Geometrien werden unverändert zurückgegeben.
Beispiele	• `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))‘

Siehe auch: *force_polygon_cw*, *force_rhr*

force_polygon_cw

Erzwingt die Einhaltung der Konvention, dass äußere Ringe im Uhrzeigersinn und innere Ringe gegen den Uhrzeigersinn verlaufen.

Syntax	force_polygon_cw(geometry)
Argumente	• **geometry** - eine Geometrie. Alle Nicht-Polygon-Geometrien werden unverändert zurückgegeben.
Beispiele	• `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))‘

Siehe auch: *force_polygon_ccw*, *force_rhr*

force_rhr

Erzwingt, dass eine Geometrie die Rechts-Hand-Regel einhält, bei der die von einem Polygon begrenzte Fläche rechts von der Begrenzung liegt. Insbesondere ist der äußere Ring im Uhrzeigersinn und die inneren Ringe gegen den Uhrzeigersinn ausgerichtet. Aufgrund der Inkonsistenz in der Definition der Rechts-Hand-Regel in einigen Zusammenhängen wird empfohlen, stattdessen die explizite Funktion force_polygon_cw zu verwenden.

Syntax	force_rhr(geometry)
Argumente	• **geometry** - eine Geometrie. Alle Nicht-Polygon-Geometrien werden unverändert zurückgegeben.
Beispiele	• `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))‘

Siehe auch: *Force right-hand-rule* algorithm, *force_polygon_ccw*, *force_polygon_cw*

geom_from_gml

Liefert eine Geometrie aus einer GML-Darstellung der Geometrie.

Syntax	geom_from_gml(gml)
Argumente	• **gml** - GML-Darstellung einer Geometrie als String
Beispiele	• `geom_from_gml('<gml:LineString srsName="EPSG:4326"><gml:coordinates>4,4 5,5 6,6</gml:coordinates></gml:LineString>')` → ein linienförmiges Geometrieobjekt

geom_from_wkb

Gibt eine Geometrie zurück, die aus einer Well-Known Binary (WKB) Darstellung erstellt wurde.

Syntax	geom_from_wkb(binary)
Argumente	• **binary** - Well-Known Binary (WKB) Darstellung einer Geometrie (als binärer Blob)
Beispiele	• `geom_from_wkb( geom_to_wkb( make_point(4,5) ) )` → ein punktförmiges Geometrieobjekt

geom_from_wkt

Gibt eine Geometrie zurück, die aus einer Well-Known Text (WKT)-Darstellung erstellt wurde.

Syntax	geom_from_wkt(text)
Argumente	• **text** - Well-Known Text (WKT) Darstellung einer Geometrie
Beispiele	• `geom_from_wkt( 'POINT(4 5)' )` → ein Geometrieobjekt

geom_to_wkb

Gibt die Well-Known Binary (WKB) Darstellung einer Geometrie zurück

Syntax	geom_to_wkb(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `geom_to_wkb( @geometry )` → binärer Blob, der ein Geometrieobjekt enthält

geom_to_wkt

Gibt die Well-Known Text (WKT)-Darstellung der Geometrie ohne SRID-Metadaten zurück.

Syntax	geom_to_wkt(geometry, [precision=8]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **precision** - numerische Genauigkeit
Beispiele	• `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

Gibt die Geometrie des aktuellen Objekts zurück. Kann für die Verarbeitung mit anderen Funktionen verwendet werden. **WARNUNG: Diese Funktion ist veraltet. Es wird empfohlen, stattdessen @geometry zu verwenden.**

Syntax	$geometry
Beispiele	• `geom_to_wkt( $geometry )` → ‚POINT(6 50)‘

geometry

Gibt die Geometrie eines Objekts zurück.

Syntax	geometry(feature)
Argumente	• **feature** - ein Objekt
Beispiele	• `geometry( @feature )` → die Geometrie des aktuellen Objekts. Bevorzugen Sie die Verwendung von @geometry. • `geom_to_wkt( geometry( get_feature_by_id( 'streets', 1 ) ) )` → die Geometrie in WKT des Objekts mit der id 1 im Layer „streets“, z. B. ‚POINT(6 50)‘ • `intersects( @geometry, geometry( get_feature( 'streets', 'name', 'Main St.' ) ) )` → TRUE, wenn das aktuelle Objekt das mit ‚Main St.‘ benannte Objekt im Layer ‚streets‘ räumlich schneidet

geometry_n

Gibt eine bestimmte Geometrie aus einer Geometriesammlung zurück, oder NULL, wenn die Eingabegeometrie keine Sammlung ist. Gibt auch einen Teil aus einer mehrteiligen Geometrie zurück.

Syntax	geometry_n(geometry, index)
Argumente	• **geometry** - Geometrie-Sammlung • **index** - Index der zurückzugebenden Geometrie, wobei 1 die erste Geometrie in der Sammlung ist
Beispiele	• `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

Gibt einen String-Wert zurück, der den Typ einer Geometrie beschreibt (Punkt, Linie oder Polygon)

Syntax	geometry_type(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `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

Gibt den Hausdorff-Abstand zwischen zwei Geometrien zurück. Dies ist ein Maß dafür, wie ähnlich oder unähnlich zwei Geometrien sind, wobei ein geringerer Abstand auf ähnlichere Geometrien hinweist.

Die Funktion kann mit dem optionalen Argument densify fraction ausgeführt werden. Wird es nicht angegeben, wird eine Annäherung an den Standard-Hausdorff-Abstand verwendet. Diese Annäherung ist für eine große Teilmenge nützlicher Fälle exakt oder nahe genug. Beispiele für diese Fälle sind:

- Berechnung des Abstands zwischen Linestrings, die ungefähr parallel zueinander verlaufen und ungefähr gleich lang sind. Dies ist der Fall bei zusammenpassenden, linearen Netzen.
- Prüfen der Ähnlichkeit von Geometrien.

Wenn die von dieser Methode bereitgestellte Standardannäherung unzureichend ist, geben Sie das optionale Argument densify fraction an. Durch die Angabe dieses Arguments wird eine Segmentverdichtung durchgeführt, bevor der diskrete Hausdorff-Abstand berechnet wird. Der Parameter legt den Bruchteil fest, um den jedes Segment verdichtet werden soll. Jedes Segment wird in eine Anzahl von Untersegmenten gleicher Länge aufgeteilt, deren Anteil an der Gesamtlänge dem angegebenen Anteil am nächsten kommt. Eine Verringerung des Parameters für den Verdichtungsanteil führt dazu, dass sich der zurückgegebene Abstand dem wahren Hausdorff-Abstand für die Geometrien annähert.

Syntax	hausdorff_distance(geometry1, geometry2, [densify_fraction]) [] kennzeichnet optionale Argumente
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie • **densify_fraction** - densify-fraction-Anteil
Beispiele	• `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

Liefert die Neigung gemessen vom Zenit (0) zum Fußpunkt (180) von point_a zu point_b.

Syntax	inclination(point_a, point_b)
Argumente	• **point_a** - Punktgeometrie • **point_b** - Punktgeometrie
Beispiele	• `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

Gibt einen bestimmten Innenring einer Polygongeometrie zurück, oder NULL, wenn die Geometrie kein Polygon ist.

Syntax	interior_ring_n(geometry, index)
Argumente	• **geometry** - Polygon-Geometrie • **index** - Index des zurückzugebenden Innenrings, wobei 1 der erste Innenring ist
Beispiele	• `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)‘

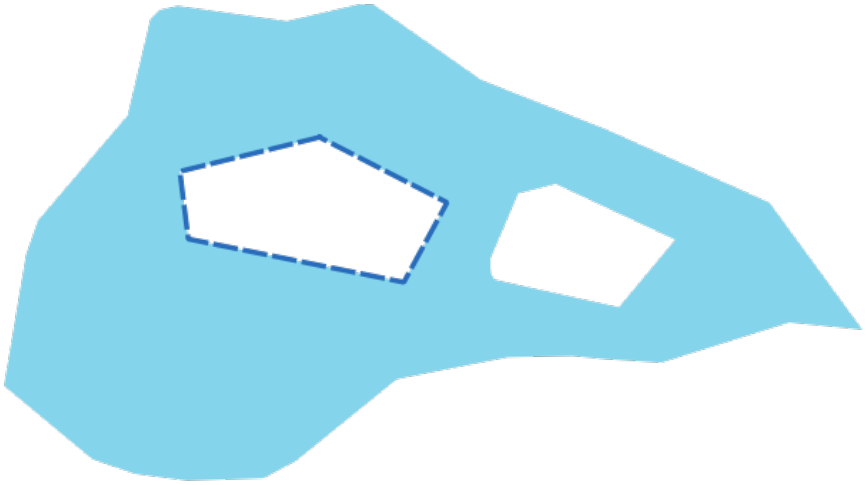


Abb. 9.19: Die gestrichelte Linie stellt einen bestimmten inneren Ring des Polygons dar.

Siehe auch: *exterior_ring*

intersection

Gibt eine Geometrie zurück, die den gemeinsamen Teil von zwei Geometrien darstellt.

Syntax	intersection(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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)'

Siehe auch: *Intersection*-Algorithmus

intersects

Prüft, ob sich eine Geometrie mit einer anderen schneidet. Gibt TRUE zurück, wenn sich die Geometrien räumlich überschneiden (einen beliebigen Bereich des Raums teilen), und FALSE, wenn sie das nicht tun.

Syntax	intersects(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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

Siehe auch: *overlay_intersects*

intersects_bbox

Prüft, ob sich der Begrenzungsrahmen einer Geometrie mit dem Begrenzungsrahmen einer anderen Geometrie überschneidet. Gibt TRUE zurück, wenn die Geometrien das definierte Begrenzungsrechteck räumlich schneiden, und FALSE, wenn sie es nicht tun.

Syntax	intersects_bbox(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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

Gibt TRUE zurück, wenn eine Linienkette geschlossen ist (Anfangs- und Endpunkt sind deckungsgleich), oder FALSE, wenn eine Linienkette nicht geschlossen ist. Wenn die Geometrie keine Linienkette ist, ist das Ergebnis NULL.

Syntax	is_closed(geometry)
Argumente	• **geometry** - eine Linien-String-Geometrie
Beispiele	• `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

Gibt TRUE zurück, wenn eine Geometrie leer ist (ohne Koordinaten), false, wenn die Geometrie nicht leer ist und NULL, wenn es keine Geometrie gibt.

Syntax	is_empty(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `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

Siehe auch: *is_empty_or_null*

is_empty_or_null

Gibt TRUE zurück, wenn eine Geometrie NULL oder leer (ohne Koordinaten) ist, ansonsten FALSE. Diese Funktion entspricht dem Ausdruck ‚@geometry IS NULL or is_empty(@geometry)'.

Syntax	is_empty_or_null(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `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

Siehe auch: *is_empty*, *NULL*

is_multipart

Gibt TRUE zurück, wenn die Geometrie vom Typ Multi ist.

Syntax	is_multipart(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `is_multipart(geom_from_wkt('MULTIPOINT ((0 0),(1 1),(2 2))'))` → TRUE • `is_multipart(geom_from_wkt('POINT (0 0)'))` → FALSE

is_valid

Gibt TRUE zurück, wenn eine Geometrie gültig ist; wenn sie gemäß den OGC-Regeln wohlgeformt in 2D ist.

Syntax	is_valid(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `is_valid(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2, 0 0)'))` → TRUE • `is_valid(geom_from_wkt('LINESTRING(0 0)'))` → FALSE

Siehe auch: *make_valid*, *Check validity*-Algorithmus

$length

Gibt die Länge eines Linienstrangs zurück. Wenn Sie die Länge des Randes eines Polygons benötigen, verwenden Sie stattdessen $perimeter. Die von dieser Funktion berechnete Länge berücksichtigt sowohl die Ellipsoid- als auch die Entfernungseinheit-Einstellungen des aktuellen Projekts. Wenn zum Beispiel ein Ellipsoid für das Projekt eingestellt wurde, dann wird die berechnete Länge ellipsoidisch sein, und wenn kein Ellipsoid eingestellt ist, dann wird die berechnete Länge planimetrisch sein.

Syntax	$length
Beispiele	• `$length` → 42.4711

length

Gibt die Anzahl der Zeichen in einer Zeichenkette oder die Länge eines geometrischen Linienstring zurück.

Zeichenketten-Variante

Gibt die Anzahl der Zeichen in einer Zeichenkette zurück.

Syntax	length(string)
Argumente	• **string** - Zeichenkette, von welcher die Länge gezählt werden soll
Beispiele	• `length('hello')` → 5

Geometrie-Variante

Berechnet die Länge eines geometrischen Linienobjekts. Die Berechnungen erfolgen immer planimetrisch im räumlichen Bezugssystem (SRS) dieser Geometrie, und die Einheiten der zurückgegebenen Länge entsprechen den Einheiten für das SRS. Dies unterscheidet sich von den Berechnungen, die von der Funktion $length durchgeführt werden, die ellipsoidische Berechnungen basierend auf den Einstellungen für Ellipsoid und Entfernungseinheiten des Projekts vornimmt.

Syntax	length(geometry)
Argumente	• **geometry** - Liniengeometrie-Objekt
Beispiele	• `length(geom_from_wkt('LINESTRING(0 0, 4 0)'))` → 4.0

Siehe auch: *straight_distance_2d*

length3D

Berechnet die 3D-Länge eines geometrischen Linienobjekts. Wenn die Geometrie kein 3D-Linienobjekt ist, wird ihre 2D-Länge zurückgegeben. Die Berechnungen sind immer planimetrisch im räumlichen Bezugssystem (SRS) dieser Geometrie, und die Einheiten der zurückgegebenen Länge entsprechen den Einheiten für das SRS. Dies unterscheidet sich von den Berechnungen, die von der Funktion $length durchgeführt werden, die ellipsoidische Berechnungen auf der Grundlage der Einstellungen für Ellipsoid und Entfernungseinheit des Projekts vornimmt.

Syntax	length3D(geometry)
Argumente	• **geometry** - Liniengeometrie-Objekt
Beispiele	• `length3D(geom_from_wkt('LINESTRINGZ(0 0 0, 3 0 4)'))` → 5.0

line_interpolate_angle

Liefert den Winkel parallel zu einer Liniengeometrie für eine vorgebenen Strecke entlang der Linie. Die Winkelangabe erfolgt in Grad im Uhrzeigersinn von Nord ausgehend.

Syntax	line_interpolate_angle(geometry, distance)
Argumente	• **geometry** - eine Linienstring-Geometrie • **distance** - Abstand entlang der Linie zur Interpolation des Winkels
Beispiele	• `line_interpolate_angle(geometry:=geom_from_wkt('LineString(0 0, 10 0)'),distance:=5)` → 90.0

line_interpolate_point

Gibt den Punkt zurück, der in einen bestimmten Abstand entlang einer Liniengeometrie interpoliert wurde.

Syntax	line_interpolate_point(geometry, distance)
Argumente	• **geometry** - eine Linienstring-Geometrie • **distance** - Abstand entlang der Linie zur Interpolation
Beispiele	• `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

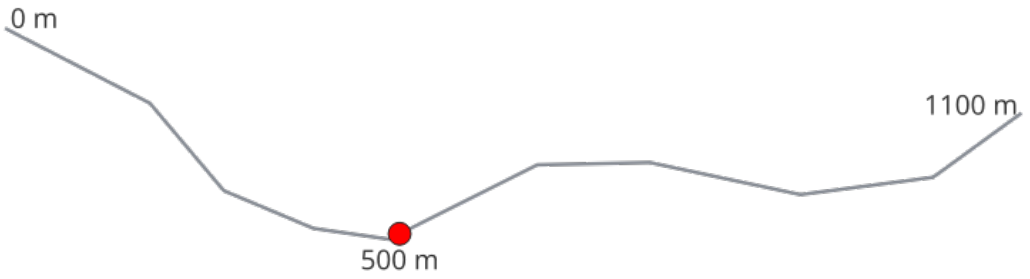


Abb. 9.20: Interpolierter Punkt in 500 m Entfernung vom Beginn der Strecke

Siehe auch: *Interpolate point on line*-Algorithmus

line_interpolate_point_by_m

Gibt den Punkt zurück, der durch einen passenden M-Wert entlang einer Liniengeometrie interpoliert wird.

Syntax	line_interpolate_point_by_m(geometry, m, [use_3d_distance=false]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Linienstring-Geometrie • **m** - ein M-Wert • **use_3d_distance** – Legt fest, ob bei der Interpolation 2D- oder 3D-Abstände zwischen den Eckpunkten verwendet werden sollen (diese Option wird nur für Linien mit z-Werten berücksichtigt).
Beispiele	• `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

Gibt die Entfernung entlang einer Linienkette zurück, die dem ersten übereinstimmenden interpolierten M-Wert entspricht.

Syntax	line_locate_m(geometry, m, [use_3d_distance=false]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Linienstring-Geometrie • **m** - ein M-Wert • **use_3d_distance** – Legt fest, ob bei der Interpolation 2D- oder 3D-Abstände zwischen den Eckpunkten verwendet werden sollen (diese Option wird nur für Linien mit z-Werten berücksichtigt).
Beispiele	• `line_locate_m(geometry:=geom_from_wkt('LineStringM(0 0 0, 10 10 10)'),m:=5)` → 7.07106

line_locate_point

Liefert den Abstand des nächsten Punktes auf einer Linie zu einem gegebenen Punkt.

Syntax	line_locate_point(geometry, point)
Argumente	• **geometry** - eine Linienstring-Geometrie • **point** - Punktgeometrie, um die nächstgelegene Position auf der Linie zu finden
Beispiele	• `line_locate_point(geometry:=geom_from_wkt('LineString(0 0, 10 0)'),point:=geom_from_wkt('Point(5 0)'))` → 5.0

line_merge

Gibt eine LineString- oder MultiLineString-Geometrie zurück, bei der alle verbundenen LineStrings aus der Eingabegeometrie zu einem einzigen LineString zusammengeführt wurden. Diese Funktion gibt NULL zurück, wenn eine Geometrie übergeben wird, die kein LineString/MultiLineString ist.

Syntax	line_merge(geometry)
Argumente	• **geometry** - eine LineString/MultiLineString-Geometrie
Beispiele	• `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

Gibt den Teil der Geometrie einer Linie (oder Kurve) zurück, der zwischen den angegebenen Start- und Endabständen liegt (gemessen vom Anfang der Linie). Z- und M-Werte werden linear aus vorhandenen Werten interpoliert.

Syntax	line_substring(geometry, start_distance, end_distance)
Argumente	• **geometry** - ein Linienstring oder eine Kurvengeometrie • **start_distance** - Abstand zum Beginn des Teilabschnitts der Geometrie • **end_distance** - Abstand zum Ende des Teilabschnitts der Geometrie
Beispiele	• `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)‘

Abb. 9.21: Teilabschnitt mit einem Startabstand von 0 Metern und einem Endabstand von 250 Metern

Siehe auch: *Line substring*-Algorithmus

m

Gibt den m-Wert (measure) einer Punktgeometrie zurück.

Syntax	m(geometry)
Argumente	• **geometry** - eine Punktgeometrie
Beispiele	• `m( geom_from_wkt( 'POINTM(2 5 4)' ) )` → 4

m_at

Liefert eine m-Koordinate der Geometrie, oder NULL, wenn die Geometrie keinen m-Wert hat.

Syntax	m_at(geometry, vertex)
Argumente	• **geometry** - Geometrie-Objekt • **vertex** - Index des Knotenpunkts der Geometrie (die Indizes beginnen bei 0; negative Werte gelten ab dem letzten Index, beginnend bei -1)
Beispiele	• `m_at(geom_from_wkt('LineStringZM(0 0 0 0, 10 10 0 5, 10 10 0 0)'), 1)` → 5

m_max

Gibt den maximalen m-Wert („measure“) einer Geometrie zurück.

Syntax	m_max(geometry)
Argumente	• **geometry** - eine Geometrie mit m-Werten
Beispiele	• `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

Gibt den minimalen m-Wert („measure“) einer Geometrie zurück.

Syntax	m_min(geometry)
Argumente	• **geometry** - eine Geometrie mit m-Werten
Beispiele	• `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

Gibt den Winkel der Längsachse (im Uhrzeigersinn, in Grad von Nord) des orientierten minimalen begrenzenden Rechtecks zurück, das die Geometrie vollständig abdeckt.

Syntax	main_angle(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `main_angle(geom_from_wkt('Polygon ((321577 129614, 321581 129618, 321585 129615, 321581 129610, 321577 129614))'))` → 38.66

make_circle

Erzeugt ein kreisförmiges Polygon.

Syntax	make_circle(center, radius, [segments=36]) [] kennzeichnet optionale Argumente
Argumente	• **center** - Mittelpunkt des Kreises • **radius** - Radius des Kreises • **segments** - optionales Argument für die Polygonsegmentierung. Standardmäßig ist dieser Wert 36.
Beispiele	• `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

Erzeugt ein elliptisches Polygon.

Syntax	make_ellipse(center, semi_major_axis, semi_minor_axis, azimuth, [segments=36]) [] kennzeichnet optionale Argumente
Argumente	• **center** - Mittelpunkt der Ellipse • **semi_major_axis** - Hauptachse der Ellipse • **semi_minor_axis** - Nebenachse der Ellipse • **azimuth** - Ausrichtung der Ellipse • **segments** - optionales Argument für die Polygonsegmentierung. Standardmäßig ist dieser Wert 36.
Beispiele	• `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

Erzeugt eine Liniengeometrie aus einer Reihe von Punktgeometrien.

Liste von Argumenten Variante

Linien-Knotenpunkte werden als separate Argumente für die Funktion angegeben.

Syntax	make_line(point1, point2, …)
Argumente	• **point** - eine Punktgeometrie (oder ein Array von Punkten)
Beispiele	• `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)‘

Array-Variante

Linien-Knotenpunkte werden als Array von Punkten angegeben.

Syntax	make_line(array)
Argumente	• **array** - Punkte-Array
Beispiele	• `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

Erzeugt eine Punktgeometrie aus einem x- und y-Wert (optional auch mit z- und/oder m-Wert).

Syntax	make_point(x, y, [z], [m]) [] kennzeichnet optionale Argumente
Argumente	• **x** - x-Koordinate des Punkts • **y** - y-Koordinate des Punkts • **z** - optionale z-Koordinate des Punkts • **m** - optionaler m-Wert des Punkts
Beispiele	• `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

Erzeugt eine Punktgeometrie aus einer x-, y-Koordinate und einem m-Wert.

Syntax	make_point_m(x, y, m)
Argumente	• **x** - x-Koordinate des Punkts • **y** - y-Koordinate des Punkts • **m** - m-Wert des Punkts
Beispiele	• `geom_to_wkt(make_point_m(2,4,6))` → ‚PointM (2 4 6)‘

make_polygon

Erzeugt eine Polygongeometrie aus einem äußeren Ring und einer optionalen Reihe von inneren Ringgeometrien.

Syntax	make_polygon(outerRing, [innerRing1], [innerRing2], …) [] kennzeichnet optionale Argumente
Argumente	• **outerRing** - geschlossene Linien-Geometrie für den äußeren Ring des Polygons • **innerRing** - optionale geschlossene Linien-Geometrie für den Innenring
Beispiele	• `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

Erzeugt ein Rechteck aus 3 Punkten.

Syntax	make_rectangle_3points(point1, point2, point3, [option=0]) [] kennzeichnet optionale Argumente
Argumente	• **point1** - Erster Punkt • **point2** - Zweiter Punkt • **point3** - Dritter Punkt • **option** - Ein optionales Argument, um das Rechteck zu konstruieren. Standardmäßig ist dieser Wert 0. Der Wert kann 0 (Abstand) oder 1 (projiziert) sein. Bei Option Abstand ist der zweite Abstand gleich dem Abstand zwischen dem 2. und 3. Punkt. Bei Option projiziert ist der zweite Abstand gleich dem Abstand der senkrechten Projektion des dritten Punktes auf das Segment oder dessen Verlängerung.
Beispiele	• `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

Erzeugt ein regelmäßiges Polygon.

Syntax	make_regular_polygon(center, radius, number_sides, [circle=0]) [] kennzeichnet optionale Argumente
Argumente	• **center** - Mittelpunkt des regelmäßigen Polygons • **radius** - zweiter Punkt. Der erste, wenn das regelmäßige Polygon eingeschrieben ist. Der Mittelpunkt der ersten Seite, wenn das reguläre Polygon umschrieben ist. • **number_sides** - Anzahl der Seiten/Kanten des regelmäßigen Polygons • **circle** - Optionales Argument zur Konstruktion des regelmäßigen Polygons. Der Wert kann 0 (eingeschrieben) oder 1 (umschrieben) sein. Standardmäßig ist dieser Wert 0.
Beispiele	• `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

Erzeugt ein Quadrat aus einer Diagonale.

Syntax	make_square(point1, point2)
Argumente	• **point1** - Erster Punkt der Diagonale • **point2** - Zweiter Punkt der Diagonale
Beispiele	• `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

Erzeugt ein dreieckiges Polygon.

Syntax	make_triangle(point1, point2, point3)
Argumente	• **point1** - erster Punkt des Dreiecks • **point2** - zweiter Punkt des Dreiecks • **point3** - dritter Punkt des Dreiecks
Beispiele	• `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

Gibt eine gültige Geometrie zurück oder eine leere Geometrie, wenn die Geometrie nicht gültig gemacht werden konnte.

Syntax	make_valid(geometry, [method=structure], [keep_collapsed=false]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **method** - Reparaturalgorithmus. Kann entweder „Struktur“ oder „Linienwerk“ sein. Die Option „Linienwerk“ kombiniert alle Ringe zu einem Satz von Knotenlinien und extrahiert dann gültige Polygone aus diesem Linienwerk. Bei der Methode „Struktur“ werden zunächst alle Ringe gültig gemacht und dann die Schalen zusammengeführt und die Löcher von den Schalen abgezogen, um ein gültiges Ergebnis zu erhalten. Setzt voraus, dass Löcher und Schalen korrekt kategorisiert sind. • **keep_collapsed** - falls auf true gesetzt, werden Komponenten, die in eine niedrigere Dimensionalität kollabiert sind, beibehalten. Zum Beispiel ein Ring, der zu einer Linie kollabiert, oder eine Linie, die zu einem Punkt kollabiert ist.
Beispiele	• `geom_to_wkt(make_valid(geom_from_wkt('POLYGON((3 2, 4 1, 5 8, 3 2, 4 2))')))` → ‚Polygon ((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)'))` → Eine leere Geometrie

Siehe auch: *is_valid*, *Fix geometries* Algorithmus

minimal_circle

Gibt den minimalen Umschließungskreis einer Geometrie zurück. Es handelt sich um den minimalen Kreis, der alle Geometrien innerhalb der Menge umschließt.

Syntax	minimal_circle(geometry, [segments=36]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **segments** - optionales Argument für die Polygonsegmentierung. Standardmäßig ist dieser Wert 36.
Beispiele	• `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))‘

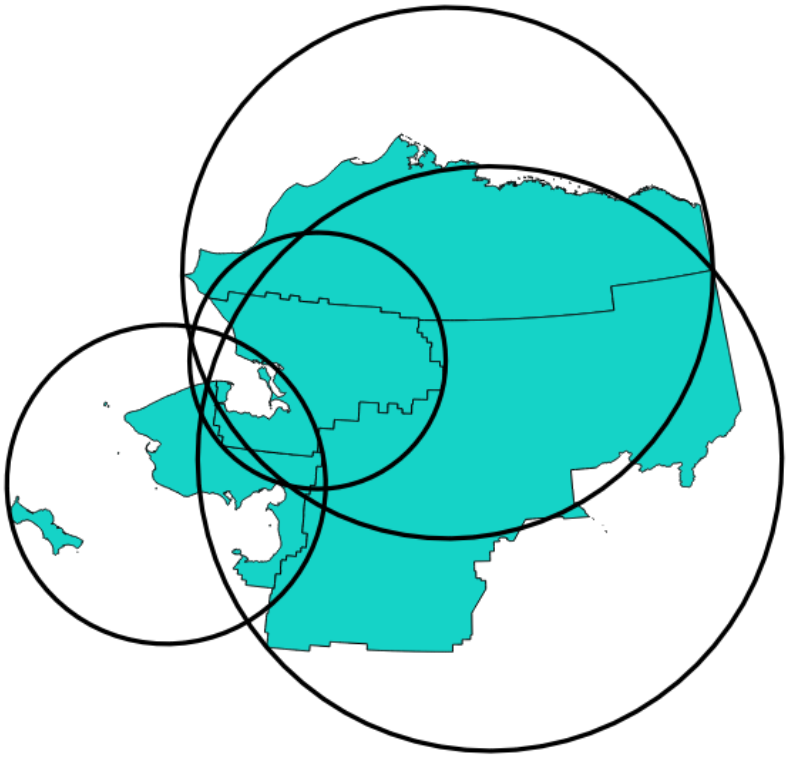


Abb. 9.22: Minimaler Umkreis für jedes Objekt

Siehe auch: *Minimum enclosing circles*-Algorithmus

nodes_to_points

Gibt eine Mehrpunktgeometrie zurück, die aus jedem Knoten in der Eingabegeometrie besteht.

Syntax	nodes_to_points(geometry, [ignore_closing_nodes=false]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - Geometrie-Objekt • **ignore_closing_nodes** - optionales Argument, das angibt, ob doppelte Knoten, die Linien oder Polygonringe schließen, einbezogen werden sollen. Der Standardwert ist false, setzen Sie ihn auf true, um diese doppelten Knoten nicht in die Ausgabe aufzunehmen.
Beispiele	• `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))‘

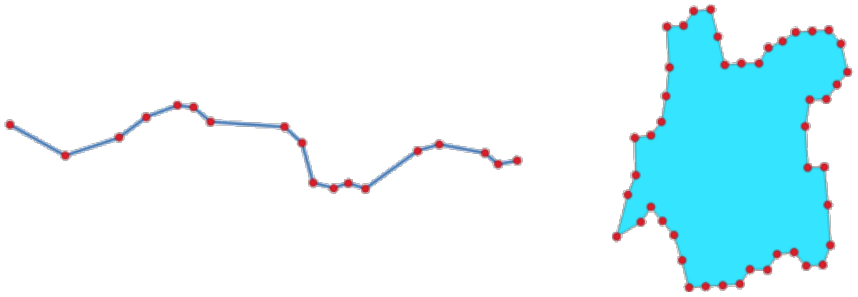


Abb. 9.23: Aus Knotenpunkten extrahierte Mehrpunkt-Objekte

Siehe auch: *Extract vertices*-Algorithmus

num_geometries

Gibt die Anzahl der Geometrien in einer Geometriesammlung oder die Anzahl der Teile in einer mehrteiligen Geometrie zurück. Die Funktion gibt NULL zurück, wenn die Eingabegeometrie keine Sammlung ist.

Syntax	num_geometries(geometry)
Argumente	• **geometry** - Geometriesammlung oder mehrteilige Geometrie
Beispiele	• `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

Gibt die Anzahl der inneren Ringe in einem Polygon oder einer Geometriesammlung zurück, oder NULL, wenn die Eingabegeometrie kein Polygon oder keine Sammlung ist.

Syntax	num_interior_rings(geometry)
Argumente	• **geometry** - Eingabegeometrie
Beispiele	• `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

Gibt die Anzahl der Knotenpunkte in einer Geometrie zurück.

Syntax	num_points(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `num_points(@geometry)` → Anzahl der Knotenpunkte in der Geometrie des aktuellen Objekts

num_rings

Gibt die Anzahl der Ringe (einschließlich der äußeren Ringe) in einem Polygon oder einer Geometriesammlung zurück, oder NULL, wenn die Eingabegeometrie kein Polygon oder keine Sammlung ist.

Syntax	num_rings(geometry)
Argumente	• **geometry** - Eingabegeometrie
Beispiele	• `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

Gibt eine Geometrie zurück, die durch Versatz einer Liniengeometrie zur Seite gebildet wird. Die Abstände sind im räumlichen Bezugssystem dieser Geometrie angegeben.

Syntax	offset_curve(geometry, distance, [segments=8], [join=1], [miter_limit=2.0]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine (Multi-)Linien-Geometrie • **distance** - Versatzabstand. Positive Werte werden links von den Linien gepuffert, negative Werte rechts. • **segments** - Anzahl der Segmente, die zur Darstellung eines Viertelkreises verwendet werden, wenn ein runder Verbindungsstil verwendet wird. Eine größere Anzahl ergibt eine glattere Linie mit mehr Knotenpunkten. • **join** - Verbindungsstil für Ecken, wobei 1 = rund, 2 = Gehrung und 3 = Fase • **miter_limit** - Begrenzung des Gehrungsverhältnisses bei sehr scharfen Ecken (wenn nur Gehrungsverbindungen verwendet werden)
Beispiele	• `offset_curve(@geometry, 10.5)` → Linie um 10,5 Einheiten nach links versetzt • `offset_curve(@geometry, -10.5)` → Linie um 10,5 Einheiten nach rechts versetzt • `offset_curve(@geometry, 10.5, segments:=16, join:=1)` → die Linie wird um 10,5 Einheiten nach links verschoben, wobei mehr Segmente verwendet werden, um eine glattere Kurve zu erhalten • `offset_curve(@geometry, 10.5, join:=3)` → eine um 10,5 Einheiten nach links versetzte Linie unter Verwendung einer abgeschrägten Verbindung

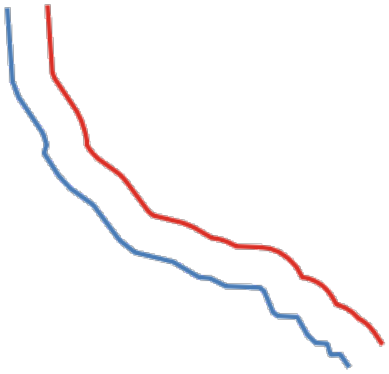


Abb. 9.24: In Blau der Quell-Layer, in Rot der Offset-Layer

Siehe auch: *Offset lines*-Algorithmus

order_parts

Sortiert die Teile eine Multigeometrie nach gegebenem Kriterium

Syntax	order_parts(geometry, orderby, [ascending=true]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie mit mehreren Typen • **orderby** - eine Zeichenkette, die die Ordnungskriterien definiert • **ascending** - Sortierung; True für aufsteigend, False für absteigend
Beispiele	• `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

Gibt eine Geometrie zurück, die den minimalen orientierten Begrenzungsrahmen einer Eingabegeometrie darstellt.

Syntax	oriented_bbox(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `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))‘

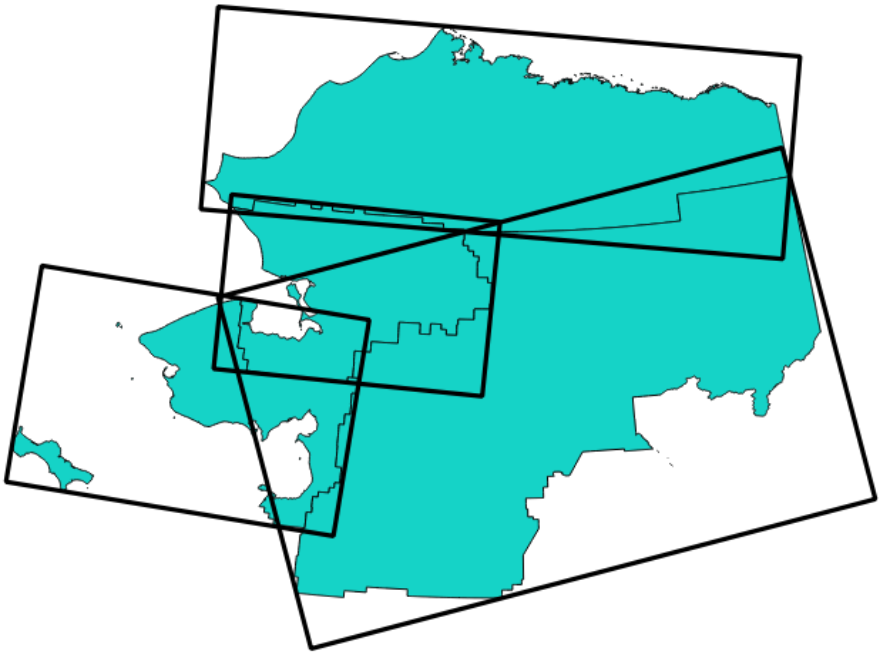


Abb. 9.25: Orientiertes minimales Begrenzungsrechteck

Siehe auch: *Orientiertes minimales Begrenzungsrechteck*-Algorithmus

overlaps

Prüft, ob eine Geometrie eine andere überlappt. Gibt TRUE zurück, wenn die Geometrien denselben Raum teilen, dieselbe Dimension haben, aber nicht vollständig ineinander enthalten sind.

Syntax	overlaps(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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

Gibt zurück, ob das aktuelle Objekt räumlich mindestens ein Objekt aus einem Ziel-Layer enthält, oder ein Array von ausdrucksbasierten Ergebnissen für die im aktuellen Objekt enthaltenen Objekte des Ziel-Layers.

Lesen Sie mehr über das zugrundeliegende GEOS „Contains“ Prädikat, wie es in der PostGIS ST_Contains Funktion beschrieben ist.

Syntax	overlay_contains(layer, [expression], [filter], [limit], [cache=false]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - der Layer, dessen Überlagerung geprüft wird • **expression** - ein optionaler Ausdruck, der für die Objekte des Ziel-Layers ausgewertet wird. Wenn er nicht gesetzt ist, gibt die Funktion nur einen booleschen Wert zurück, der angibt, ob es mindestens eine Übereinstimmung gibt. • **filter** - ein optionaler Ausdruck zum Filtern der zu prüfenden Objekte. Ist er nicht gesetzt, werden alle Objekte geprüft. • **limit** - eine optionale Ganzzahl zur Begrenzung der Anzahl der übereinstimmenden Objekte. Ist sie nicht gesetzt, werden alle übereinstimmenden Objekte zurückgegeben. • **cache** - Setzen Sie diesen Wert auf true, um einen lokalen räumlichen Index zu erstellen (in den meisten Fällen ist dies nicht ratsam, es sei denn, Sie arbeiten mit einem besonders langsamen Datenanbieter)
Beispiele	• `overlay_contains('regions')` → TRUE, wenn das aktuelle Objekt räumlich eine Region enthält • `overlay_contains('regions', filter:= population > 10000)` → TRUE, wenn das aktuelle Objekt räumlich eine Region mit einer Bevölkerung von mehr als 10.000 enthält • `overlay_contains('regions', name)` → ein Array von Namen für die Regionen, die im aktuellen Objekt enthalten sind • `array_to_string(overlay_contains('regions', name))` → eine durch Kommata getrennte Liste von Namen für die Regionen, die im aktuellen Objekt enthalten sind • `array_sort(overlay_contains(layer:='regions', expression:="name", filter:= population > 10000))` → ein geordnetes Array von Namen für die Regionen, die im aktuellen Objekt enthalten sind und eine Bevölkerung von mehr als 10.000 haben • `overlay_contains(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → eine Reihe von Geometrien (in WKT) für bis zu zwei Regionen, die im aktuellen Objekt enthalten sind

Siehe auch: *contains*, array manipulation 1, *Auswahl nach der Lage*-Algorithmus

overlay_crosses

Gibt zurück, ob das aktuelle Objekt mindestens ein Objekt des Ziel-Layers räumlich kreuzt, oder eine Reihe von ausdrucksbasierten Ergebnissen für die vom aktuellen Objekt gekreuzten Objekte im Ziel-Layer.

Lesen Sie mehr über das zugrundeliegende GEOS-Prädikat „Kreuze“, wie es in der PostGIS-Funktion ST_Crosses beschrieben ist.

Syntax	overlay_crosses(layer, [expression], [filter], [limit], [cache=false]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - der Layer, dessen Überlagerung geprüft wird • **expression** - ein optionaler Ausdruck, der für die Objekte des Ziel-Layers ausgewertet wird. Wenn er nicht gesetzt ist, gibt die Funktion nur einen booleschen Wert zurück, der angibt, ob es mindestens eine Übereinstimmung gibt. • **filter** - ein optionaler Ausdruck zum Filtern der zu prüfenden Objekte. Ist er nicht gesetzt, werden alle Objekte geprüft. • **limit** - eine optionale Ganzzahl zur Begrenzung der Anzahl der übereinstimmenden Objekte. Ist sie nicht gesetzt, werden alle übereinstimmenden Objekte zurückgegeben. • **cache** - Setzen Sie diesen Wert auf true, um einen lokalen räumlichen Index zu erstellen (in den meisten Fällen ist dies nicht ratsam, es sei denn, Sie arbeiten mit einem besonders langsamen Datenanbieter)
Beispiele	• `overlay_crosses('regions')` → TRUE, wenn das aktuelle Objekt räumlich eine Region durchquert • `overlay_crosses('regions', filter:= population > 10000)` → TRUE, wenn das aktuelle Objekt räumlich eine Region mit mehr als 10.000 Einwohnern durchquert • `overlay_crosses('regions', name)` → ein Array von Namen mit den Regionen, die von dem aktuellen Objekt durchquert werden • `array_to_string(overlay_crosses('regions', name))` → eine durch Kommata getrennte Liste von Namen für die Regionen, die von dem aktuellen Objekt durchquert werden • `array_sort(overlay_crosses(layer:='regions', expression:="name", filter:= population > 10000))` → ein geordnetes Array von Namen für die Regionen, die von dem aktuellen Objekt durchquert werden und eine Bevölkerung von mehr als 10.000 haben • `overlay_crosses(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → eine Reihe von Geometrien (in WKT) für bis zu zwei Regionen, die von dem aktuellen Objekt durchquert werden

Siehe auch: *crosses*, *array manipulation*, *Auswahl nach der Lage* algorithm

overlay_disjoint

Gibt zurück, ob das aktuelle Objekt räumlich getrennt von allen Objekten eines Ziel-Layers ist, oder eine Reihe von ausdrucksbasierten Ergebnissen für die Objekte im Ziel-Layer, die vom aktuellen Objekt getrennt sind.

Lesen Sie mehr über das zugrundeliegende GEOS-Prädikat „Disjoint“, wie es in der PostGIS-Funktion ST_Disjoint beschrieben ist.

Syntax	overlay_disjoint(layer, [expression], [filter], [limit], [cache=false]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - der Layer, dessen Überlagerung geprüft wird • **expression** - ein optionaler Ausdruck, der für die Objekte des Ziel-Layers ausgewertet wird. Wenn er nicht gesetzt ist, gibt die Funktion nur einen booleschen Wert zurück, der angibt, ob es mindestens eine Übereinstimmung gibt. • **filter** - ein optionaler Ausdruck zum Filtern der zu prüfenden Objekte. Ist er nicht gesetzt, werden alle Objekte geprüft. • **limit** - eine optionale Ganzzahl zur Begrenzung der Anzahl der übereinstimmenden Objekte. Ist sie nicht gesetzt, werden alle übereinstimmenden Objekte zurückgegeben. • **cache** - Setzen Sie diesen Wert auf true, um einen lokalen räumlichen Index zu erstellen (in den meisten Fällen ist dies nicht ratsam, es sei denn, Sie arbeiten mit einem besonders langsamen Datenanbieter)
Beispiele	• `overlay_disjoint('regions')` → TRUE, wenn das aktuelle Objekt räumlich unzusammenhängend mit allen Regionen ist • `overlay_disjoint('regions', filter:= population > 10000)` → TRUE, wenn das aktuelle Objekt räumlich von allen Regionen mit einer Bevölkerung von mehr als 10.000 getrennt ist • `overlay_disjoint('regions', name)` → ein Array von Namen für die Regionen, die räumlich vom aktuellen Objekt getrennt sind • `array_to_string(overlay_disjoint('regions', name))` → eine durch Kommata getrennte Liste von Namen für die Regionen, die räumlich nicht mit dem aktuellen Objekt verbunden sind • `array_sort(overlay_disjoint(layer:='regions', expression:="name", filter:= population > 10000))` → ein geordnetes Array von Namen für die Regionen, die räumlich nicht mit dem aktuellen Objekt verbunden sind und eine Bevölkerung von mehr als 10.000 haben • `overlay_disjoint(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → eine Reihe von Geometrien (in WKT) für bis zu zwei Regionen, die räumlich vom aktuellen Objekt getrennt sind

Siehe auch: *disjoint*, *array manipulation*, *Auswahl nach der Lage* algorithm

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.

Syntax	overlay_equals(layer, [expression], [filter], [limit], [cache=false]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - der Layer, dessen Überlagerung geprüft wird • **expression** - ein optionaler Ausdruck, der für die Objekte des Ziel-Layers ausgewertet wird. Wenn er nicht gesetzt ist, gibt die Funktion nur einen booleschen Wert zurück, der angibt, ob es mindestens eine Übereinstimmung gibt. • **filter** - ein optionaler Ausdruck zum Filtern der zu prüfenden Objekte. Ist er nicht gesetzt, werden alle Objekte geprüft. • **limit** - eine optionale Ganzzahl zur Begrenzung der Anzahl der übereinstimmenden Objekte. Ist sie nicht gesetzt, werden alle übereinstimmenden Objekte zurückgegeben. • **cache** - Setzen Sie diesen Wert auf true, um einen lokalen räumlichen Index zu erstellen (in den meisten Fällen ist dies nicht ratsam, es sei denn, Sie arbeiten mit einem besonders langsamen Datenanbieter)
Beispiele	• `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

Bemerkung

This function requires exact equality of geometries, meaning that the geometries ‚LINESTRING(0 0, 1 1)' and ‚LINESTRING(1 1, 0 0)' are different.

Siehe auch: *Array-Manipulation*, *Auswahl nach der Lage*-Algorithmus

overlay_intersects

Gibt zurück, ob das aktuelle Objekt mindestens ein Objekt aus einem Ziel-Layer räumlich schneidet, oder ein Array von ausdrucksbasierten Ergebnissen für die Objekte im Ziel-Layer, die vom aktuellen Objekt geschnitten werden.

Lesen Sie mehr über das zugrundeliegende GEOS „Intersects“ Prädikat, wie es in der PostGIS ST_Intersects Funktion beschrieben ist.

Syntax	overlay_intersects(layer, [expression], [filter], [limit], [cache=false], [min_overlap], [min_inscribed_circle_radius], [return_details], [sort_by_intersection_size]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - der Layer, dessen Überlagerung geprüft wird • **expression** - ein optionaler Ausdruck, der für die Objekte des Ziel-Layers ausgewertet wird. Wenn er nicht gesetzt ist, gibt die Funktion nur einen booleschen Wert zurück, der angibt, ob es mindestens eine Übereinstimmung gibt. • **filter** - ein optionaler Ausdruck zum Filtern der zu prüfenden Objekte. Ist er nicht gesetzt, werden alle Objekte geprüft. • **limit** - eine optionale Ganzzahl zur Begrenzung der Anzahl der übereinstimmenden Objekte. Ist sie nicht gesetzt, werden alle übereinstimmenden Objekte zurückgegeben. • **cache** - Setzen Sie diesen Wert auf true, um einen lokalen räumlichen Index zu erstellen (in den meisten Fällen ist dies nicht ratsam, es sei denn, Sie arbeiten mit einem besonders langsamen Datenanbieter) • **min_overlap** - definiert einen optionalen Ausschlussfilter: – bei Polygonen eine Mindestfläche in aktuellen Objekteinheiten zum Quadrat für den Schnittpunkt. Ergibt die Schnittmenge mehrere Polygone, wird die Schnittmenge zurückgegeben, wenn mindestens eines der Polygone eine Fläche hat, die größer oder gleich dem Wert – für Linien eine Mindestlänge in aktuellen Objekteinheiten. Wenn der Schnitt mehrere Linien ergibt, wird der Schnittpunkt zurückgegeben, wenn mindestens eine der Linien eine Länge größer oder gleich dem Wert. • **min_inscribed_circle_radius** - definiert einen optionalen Ausschlussfilter (nur für Polygone): Mindestradius in aktuellen Objekteinheiten für den maximalen Inkreis des Schnittpunkts. Wenn der Schnitt mehrere Polygone ergibt, wird der Schnitt zurückgegeben, wenn mindestens eines der Polygone einen Radius für den maximalen Inkreis hat, der größer oder gleich dem Wert. Lesen Sie mehr über das zugrundeliegende GEOS-Prädikat, wie es in der PostGIS-Funktion ST_MaximumInscribedCircle beschrieben ist. Dieses Argument erfordert GEOS >= 3.9. • **return_details** – Setzen Sie diesen Parameter auf „true“, um eine Liste von Zuordnungen zurückzugeben, die (Schlüsselfelder in Anführungszeichen) die „id“ des Objekts , den Ausdruck „result“ und den Wert „overlap“ (im Falle von Multipart das größte Element) enthalten. Der „radius“ des maximalen eingeschriebenen Kreises wird ebenfalls zurückgegeben, wenn der Ziellayer ein Polygon ist. Nur gültig bei Verwendung mit dem Ausdruck-Parameter. • **sort_by_intersection_size** - nur gültig bei Verwendung mit einem Ausdruck, setzen Sie dies auf ‚des‘, um die Ergebnisse nach dem Überschneidungswert in absteigender Reihenfolge zu sortieren oder setzen Sie dies auf ‚asc‘ für aufsteigende Reihenfolge.
Beispiele	• `overlay_intersects('regions')` → TRUE, wenn das aktuelle Objekt eine Region räumlich schneidet • `overlay_intersects('regions', filter:= population > 10000)` → TRUE, wenn das aktuelle Objekt eine Region mit mehr als 10.000 Einwohnern räumlich schneidet • `overlay_intersects('regions', name)` → ein Array von Namen für die Regionen, die von dem aktuellen Objekt durchschnitten werden • `array_to_string(overlay_intersects('regions', name))` → eine durch Kommata getrennte Liste von Namen für die Regionen, die von dem aktuellen Objekt geschnitten werden • `array_sort(overlay_intersects(layer:='regions', expression:="name", filter:= population > 10000))` → ein geordnetes Array von Namen für die Regionen, die vom aktuellen Objekt geschnitten werden und eine Bevölkerung von mehr als 10.000 haben • `overlay_intersects(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → ein Array von Geometrien (in WKT) für bis zu zwei Regionen, die von dem aktuellen Objekt geschnitten werden • `overlay_intersects(layer:='regions', min_overlap:=0.54)` → TRUE, wenn das aktuelle Objekt eine Region räumlich schneidet und die Schnittfläche (von mindestens einem der Teile im Falle von Multipolygonen) größer oder gleich 0,54 ist • `overlay_intersects(layer:='regions', min_inscribed_circle_radius:=0.54)` → TRUE, wenn das aktuelle Objekt eine Region räumlich schneidet und der Radius

Siehe auch: *intersects*, array manipulation 1, *Auswahl nach der Lage*-Algorithmus

overlay_nearest

Gibt zurück, ob das aktuelle Objekt ein oder mehrere Objekte eines Ziel-Layers in einem bestimmten Abstand hat, oder eine Reihe von ausdrucksbasierten Ergebnissen für die Objekte im Ziel-Layer in einem Abstand zum aktuellen Objekt.

Hinweis: Diese Funktion kann langsam sein und bei großen Layern viel Speicherplatz beanspruchen.

Syntax	overlay_nearest(layer, [expression], [filter], [limit=1], [max_distance], [cache=false]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - der Ziellayer • **expression** - ein optionaler Ausdruck, der für die Objekte des Ziel-Layers ausgewertet wird. Wenn er nicht gesetzt ist, gibt die Funktion nur einen booleschen Wert zurück, der angibt, ob es mindestens eine Übereinstimmung gibt. • **filter** - ein optionaler Ausdruck zum Filtern der zu prüfenden Objekte. Ist er nicht gesetzt, werden alle Objekte im Ziel-Layer verwendet. • **limit** - eine optionale Ganzzahl zur Begrenzung der Anzahl der übereinstimmenden Objekte. Ist sie nicht gesetzt, wird nur das nächstgelegene Objekt zurückgegeben. Wird sie auf -1 gesetzt, werden alle übereinstimmenden Objekte zurückgegeben. • **max_distance** - ein optionaler Abstand, um die Suche nach passenden Objekten einzuschränken. Ist dieser Wert nicht angegeben, werden alle Objekte im Ziel-Layer verwendet. • **cache** - Setzen Sie diesen Wert auf true, um einen lokalen räumlichen Index zu erstellen (in den meisten Fällen ist dies nicht ratsam, es sei denn, Sie arbeiten mit einem besonders langsamen Datenanbieter)
Beispiele	• `overlay_nearest('airports')` → TRUE, wenn der Layer „airports“ mindestens ein Objekt enthält • `overlay_nearest('airports', max_distance:= 5000)` → TRUE, wenn sich ein Flughafen in einer Entfernung von 5.000 Karteneinheiten vom aktuellen Objekt befindet • `overlay_nearest('airports', name)` → den Namen des Flughafens, der dem aktuellen Objekt am nächsten liegt, als Array • `array_to_string(overlay_nearest('airports', name))` → der Name des Flughafens, der dem aktuellen Objekt am nächsten liegt, als String • `overlay_nearest(layer:='airports', expression:= name, max_distance:= 5000)` → den Namen des nächstgelegenen Flughafens in einer Entfernung von 5.000 Karteneinheiten vom aktuellen Objekt, als Array • `overlay_nearest(layer:='airports', expression:="name", filter:= "Use"='Civilian', limit:=3)` → ein Array von Namen mit bis zu drei nächstgelegenen, zivilen Flughäfen, geordnet nach Entfernung • `overlay_nearest(layer:='airports', expression:="name", limit:= -1, max_distance:= 5000)` → ein Array mit den Namen aller Flughäfen im Umkreis von 5.000 Karteneinheiten um das aktuelle Objekt, geordnet nach Entfernung

Siehe auch: *Array Manipulation*, *Join attributes by nearest* Algorithmus

overlay_touches

Gibt zurück, ob das aktuelle Objekt mindestens ein Objekt eines Ziel-Layers räumlich berührt, oder eine Reihe von ausdrucksbasierten Ergebnissen für die vom aktuellen Objekt berührten Objekte im Ziel-Layer.

Lesen Sie mehr über das zugrundeliegende GEOS-Prädikat „Touches“, wie in der PostGIS-Funktion „ST_Touches <https://postgis.net/docs/ST_Touches.html>“ beschrieben.

Syntax	overlay_touches(layer, [expression], [filter], [limit], [cache=false]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - der Layer, dessen Überlagerung geprüft wird • **expression** - ein optionaler Ausdruck, der für die Objekte des Ziel-Layers ausgewertet wird. Wenn er nicht gesetzt ist, gibt die Funktion nur einen booleschen Wert zurück, der angibt, ob es mindestens eine Übereinstimmung gibt. • **filter** - ein optionaler Ausdruck zum Filtern der zu prüfenden Objekte. Ist er nicht gesetzt, werden alle Objekte geprüft. • **limit** - eine optionale Ganzzahl zur Begrenzung der Anzahl der übereinstimmenden Objekte. Ist sie nicht gesetzt, werden alle übereinstimmenden Objekte zurückgegeben. • **cache** - Setzen Sie diesen Wert auf true, um einen lokalen räumlichen Index zu erstellen (in den meisten Fällen ist dies nicht ratsam, es sei denn, Sie arbeiten mit einem besonders langsamen Datenanbieter)
Beispiele	• `overlay_touches('regions')` → TRUE, wenn das aktuelle Objekt eine Region räumlich berührt • `overlay_touches('regions', filter:= population > 10000)` → TRUE, wenn das aktuelle Objekt räumlich eine Region mit mehr als 10.000 Einwohnern berührt • `overlay_touches('regions', name)` → ein Array von Namen für die Regionen, die von dem aktuellen Objekt berührt werden • `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))` → ein geordnetes Array von Namen für die Regionen, die von dem aktuellen Objekt berührt werden und eine Bevölkerung von mehr als 10.000 haben • `overlay_touches(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → ein Array von Geometrien (in WKT) für bis zu zwei Regionen, die von dem aktuellen Objekt berührt werden

Siehe auch: *touches*, array manipulation 1, *Auswahl nach der Lage*-Algorithmus

overlay_within

Gibt zurück, ob das aktuelle Objekt räumlich innerhalb mindestens eines Objekts eines Ziel-Layers liegt, oder eine Reihe von ausdrucksbasierten Ergebnissen für die Objekte im Ziel-Layer, die das aktuelle Objekt enthalten.

Lesen Sie mehr über das zugrundeliegende GEOS-Prädikat „Within“, wie es in der PostGIS-Funktion ST_Within beschrieben ist.

Syntax	overlay_within(layer, [expression], [filter], [limit], [cache=false]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - der Layer, dessen Überlagerung geprüft wird • **expression** - ein optionaler Ausdruck, der für die Objekte des Ziel-Layers ausgewertet wird. Wenn er nicht gesetzt ist, gibt die Funktion nur einen booleschen Wert zurück, der angibt, ob es mindestens eine Übereinstimmung gibt. • **filter** - ein optionaler Ausdruck zum Filtern der zu prüfenden Objekte. Ist er nicht gesetzt, werden alle Objekte geprüft. • **limit** - eine optionale Ganzzahl zur Begrenzung der Anzahl der übereinstimmenden Objekte. Ist sie nicht gesetzt, werden alle übereinstimmenden Objekte zurückgegeben. • **cache** - Setzen Sie diesen Wert auf true, um einen lokalen räumlichen Index zu erstellen (in den meisten Fällen ist dies nicht ratsam, es sei denn, Sie arbeiten mit einem besonders langsamen Datenanbieter)
Beispiele	• `overlay_within('regions')` → TRUE, wenn das aktuelle Objekt räumlich innerhalb einer Region liegt • `overlay_within('regions', filter:= population > 10000)` → TRUE, wenn das aktuelle Objekt räumlich innerhalb einer Region mit einer Bevölkerung von mehr als 10.000 liegt • `overlay_within('regions', name)` → ein Array von Namen für die Regionen, die das aktuelle Objekt enthalten • `array_to_string(overlay_within('regions', name))` → eine durch Kommata getrennte Liste von Namen für die Regionen, die das aktuelle Objekt enthalten • `array_sort(overlay_within(layer:='regions', expression:="name", filter:= population > 10000))` → ein geordnetes Array von Namen für die Regionen, die das aktuelle Objekt enthalten und eine Bevölkerung von mehr als 10.000 haben • `overlay_within(layer:='regions', expression:= geom_to_wkt(@geometry), limit:=2)` → ein Array von Geometrien (in WKT), für bis zu zwei Regionen, die das aktuelle Objekt enthalten

Siehe auch: *within*, *array manipulation*, *Auswahl nach der Lage*-Algorithmus

$perimeter

Gibt die Länge des Umfangs des aktuellen Objekts zurück. Die von dieser Funktion berechnete Umfangslänge berücksichtigt sowohl die Ellipsoid- als auch die Entfernungseinheit-Einstellungen des aktuellen Projekts. Wenn zum Beispiel ein Ellipsoid für das Projekt festgelegt wurde, ist die berechnete Umfangslänge ellipsoidisch, und wenn kein Ellipsoid festgelegt wurde, ist die berechnete Umfangslänge planimetrisch.

Syntax	$perimeter
Beispiele	• `$perimeter` → 42

perimeter

Gibt den Umfang eines Geometrie-Polygon-Objekts zurück. Die Berechnungen sind immer planimetrisch im räumlichen Bezugssystem (SRS) dieser Geometrie, und die Einheiten des zurückgegebenen Umfangs entsprechen den Einheiten für das SRS. Dies unterscheidet sich von den Berechnungen, die von der Funktion $perimeter durchgeführt werden, die ellipsoidische Berechnungen auf der Grundlage der Einstellungen für Ellipsoid und Entfernungseinheiten des Projekts vornimmt.

Syntax	perimeter(geometry)
Argumente	• **geometry** - Polygon-Geometrie-Objekt
Beispiele	• `perimeter(geom_from_wkt('POLYGON((0 0, 4 0, 4 2, 0 2, 0 0))'))` → 12.0

point_n

Gibt einen bestimmten Knoten aus einer Geometrie zurück.

Syntax	point_n(geometry, index)
Argumente	• **geometry** - Geometrie-Objekt • **index** - Index des zurückzugebenden Knotens, wobei 1 der erste Knoten ist; wenn der Wert negativ ist, entspricht der ausgewählte Scheitelpunktindex der Gesamtzahl abzüglich des absoluten Werts.
Beispiele	• `geom_to_wkt(point_n(geom_from_wkt('POLYGON((0 0, 4 0, 4 2, 0 2, 0 0))'),2))` → ‚Point (4 0)‘

Siehe auch: *Extract specific vertices*-Algorithmus

point_on_surface

Gibt einen Punkt zurück, der garantiert auf der Oberfläche einer Geometrie liegt.

Syntax	point_on_surface(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `point_on_surface(@geometry)` → eine Punktgeometrie

Siehe auch: *Point on Surface*-Algorithmus

pole_of_inaccessibility

Berechnet den ungefähren Pol der Unzugänglichkeit für eine Fläche, d. h. den am weitesten von der Begrenzung der Fläche entfernten inneren Punkt. Diese Funktion verwendet den „Polylabel“-Algorithmus (Vladimir Agafonkin, 2016), einen iterativen Ansatz, der garantiert, dass der wahre Pol der Unzugänglichkeit innerhalb einer bestimmten Toleranz gefunden wird. Präzisere Toleranzen erfordern mehr Iterationen und benötigen mehr Zeit für die Berechnung.

Syntax	pole_of_inaccessibility(geometry, tolerance)
Argumente	• **geometry** - eine Geometrie • **tolerance** - maximaler Abstand zwischen dem zurückgegebenen Punkt und dem tatsächlichen Standort des Pols
Beispiele	• `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)‘

Abb. 9.26: Pole of inaccessibility

Siehe auch: *Pole of inaccessibility*-Algorithmus

project

Gibt einen Punkt zurück, der von einem Startpunkt aus projiziert wird, wobei eine Entfernung, eine Richtung (Azimut) und eine Neigung im Bogenmaß angegeben werden.

Syntax	project(point, distance, azimuth, [elevation]) [] kennzeichnet optionale Argumente
Argumente	• **point** - Startpunkt • **distance** - Entfernung, die projiziert werden soll • **azimuth** - Azimut in Grad im Uhrzeigersinn, wobei 0 dem Norden entspricht • **elevation** - Neigungswinkel im Bogenmaß
Beispiele	• `geom_to_wkt(project(make_point(1, 2), 3, radians(270)))` → ‚Point(-2, 2)‘

Siehe auch: *Project points (Cartesian)*-Algorithmus

relate

Test des Dimensional Extended 9 Intersection Model (DE-9IM) zur Darstellung der Beziehung zwischen zwei Geometrien.

Beziehungs-Variante

Gibt die Darstellung des Dimensional Extended 9 Intersection Model (DE-9IM) der Beziehung zwischen zwei Geometrien zurück.

Syntax	relate(geometry, geometry)
Argumente	• **geometry** - eine Geometrie • **geometry** - eine Geometrie
Beispiele	• `relate( geom_from_wkt( 'LINESTRING(40 40,120 120)' ), geom_from_wkt( 'LINESTRING(40 40,60 120)' ) )` → ‚FF1F00102'

Mustervergleich Variante

Prüft, ob die DE-9IM-Beziehung zwischen zwei Geometrien einem bestimmten Muster entspricht.

Syntax	relate(geometry, geometry, pattern)
Argumente	• **geometry** - eine Geometrie • **geometry** - eine Geometrie • **pattern** - DE-9IM-Muster für die Übereinstimmung
Beispiele	• `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.

Zeichenketten-Variante

Reverses the order of characters in a string.

Syntax	reverse(string)
Argumente	• **string** - string to reverse
Beispiele	• `reverse('hello')` → ‚olleh'

Geometrie-Variante

Kehrt die Richtung einer Linienkette um, indem es die Reihenfolge ihrer Knotenpunkte umkehrt.

Syntax	reverse(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `geom_to_wkt(reverse(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)')))` → ‚LINESTRING(2 2, 1 1, 0 0)‘

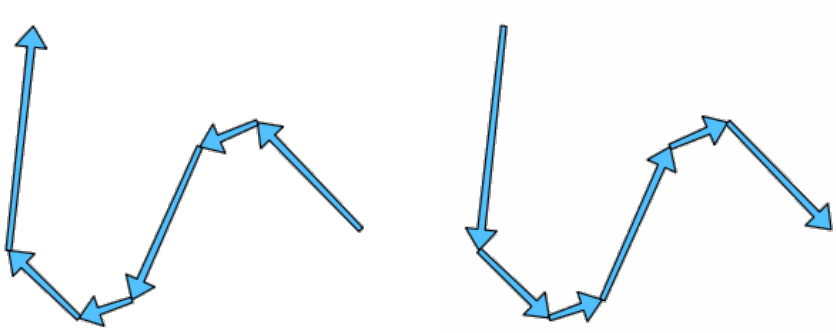


Abb. 9.27: Umkehrung der Linienrichtung

Siehe auch: *Reverse line direction*-Algorithmus

rotate

Gibt eine gedrehte Version einer Geometrie zurück. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	rotate(geometry, rotation, [center=NULL], [per_part=false]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **rotation** - Drehung im Uhrzeigersinn in Grad • **center** - Rotationsmittelpunkt. Wenn nicht angegeben, wird der Mittelpunkt der Bounding Box der Geometrie verwendet. • **per_part** - Rotation pro Teil anwenden. Wenn true, dann wird die Rotation um den Mittelpunkt der Bounding Box jedes Teils angewendet, wenn die Eingabegeometrie mehrteilig ist und kein expliziter Rotationsmittelpunkt angegeben ist.
Beispiele	• `rotate(@geometry, 45, make_point(4, 5))` → Geometrie um 45 Grad im Uhrzeigersinn um den Punkt (4, 5) gedreht • `rotate(@geometry, 45)` → Geometrie um 45 Grad im Uhrzeigersinn um den Mittelpunkt ihres Begrenzungsrahmens gedreht

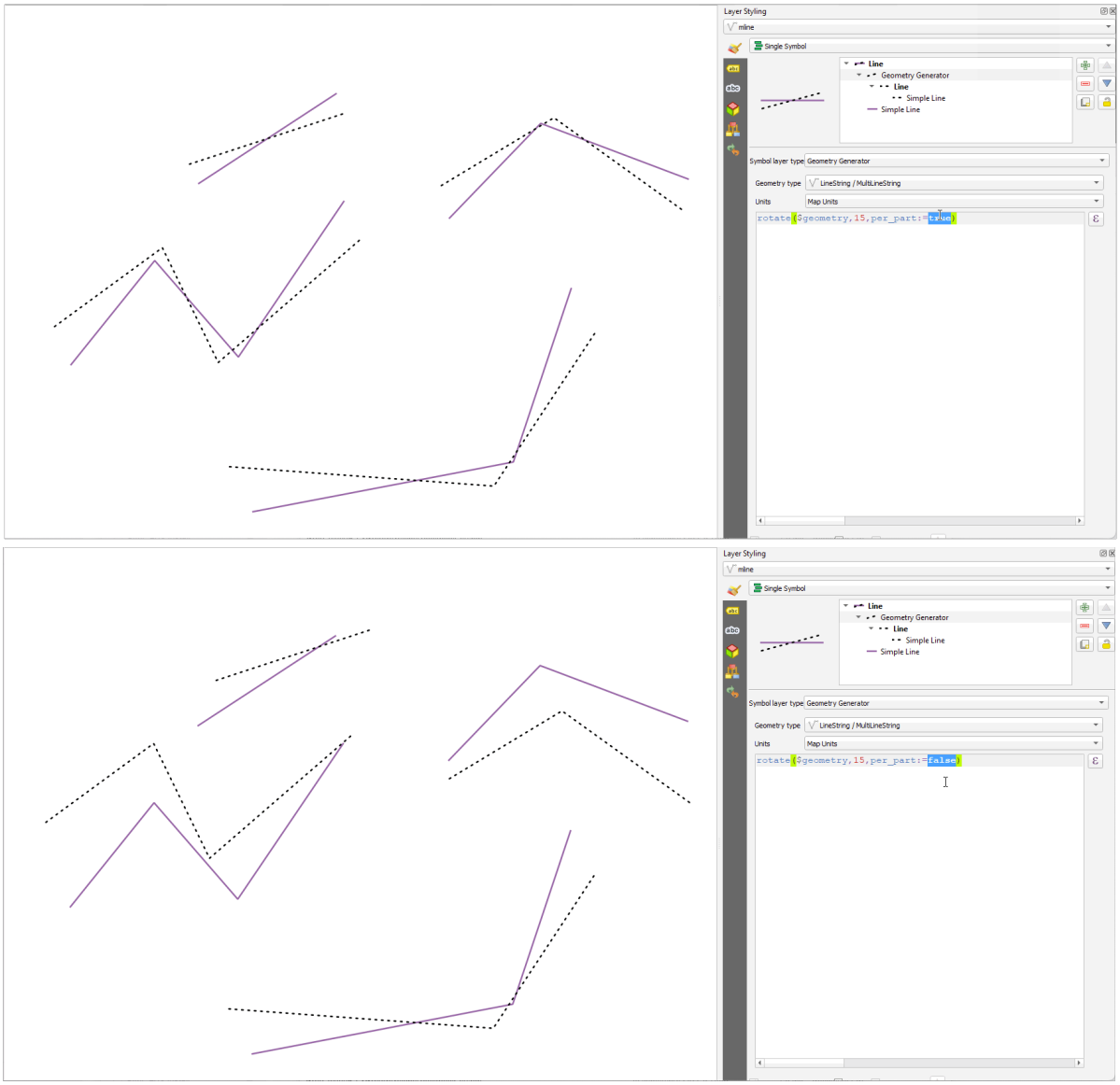


Abb. 9.28: Rotating features

roundness

Berechnet, wie nah eine Polygonform an einem Kreis liegt. Die Funktion gibt 1 zurück, wenn die Polygonform ein perfekter Kreis ist, und 0, wenn sie vollständig flach ist.

Syntax	roundness(geometry)
Argumente	• **geometry** - ein Polygon
Beispiele	• `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

Siehe auch: *Roundness*-Algorithmus

scale

Gibt eine skalierte Version einer Geometrie zurück. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	scale(geometry, x_scale, y_scale, [center]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **x_scale** - Skalierungsfaktor der x-Achse • **y_scale** - Skalierungsfaktor der y-Achse • **center** - Skalierungsmittelpunkt. Wenn nicht angegeben, wird der Mittelpunkt der Bounding Box der Geometrie verwendet.
Beispiele	• `scale(@geometry, 2, 0.5, make_point(4, 5))` → Geometrie, die horizontal zweimal skaliert und vertikal halbiert wird, um den Punkt (4, 5) • `scale(@geometry, 2, 0.5)` → Geometrie horizontal zweimal und vertikal halbiert, um den Mittelpunkt des Begrenzungsrahmens

segments_to_lines

Gibt eine Multi-Line-Geometrie zurück, die aus einer Linie für jedes Segment in der Eingabegeometrie besteht.

Syntax	segments_to_lines(geometry)
Argumente	• **geometry** - Geometrie-Objekt
Beispiele	• `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))‘

Siehe auch: *Explode lines*-Algorithmus

shared_paths

Liefert eine Sammlung mit Pfaden, die von den beiden Eingabegeometrien gemeinsam genutzt werden. Diejenigen, die in die gleiche Richtung gehen, befinden sich im ersten Element der Sammlung, diejenigen, die in die entgegengesetzte Richtung gehen, im zweiten Element. Die Pfade selbst sind in der Richtung der ersten Geometrie angegeben.

Syntax	shared_paths(geometry1, geometry2)
Argumente	• **geometry1** - eine LineString/MultiLineString-Geometrie • **geometry2** - eine LineString/MultiLineString-Geometrie
Beispiele	• `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

Gibt die kürzeste Linie zurück, die Geometrie1 mit Geometrie2 verbindet. Die resultierende Linie beginnt bei Geometrie1 und endet bei Geometrie2.

Syntax	shortest_line(geometry1, geometry2)
Argumente	• **geometry1** - Geometrie, von der die kürzeste Linie gefunden werden soll • **geometry2** - Geometrie, zu der die kürzeste Linie gefunden werden soll
Beispiele	• `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

Vereinfacht eine Geometrie durch Entfernen von Knoten unter Verwendung eines abstandsbasierten Schwellenwerts (d. h. des Douglas-Peucker-Algorithmus). Der Algorithmus bewahrt große Abweichungen in Geometrien und reduziert die Anzahl der Scheitelpunkte in nahezu geraden Segmenten.

Syntax	simplify(geometry, tolerance)
Argumente	• **geometry** - eine Geometrie • **tolerance** - maximale Abweichung von geraden Segmenten für zu entfernende Punkte
Beispiele	• `geom_to_wkt(simplify(geometry:=geom_from_wkt('LineString(0 0, 5 0.1, 10 0)'),tolerance:=5))` → ‚LineString(0 0, 10 0)‘

Abb. 9.29: Von links nach rechts: Quell-Layer und zunehmende Vereinfachungs-Toleranzen

Siehe auch: *Simplify*-Algorithmus

simplify_vw

Vereinfacht eine Geometrie durch Entfernen von Knoten unter Verwendung eines flächenbasierten Schwellenwerts (d. h. des Visvalingam-Whyatt-Algorithmus). Der Algorithmus entfernt Scheitelpunkte, die kleine Bereiche in Geometrien erzeugen, z. B. schmale Spitzen oder fast gerade Segmente.

Syntax	simplify_vw(geometry, tolerance)
Argumente	• **geometry** - eine Geometrie • **tolerance** - ein Maß für die maximale Fläche, die durch einen Knoten entsteht, wenn dieser entfernt werden soll
Beispiele	• `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)‘

Siehe auch: *Simplify*-Algorithmus

single_sided_buffer

Gibt eine Geometrie zurück, die durch Pufferung nur einer Seite einer Liniengeometrie gebildet wird. Die Abstände sind im räumlichen Bezugssystem dieser Geometrie angegeben.

Syntax	single_sided_buffer(geometry, distance, [segments=8], [join=1], [miter_limit=2.0]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine (Multi-)Linien-Geometrie • **distance** - Pufferabstand. Positive Werte werden links von den Linien gepuffert, negative Werte rechts. • **segments** - Anzahl der Segmente, die zur Darstellung eines Viertelkreises verwendet werden, wenn ein runder Verbindungsstil verwendet wird. Eine größere Anzahl führt zu einem glatteren Puffer mit mehr Knotenpunkten. • **join** - Verbindungsstil für Ecken, wobei 1 = rund, 2 = Gehrung und 3 = Fase • **miter_limit** - Begrenzung des Gehrungsverhältnisses bei sehr scharfen Ecken (wenn nur Gehrungsverbindungen verwendet werden)
Beispiele	• `single_sided_buffer(@geometry, 10.5)` → Linie um 10,5 Einheiten nach links gepuffert • `single_sided_buffer(@geometry, -10.5)` → Linie um 10,5 Einheiten nach rechts gepuffert • `single_sided_buffer(@geometry, 10.5, segments:=16, join:=1)` → eine um 10,5 Einheiten nach links gepufferte Linie, wobei mehr Segmente verwendet werden, um einen gleichmäßigeren Puffer zu erhalten • `single_sided_buffer(@geometry, 10.5, join:=3)` → eine um 10,5 Einheiten nach links gepufferte Linie, die eine abgeschrägte Verbindung verwendet

Abb. 9.30: Linke und rechte Pufferung der gleichen Linie

Siehe auch: *Single sided buffer*-Algorithmus

sinuosity

Liefert die Sinuskurve einer Kurve, d. h. das Verhältnis zwischen der Länge der Kurve und dem geraden (2D-)Abstand zwischen ihren Endpunkten.

Syntax	sinuosity(geometry)
Argumente	• **geometry** - Eingabekurve (circularstring, linestring)
Beispiele	• `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

Glättet eine Geometrie durch Hinzufügen zusätzlicher Knoten, die die Ecken der Geometrie abrunden. Wenn Eingabegeometrien Z- oder M-Werte enthalten, werden diese ebenfalls geglättet und die Ausgabegeometrie behält die gleiche Dimensionalität wie die Eingabegeometrie.

Syntax	smooth(geometry, [iterations=1], [offset=0.25], [min_length=-1], [max_angle=180]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **iterations** - Anzahl der anzuwendenden Glättungswiederholungen. Größere Zahlen führen zu glatteren, aber komplexeren Geometrien. • **offset** - Wert zwischen 0 und 0,5, der bestimmt, wie eng die geglättete Geometrie der ursprünglichen Geometrie folgt. Kleinere Werte führen zu einer engeren Glättung, größere Werte zu einer lockereren Glättung. • **min_length** - Mindestlänge der Segmente, auf die die Glättung angewendet werden soll. Dieser Parameter kann verwendet werden, um zu vermeiden, dass zu viele zusätzliche Knoten in kürzeren Segmenten der Geometrie platziert werden. • **max_angle** - maximaler Winkel am Knoten für die Anwendung der Glättung (0-180). Durch Herabsetzen des maximalen Winkels können absichtlich scharfe Ecken in der Geometrie erhalten bleiben. Bei einem Wert von 80 Grad bleiben zum Beispiel rechte Winkel in der Geometrie erhalten.
Beispiele	• `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)‘

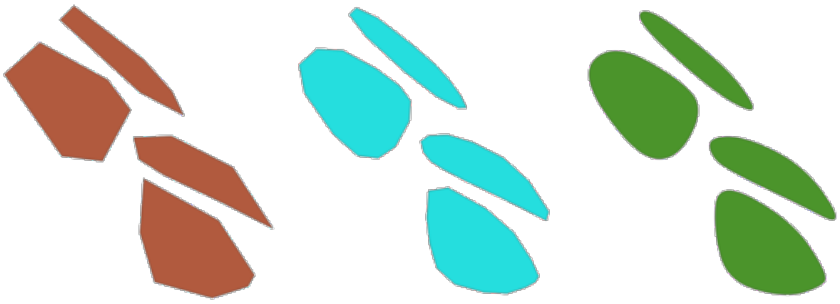


Abb. 9.31: Mit zunehmender Anzahl von Iterationen werden die Geometrien glatter

Siehe auch: *Smooth*-Algorithmus

square_wave

Konstruiert quadratische/rechteckige Wellen entlang des Randes einer Geometrie.

Syntax	square_wave(geometry, wavelength, amplitude, [strict=False]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **wavelength** - Wellenlänge der Rechteck-Wellenform • **amplitude** - Amplitude der Rechteck-Wellenform • **strict** - Standardmäßig wird das Argument Wellenlänge als „maximale Wellenlänge“ behandelt, wobei die tatsächliche Wellenlänge dynamisch angepasst wird, so dass eine genaue Anzahl von Rechteckwellen entlang der Grenzen der Geometrie erzeugt wird. Wenn das Strict-Argument auf true gesetzt wird, wird die Wellenlänge exakt verwendet und es kann ein unvollständiges Muster für die endgültige Wellenform verwendet werden.
Beispiele	• `square_wave(geom_from_wkt('LineString(0 0, 10 0)'), 3, 1)` → Rechteckwellen mit der Wellenlänge 3 und der Amplitude 1 entlang des Linienstrangs

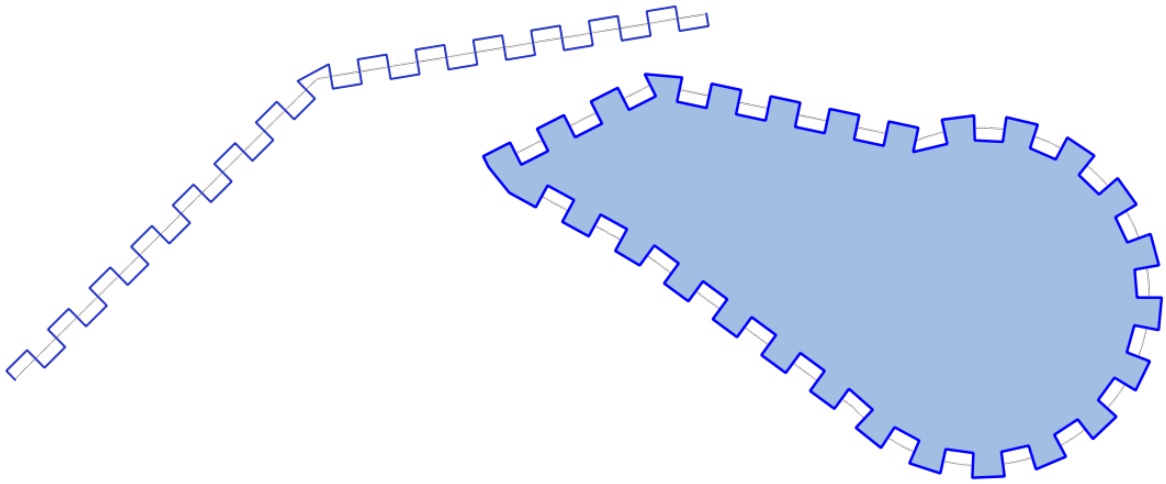


Abb. 9.32: Symbolisierung von Objekten mit Rechteckwellen

square_wave_randomized

Konstruiert zufällige quadratische/rechteckige Wellen entlang des Randes einer Geometrie.

Syntax	square_wave_randomized(geometry, min_wavelength, max_wavelength, min_amplitude, max_amplitude, [seed=0]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **min_wavelength** - Mindestwellenlänge der Wellen • **max_wavelength** - maximale Wellenlänge der Wellen • **min_amplitude** - Mindestamplitude der Wellen • **max_amplitude** - maximale Amplitude der Wellen • **seed** - gibt einen Zufallswert für die Erzeugung von Wellen an. Wenn der Seed 0 ist, wird ein völlig zufälliger Satz von Wellen erzeugt.
Beispiele	• `square_wave_randomized(geom_from_wkt('LineString(0 0, 10 0)'), 2, 3, 0.1, 0.2)` → Quadratische Wellen zufälliger Größe mit Wellenlängen zwischen 2 und 3 und Amplituden zwischen 0,1 und 0,2 entlang des Linienstrangs

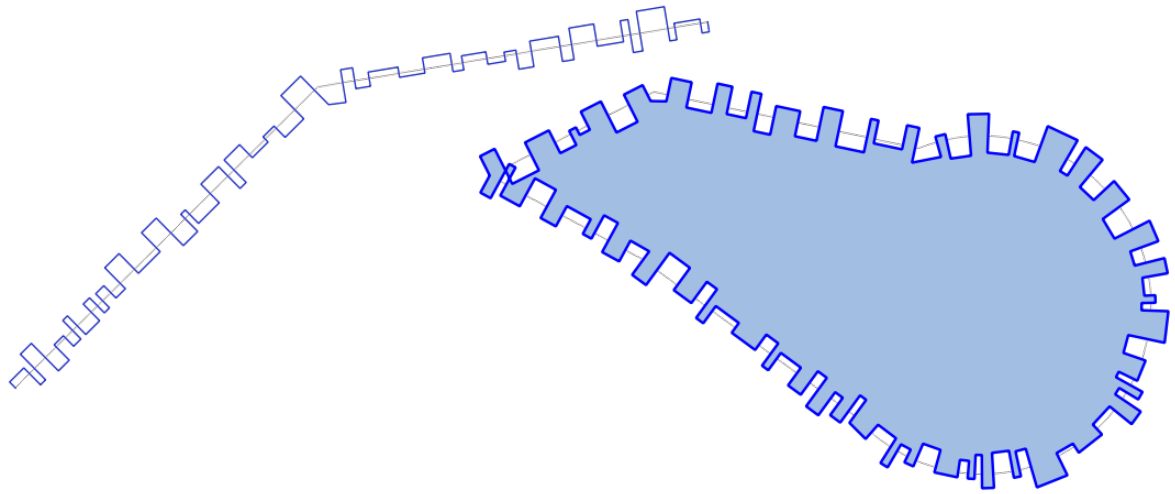


Abb. 9.33: Symbolisierung von Objekten mit quadratischen Zufallswellen

start_point

Gibt den ersten Knoten aus einer Geometrie zurück.

Syntax	start_point(geometry)
Argumente	• **geometry** - Geometrie-Objekt
Beispiele	• `geom_to_wkt(start_point(geom_from_wkt('LINESTRING(4 0, 4 2, 0 2)')))` → ‚Point (4 0)‘

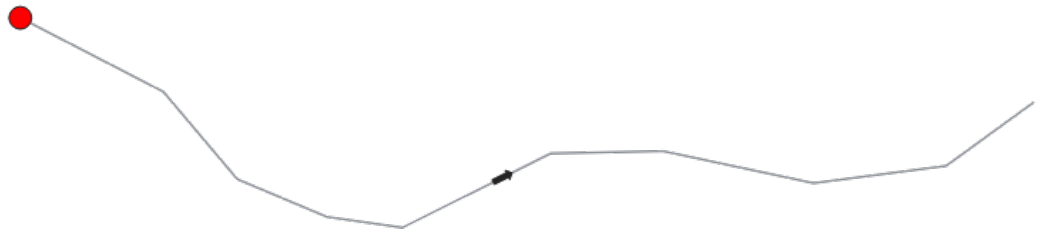


Abb. 9.34: Startpunkt eines Linienobjekts

Siehe auch: *end_point*, *Extract specific vertices* Algorithmus

straight_distance_2d

Gibt den direkten/euklidischen Abstand zwischen dem ersten und dem letzten Scheitelpunkt einer Geometrie zurück. Die Geometrie muss eine Kurve sein (circularstring, linestring).

Syntax	straight_distance_2d(geometry)
Argumente	• **geometry** - Die Geometrie.
Beispiele	• `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

Siehe auch: *length*

sym_difference

Gibt eine Geometrie zurück, die die Teile von zwei Geometrien darstellt, die sich nicht schneiden.

Syntax	sym_difference(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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)‘

Siehe auch: *Symmetrical difference*-Algorithmus

tapered_buffer

Erzeugt einen Puffer entlang einer Liniengeometrie, wobei der Pufferdurchmesser gleichmäßig über die Länge der Linie variiert.

Syntax	tapered_buffer(geometry, start_width, end_width, [segments=8]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - Eingabegeometrie. Muss eine (Multi-)Liniengeometrie sein. • **start_width** - Breite des Puffers am Anfang der Linie • **end_width** - Breite des Puffers am Ende der Linie • **segments** - Anzahl der Segmente zur Annäherung von Viertelkreiskurven im Puffer.
Beispiele	• `tapered_buffer(geometry:=geom_from_wkt('LINESTRING(1 2, 4 2)'),start_width:=1,end_width:=2,segments:=8)` → Ein sich verjüngender Puffer, der mit einem Durchmesser von 1 beginnt und mit einem Durchmesser von 2 entlang der Geometrie des Linestrings endet.

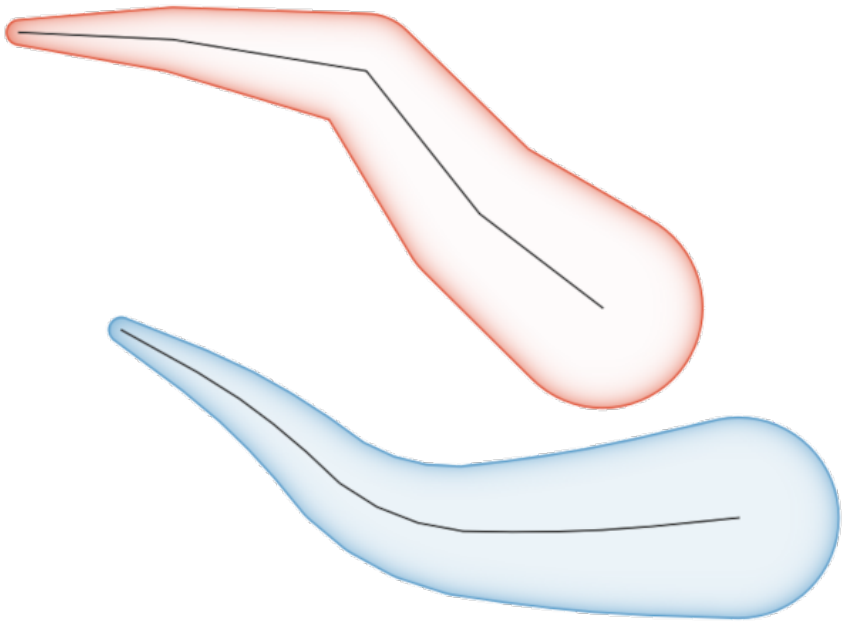


Abb. 9.35: Konische Puffer um Linien-Objekte

Siehe auch: *Tapered buffers*-Algorithmus

touches

Prüft, ob eine Geometrie eine andere berührt. Gibt TRUE zurück, wenn die Geometrien mindestens einen gemeinsamen Punkt haben, aber ihre Innenräume sich nicht überschneiden.

Syntax	touches(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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

Siehe auch: *overlay_touches*

transform

Gibt die Geometrie zurück, die von einem Quell-KBS in ein Ziel-KBS transformiert wurde.

Syntax	transform(geometry, source_auth_id, dest_auth_id)
Argumente	• **geometry** - eine Geometrie • **source_auth_id** - the source CRS definition or CRS object • **dest_auth_id** - the destination CRS definition or CRS object
Beispiele	• `geom_to_wkt( transform( make_point(488995.53240249, 7104473.38600835), 'EPSG:2154', 'EPSG:4326' ) )` → ‚POINT(0 51)‘

Siehe auch: *Reproject layer*-Algorithmus

translate

Gibt eine verschobene Version einer Geometrie zurück. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	translate(geometry, dx, dy)
Argumente	• **geometry** - eine Geometrie • **dx** - delta x • **dy** - delta y
Beispiele	• `translate(@geometry, 5, 10)` → eine Geometrie des gleichen Typs wie die ursprüngliche Geometrie

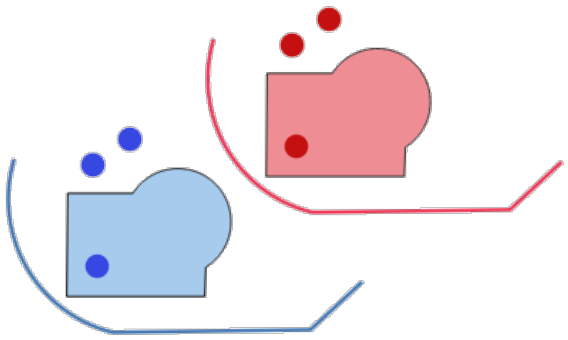


Abb. 9.36: Verschobene Objekte

Siehe auch: *Translate*-Algorithmus

triangular_wave

Konstruiert dreieckige Wellen entlang der Begrenzung einer Geometrie.

Syntax	triangular_wave(geometry, wavelength, amplitude, [strict=False]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **wavelength** - Wellenlänge der Dreieckswellenform • **amplitude** - Amplitude der Dreieckswellenform • **strict** - Standardmäßig wird das Argument Wellenlänge als „maximale Wellenlänge“ behandelt, wobei die tatsächliche Wellenlänge dynamisch angepasst wird, so dass eine genaue Anzahl von Dreieckswellen entlang der Grenzen der Geometrie erzeugt wird. Wenn das Argument strict auf true gesetzt wird, wird die Wellenlänge exakt verwendet, und es kann ein unvollständiges Muster für die endgültige Wellenform entstehen.
Beispiele	• `triangular_wave(geom_from_wkt('LineString(0 0, 10 0)'), 3, 1)` → Dreieckswellen mit einer Wellenlänge von 3 und einer Amplitude von 1 entlang des Linienstrangs

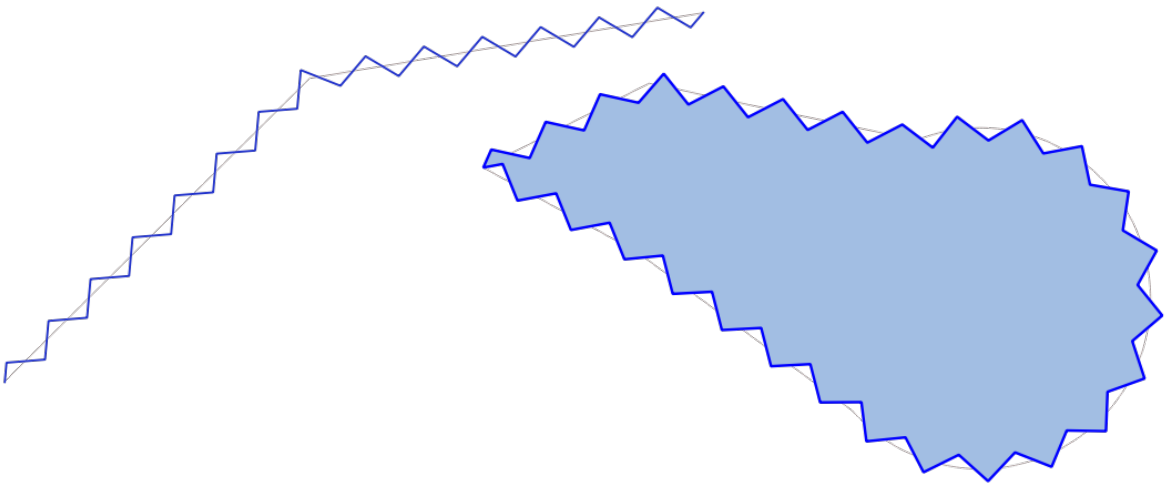


Abb. 9.37: Symbolisierung von Objekten mit Dreieckswellen

triangular_wave_randomized

Konstruiert zufällige Dreieckswellen entlang des Randes einer Geometrie.

Syntax	triangular_wave_randomized(geometry, min_wavelength, max_wavelength, min_amplitude, max_amplitude, [seed=0]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **min_wavelength** - Mindestwellenlänge der Wellen • **max_wavelength** - maximale Wellenlänge der Wellen • **min_amplitude** - Mindestamplitude der Wellen • **max_amplitude** - maximale Amplitude der Wellen • **seed** - gibt einen Zufallswert für die Erzeugung von Wellen an. Wenn der Seed 0 ist, wird ein völlig zufälliger Satz von Wellen erzeugt.
Beispiele	• `triangular_wave_randomized(geom_from_wkt('LineString(0 0, 10 0)'), 2, 3, 0.1, 0.2)` → Dreieckige Wellen mit zufälliger Größe, mit Wellenlängen zwischen 2 und 3 und Amplituden zwischen 0,1 und 0,2 entlang des Linienzugs

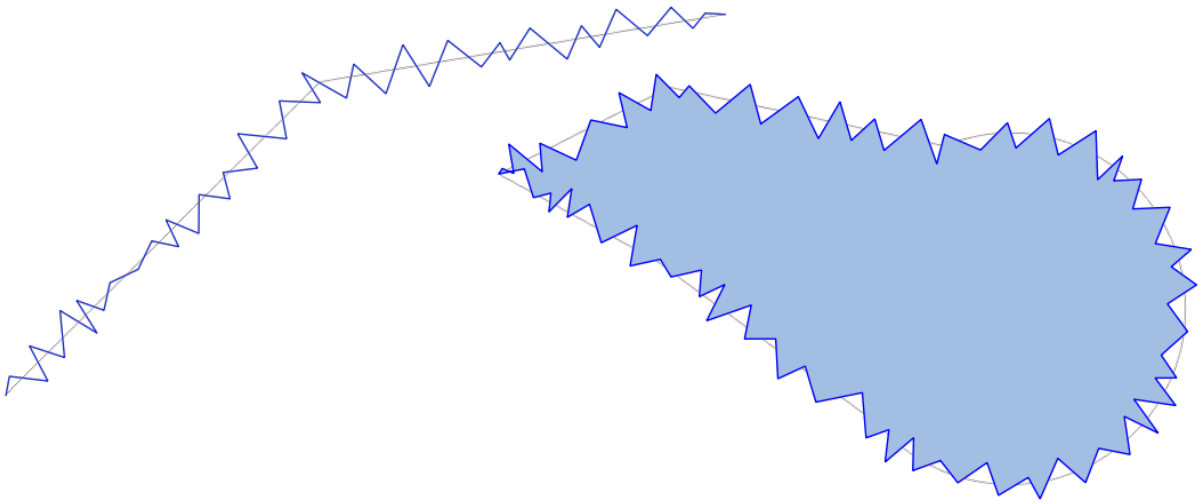


Abb. 9.38: Symbolisierung von Objekten mit dreieckigen Zufallswellen

union

Gibt eine Geometrie zurück, die die Punktmengenvereinigung der Eingabe-Geometrien darstellt.

Syntax	union(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `geom_to_wkt( union( make_point(4, 4), make_point(5, 5) ) )` → ‚MULTIPOINT(4 4, 5 5)‘

wave

Konstruiert gerundete (sinusförmige) Wellen entlang der Begrenzung einer Geometrie.

Syntax	wave(geometry, wavelength, amplitude, [strict=False]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **wavelength** - Wellenlänge der sinusförmigen Wellenform • **amplitude** - Amplitude der sinusförmigen Wellenform • **strict** - Standardmäßig wird das Argument Wellenlänge als „maximale Wellenlänge“ behandelt, wobei die tatsächliche Wellenlänge dynamisch angepasst wird, so dass eine genaue Anzahl von Wellen entlang der Grenzen der Geometrie erzeugt wird. Wenn das Strict-Argument auf true gesetzt wird, wird die Wellenlänge exakt verwendet und es kann ein unvollständiges Muster für die endgültige Wellenform entstehen.
Beispiele	• `wave(geom_from_wkt('LineString(0 0, 10 0)'), 3, 1)` → Sinusförmige Wellen mit der Wellenlänge 3 und der Amplitude 1 entlang des Linienstrangs

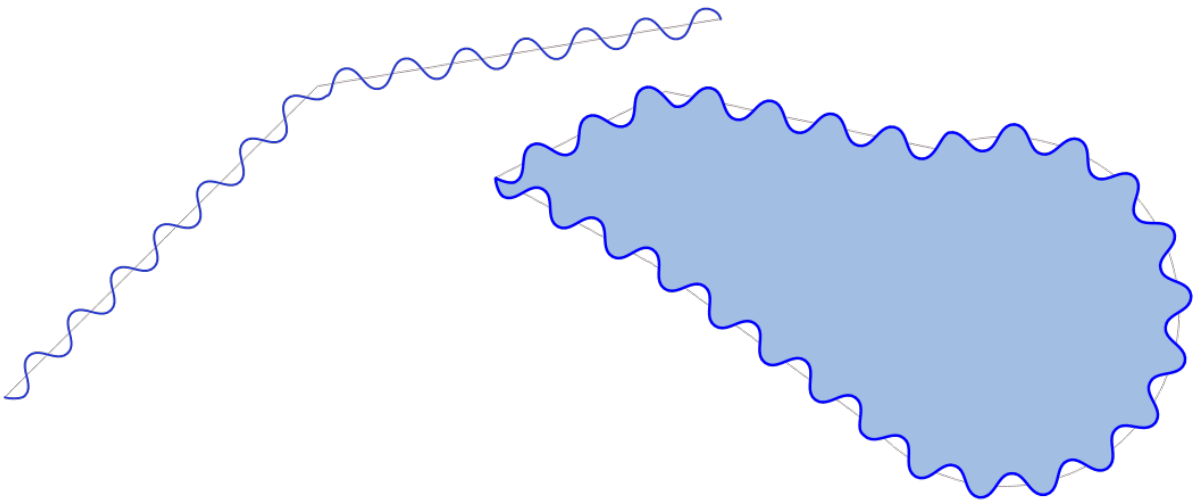


Abb. 9.39: Objekte mit Wellen symbolisieren

wave_randomized

Konstruiert zufällig gekrümmte (sinusförmige) Wellen entlang der Begrenzung einer Geometrie.

Syntax	wave_randomized(geometry, min_wavelength, max_wavelength, min_amplitude, max_amplitude, [seed=0]) [] kennzeichnet optionale Argumente
Argumente	• **geometry** - eine Geometrie • **min_wavelength** - Mindestwellenlänge der Wellen • **max_wavelength** - maximale Wellenlänge der Wellen • **min_amplitude** - Mindestamplitude der Wellen • **max_amplitude** - maximale Amplitude der Wellen • **seed** - gibt einen Zufallswert für die Erzeugung von Wellen an. Wenn der Seed 0 ist, wird ein völlig zufälliger Satz von Wellen erzeugt.
Beispiele	• `wave_randomized(geom_from_wkt('LineString(0 0, 10 0)'), 2, 3, 0.1, 0.2)` → Gekrümmte Wellen von zufälliger Größe mit Wellenlängen zwischen 2 und 3 und Amplituden zwischen 0,1 und 0,2 entlang des Linienstrangs

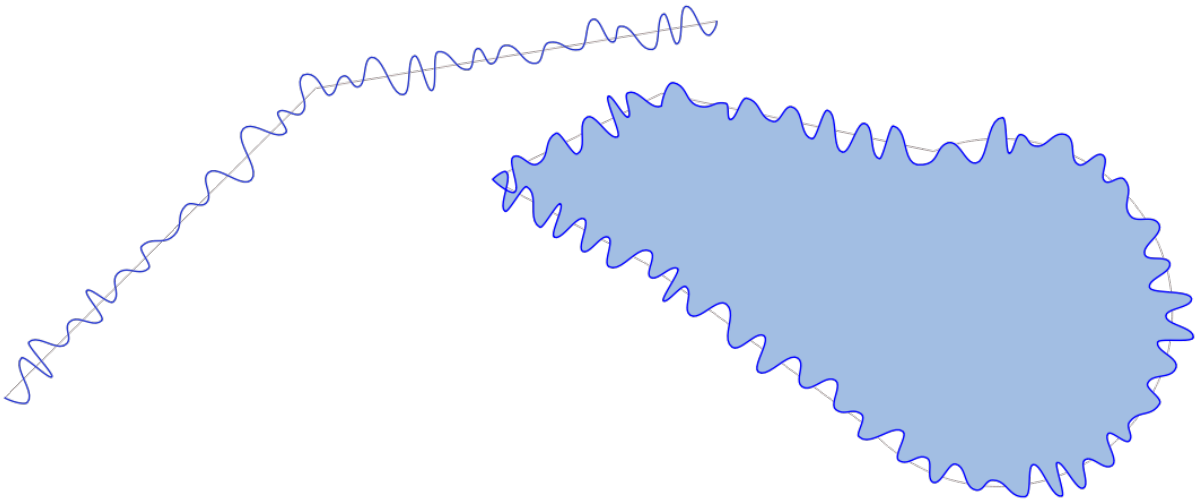


Abb. 9.40: Symbolisierung von Objekten mit randomisierten Wellen

wedge_buffer

Gibt einen keilförmigen Puffer zurück, der von einer Punktgeometrie ausgeht.

Syntax	wedge_buffer(center, azimuth, width, outer_radius, [inner_radius=0.0]) [] kennzeichnet optionale Argumente
Argumente	• **center** - Mittelpunkt (Ursprung) des Puffers. Muss eine Punktgeometrie sein. • **azimuth** - Winkel (in Grad), in dem die Mitte des Keils nach oben zeigt. • **width** - Pufferbreite (in Grad). Beachten Sie, dass sich der Keil bis zur Hälfte der Winkelbreite auf beiden Seiten der Azimutrichtung erstreckt. • **outer_radius** - Außenradius bei Puffern • **inner_radius** - optionaler Innenradius für Puffer
Beispiele	• `wedge_buffer(center:=geom_from_wkt('POINT(1 2)'), azimuth:=90,width:=180,outer_radius:=1)` → Ein keilförmiger Puffer, der auf dem Punkt (1,2) zentriert ist, nach Osten ausgerichtet ist, eine Breite von 180 Grad und einen Außenradius von 1 hat.

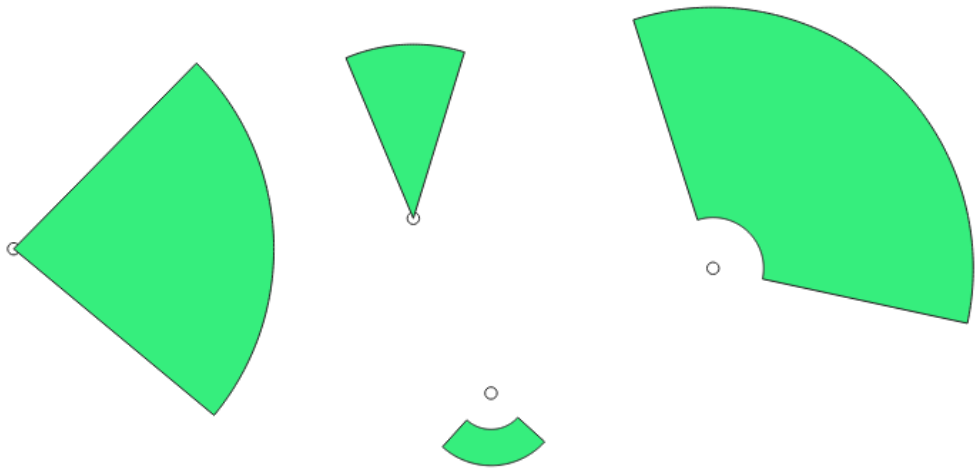


Abb. 9.41: Objekte mit Keilpufferung

Siehe auch: *Create wedge buffers*-Algorithmus

within

Prüft, ob eine Geometrie innerhalb einer anderen liegt. Gibt TRUE zurück, wenn die Geometrie1 vollständig innerhalb der Geometrie2 liegt.

Syntax	within(geometry1, geometry2)
Argumente	• **geometry1** - eine Geometrie • **geometry2** - eine weitere Geometrie
Beispiele	• `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

Siehe auch: *overlay_within*

$x

Gibt die x-Koordinate des aktuellen Punktmerkmals zurück. Handelt es sich bei dem Objekt um ein Mehrpunkt-Feature, wird die x-Koordinate des ersten Punktes zurückgegeben. **WARNUNG: Diese Funktion ist veraltet. Es wird empfohlen, stattdessen die Ersatzfunktion x() mit der Variablen @geometry zu verwenden.**

Syntax	$x
Beispiele	• `$x` → 42

Siehe auch: *x*

x

Gibt die x-Koordinate einer Punktgeometrie oder die x-Koordinate des Schwerpunkts bei einer Nicht-Punktgeometrie zurück.

Syntax	x(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `x( geom_from_wkt( 'POINT(2 5)' ) )` → 2 • `x( @geometry )` → x-Koordinate des Schwerpunkts des aktuellen Objekts

$x_at

Ermittelt eine x-Koordinate der Geometrie des aktuellen Objekts. **WARNUNG: Diese Funktion ist veraltet. Es wird empfohlen, stattdessen die Ersatzfunktion x_at mit der Variablen @geometry zu verwenden.**

Syntax	$x_at(vertex)
Argumente	• **vertex** - Index des Scheitelpunkts der aktuellen Geometrie (die Indizes beginnen bei 0; negative Werte gelten ab dem letzten Index, beginnend bei -1)
Beispiele	• `$x_at(1)` → 5

Siehe auch: *x_at*

x_at

Liefert eine x-Koordinate der Geometrie.

Syntax	x_at(geometry, vertex)
Argumente	• **geometry** - Geometrie-Objekt • **vertex** - Index des Knotenpunkts der Geometrie (die Indizes beginnen bei 0; negative Werte gelten ab dem letzten Index, beginnend bei -1)
Beispiele	• `x_at( geom_from_wkt( 'POINT(4 5)' ), 0 )` → 4

x_max

Gibt die maximale x-Koordinate einer Geometrie zurück. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	x_max(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `x_max( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 4

x_min

Gibt die minimale x-Koordinate einer Geometrie zurück. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	x_min(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `x_min( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 2

$y

Gibt die y-Koordinate des aktuellen Punkt-Features zurück. Handelt es sich bei dem Objekt um ein Mehrpunkt-Feature, wird die y-Koordinate des ersten Punktes zurückgegeben. **WARNUNG: Diese Funktion ist veraltet. Es wird empfohlen, stattdessen die Ersatzfunktion y() mit der Variablen @geometry zu verwenden.**

Syntax	$y
Beispiele	• `$y` → 42

Siehe auch: *y*

y

Gibt die y-Koordinate einer Punktgeometrie oder die y-Koordinate des Schwerpunkts bei einer Nicht-Punktgeometrie zurück.

Syntax	y(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `y( geom_from_wkt( 'POINT(2 5)' ) )` → 5 • `y( @geometry )` → y-Koordinate des Schwerpunkts des aktuellen Objekts

$y_at

Liefert eine y-Koordinate der Geometrie des aktuellen Objekts. **WARNUNG: Diese Funktion ist veraltet. Es wird empfohlen, stattdessen die Ersatzfunktion y_at mit der Variablen @geometry zu verwenden.**

Syntax	$y_at(vertex)
Argumente	• **vertex** - Index des Scheitelpunkts der aktuellen Geometrie (die Indizes beginnen bei 0; negative Werte gelten ab dem letzten Index, beginnend bei -1)
Beispiele	• `$y_at(1)` → 2

Siehe auch: *y_at*

y_at

Liefert eine y-Koordinate der Geometrie.

Syntax	y_at(geometry, vertex)
Argumente	• **geometry** - Geometrie-Objekt • **vertex** - Index des Knotenpunkts der Geometrie (die Indizes beginnen bei 0; negative Werte gelten ab dem letzten Index, beginnend bei -1)
Beispiele	• `y_at( geom_from_wkt( 'POINT(4 5)' ), 0 )` → 5

y_max

Gibt die maximale y-Koordinate einer Geometrie zurück. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	y_max(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `y_max( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 8

y_min

Gibt die minimale y-Koordinate einer Geometrie zurück. Die Berechnungen erfolgen im räumlichen Bezugssystem dieser Geometrie.

Syntax	y_min(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `y_min( geom_from_wkt( 'LINESTRING(2 5, 3 6, 4 8)') )` → 5

$z

Gibt den z-Wert des aktuellen Punktobjekts zurück, wenn es sich um ein 3D-Objekt handelt. Handelt es sich bei dem Objekt um ein Multipunkt- Objekt, dann wird der z-Wert des ersten Punktes zurückgegeben. **WARNUNG: Diese Funktion ist veraltet. Es wird empfohlen, stattdessen die Ersatzfunktion z() mit der Variablen @geometry zu verwenden.**

Syntax	$z
Beispiele	• `$z` → 123

z

Gibt die z-Koordinate einer Punktgeometrie zurück, oder NULL, wenn die Geometrie keinen z-Wert hat.

Syntax	z(geometry)
Argumente	• **geometry** - eine Punktgeometrie
Beispiele	• `z( geom_from_wkt( 'POINTZ(2 5 7)' ) )` → 7

z_at

Liefert eine z-Koordinate der Geometrie oder NULL, wenn die Geometrie keinen z-Wert hat.

Syntax	z_at(geometry, vertex)
Argumente	• **geometry** - Geometrie-Objekt • **vertex** - Index des Knotenpunkts der Geometrie (die Indizes beginnen bei 0; negative Werte gelten ab dem letzten Index, beginnend bei -1)
Beispiele	• `z_at(geom_from_wkt('LineStringZ(0 0 0, 10 10 5, 10 10 0)'), 1)` → 5

z_max

Gibt die maximale z-Koordinate einer Geometrie zurück, oder NULL, wenn die Geometrie keinen z-Wert hat.

Syntax	z_max(geometry)
Argumente	• **geometry** - eine Geometrie mit z-Koordinate
Beispiele	• `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

Gibt die minimale z-Koordinate einer Geometrie zurück, oder NULL, wenn die Geometrie keinen z-Wert hat.

Syntax	z_min(geometry)
Argumente	• **geometry** - eine Geometrie mit z-Koordinate
Beispiele	• `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 Layout Funktionen

Diese Gruppe enthält Funktionen zur Bearbeitung der Eigenschaften von Drucklayout-Elementen.

item_variables

Liefert eine Zuordnung von Variablen aus einem Layoutelement innerhalb dieses Drucklayouts.

Syntax	item_variables(id)
Argumente	• **id** - Layoutelement-ID
Beispiele	• `map_get( item_variables('Map 0'), 'map_scale')` → Maßstab des Elements „Map 0“ im aktuellen Drucklayout

Siehe auch: Liste der standardmäßigen *Variablen*

map_credits

Gibt eine Liste von Credit-Strings (Nutzungsrechte) für die in einem Layout angezeigten Layer oder ein bestimmtes Layout-Kartenelement zurück.

Syntax	map_credits([id], [include_layer_names=false], [layer_name_separator=': ']) [] kennzeichnet optionale Argumente
Argumente	• **id** - Kartenelement-ID. Wenn nicht angegeben, werden die Layer aus allen Karten im Layout verwendet.. • **include_layer_names** - Auf true setzen, um Layer-Namen vor ihren Credit-Strings einzuschließen • **layer_name_separator** - Zeichenkette, die zwischen den Namen der Layer und ihren Credit-Strings eingefügt wird, wenn include_layer_names true ist
Beispiele	• `array_to_string( map_credits() )` → durch Kommata getrennte Liste der Layer-Credits für alle Layer, die in allen Kartenelementen im Layout angezeigt werden, z. B. „CC-BY-NC, CC-BY-SA“. • `array_to_string( map_credits( 'Main Map' ) )` → kommagetrennte Liste der Layer-Credits für die im Layoutelement „Hauptkarte“ angezeigten Layer, z. B. „CC-BY-NC, CC-BY-SA“. • `array_to_string( map_credits( 'Main Map', include_layer_names := true, layer_name_separator := ': ' ) )` → kommagetrennte Liste von Layernamen und deren Credits für Layer, die im Layoutelement „Hauptkarte“ angezeigt werden, z. B. „Eisenbahnlinien: CC-BY-NC, Basemap: CC-BY-SA

Diese Funktion setzt voraus, dass die *Zugriffs-Metadateneigenschaften* der Layer gefüllt sind.

9.2.16 Kartenlayer

Diese Gruppe enthält eine Liste der im aktuellen Projekt verfügbaren Layer und für jeden Layer die zugehörigen Felder (im Datensatz gespeicherte, virtuelle oder zusätzliche Felder sowie Felder aus Verknüpfungen). Mit den Feldern kann auf die gleiche Weise interagiert werden wie in *Felder und Werte*, mit dem Unterschied, dass ein Doppelklick den Namen als String (in einfachen Anführungszeichen) zum Ausdruck hinzufügt und nicht als Feldreferenz, da sie nicht zum aktiven Layer gehören. Dies bietet eine bequeme Möglichkeit, Ausdrücke zu schreiben, die sich auf verschiedene Layer beziehen, z.B. bei der Ausführung von *Aggregate-*, *Attribut-* oder *Räumlichen* Abfragen.

Es bietet auch einige nützliche Funktionen zur Bearbeitung von Layern.

decode_uri

Nimmt einen Layer und dekodiert die uri des zugrunde liegenden Datenanbieters. Es hängt vom Datenanbieter ab, welche Daten verfügbar sind.

Syntax	decode_uri(layer, [part]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - Der Layer, für den die uri dekodiert werden soll. • **part** - Der Teil der uri, der zurückgegeben werden soll. Wenn nichts angegeben ist, wird eine Map mit allen uri-Teilen zurückgegeben.
Beispiele	• `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

Gibt eine passende Layer-Eigenschaft oder einen Metadatenwert zurück.

Syntax	layer_property(layer, property)
Argumente	• **layer** - eine Zeichenkette, die entweder einen Layernamen oder eine Layer-ID darstellt • **property** - eine Zeichenkette, die der zurückzugebenden Eigenschaft entspricht. Gültige Optionen sind: – name: Layer-Name – id: Layer-ID – title: Metadaten Titelstring – abstract: Metadata Zusammenfassung – keywords: Metadaten-Schlüsselwörter – data_url: Metadaten-URL – attribution: Metadaten-Attribut-String – attribution_url: Metadaten-Attribut-URL – source: Layer-Quelle – min_scale: minimaler Anzeigemaßstab für Layer – max_scale: maximaler Anzeigemaßstab für Layer – is_editable: ob sich der Layer im Bearbeitungsmodus befindet – crs: Layer KBS – crs_definition: Layer KBS vollständige Definition – crs_description: Layer KBS Beschreibung – crs_ellipsoid: Abkürzung des Layers KBS-Ellipsoid – extent: Ausdehnung des Layers (als Geometrieobjekt) – distance_units: Layer-Abstandseinheiten – type: Layer-Typ, z. B. Vektor oder Raster – storage_type: Speicherformat (nur Vektorlayer) – geometry_type: Geometrietyp, z. B. Punkt (nur Vektor-Layer) – feature_count: ungefähre Anzahl der Objekte für Layer (nur Vektorlayer) – path: Pfad zur Datenquelle des Layers. Nur für dateibasierte Layer verfügbar.
Beispiele	• `layer_property('streets','title')` → ‚Basemap Streets‘ • `layer_property('airports','feature_count')` → 120 • `layer_property('landsat','crs')` → ‚EPSG:4326‘

Siehe auch: *Vektor-*, *Raster-* und *Netz-* Layer-Eigenschaften

load_layer

Lädt einen Layer nach Quell-URI und Anbieternamen.

Syntax	load_layer(uri, provider)
Argumente	• **uri** - Layer-Quelle URI-String • **provider** - Layer Datenanbieter Name
Beispiele	• `layer_property(load_layer('c:/data/roads.shp', 'ogr'), 'feature_count')` → Anzahl der Objekte aus dem Vektor-Layer c:/data/roads.shp

9.2.17 Zuordnungs-Funktionen

Diese Gruppe enthält Funktionen zum Erstellen oder Bearbeiten von Schlüsseln und Werten der Map-Datenstrukturen (auch bekannt als Wörterbuchobjekte, Schlüssel-Wert-Paare oder assoziative Arrays). Anders als bei den *Array-Funktionen*, bei der die Reihenfolge der Werte eine Rolle spielt, ist die Reihenfolge der Schlüssel-Wert-Paare im Map-Objekt nicht relevant, da die Werte durch ihre Schlüssel identifiziert werden.

from_json

Lädt eine JSON-formatierte Zeichenkette.

Syntax	from_json(string)
Argumente	• **string** - JSON-Zeichenkette
Beispiele	• `from_json('{"1":"one","2":"two"}')` → { ‚1': ‚one', ‚2': ‚two' } • `from_json('[1,2,3]')` → [1,2,3]

hstore_to_map

Erzeugt eine Zuordnung aus einer hstore-formatierten Zeichenkette.

Syntax	hstore_to_map(string)
Argumente	• **string** - die Eingabezeichenkette
Beispiele	• `hstore_to_map('qgis=>rocks')` → { ‚qgis': ‚rocks' }

map

Gibt eine Zuordnung zurück, die alle Schlüssel und Werte enthält, die als Paar von Parametern übergeben wurden.

Syntax	map(key1, value1, key2, value2, …)
Argumente	• **key** - a key (string) • **value** - ein Wert
Beispiele	• `map('1','one','2', 'two')` → { ‚1': ‚one', ‚2': ‚two' } • `map('1','one','2', 'two')['1']` → ‚one'

map_akeys

Gibt alle Schlüssel einer Zuordnung als Array zurück.

Syntax	map_akeys(map)
Argumente	• **map** - eine Zuordnung
Beispiele	• `map_akeys(map('1','one','2','two'))` → [‚1‘, ‚2‘]

map_avals

Gibt alle Werte einer Zuordnung als Array zurück.

Syntax	map_avals(map)
Argumente	• **map** - eine Zuordnung
Beispiele	• `map_avals(map('1','one','2','two'))` → [‚one‘, ‚two‘]

map_concat

Gibt eine Zuordnung zurück, die alle Einträge der angegebenen Zuordnungen enthält. Wenn zwei Zuordnungen den gleichen Schlüssel enthalten, wird der Wert der zweiten Zuordnung genommen.

Syntax	map_concat(map1, map2, …)
Argumente	• **map** - eine Zuordnung
Beispiele	• `map_concat(map('1','one', '2','overridden'),map('2','two', '3','three'))` → { ‚1‘: ‚one‘, ‚2‘: ‚two‘, ‚3‘: ‚three‘ }

map_delete

Gibt eine Zuordnung zurück, in der der angegebenen Schlüssel und der dazugehörige Wert gelöscht wurden.

Syntax	map_delete(map, key)
Argumente	• **map** - eine Zuordnung • **key** - der zu löschende Schlüssel
Beispiele	• `map_delete(map('1','one','2','two'),'2')` → { ‚1‘: ‚one‘ }

map_exist

Gibt TRUE zurück, wenn der angegebene Schlüssel in der Zuordnung existiert.

Syntax	map_exist(map, key)
Argumente	• **map** - eine Zuordnung • **key** - der Schlüssel zum Nachschlagen
Beispiele	• `map_exist(map('1','one','2','two'),'3')` → FALSE

map_get

Gibt den Wert einer Zuordnung mit ihrem Schlüssel zurück. Gibt NULL zurück, wenn der Schlüssel nicht vorhanden ist.

Syntax	map_get(map, key)
Argumente	• **map** - eine Zuordnung • **key** - der Schlüssel zum Nachschlagen
Beispiele	• `map_get(map('1','one','2','two'),'2')` → ‚two' • `map_get( item_variables('Map 0'), 'map_scale')` → Maßstab des Elements „Map 0" (falls vorhanden) im aktuellen Drucklayout

Hinweis

Sie können auch den *Index-Operator ([])* verwenden, um einen Wert aus einer Zuordnung zu erhalten.

map_insert

Gibt eine Zuordnung mit einem hinzugefügten Schlüssel/Wert zurück. Wenn der Schlüssel bereits existiert, wird sein Wert überschrieben.

Syntax	map_insert(map, key, value)
Argumente	• **map** - eine Zuordnung • **key** - der Schlüssel zum Hinzufügen • **value** - der hinzuzufügende Wert
Beispiele	• `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

Gibt eine Zuordnung mit allen Schlüsseln zurück, denen eine bestimmte Zeichenkette vorangestellt ist.

Syntax	map_prefix_keys(map, prefix)
Argumente	• **map** - eine Zuordnung • **prefix** - eine Zeichenkette
Beispiele	• `map_prefix_keys(map('1','one','2','two'), 'prefix-')` → { ‚prefix-1': ‚one', ‚prefix-2': ‚two' }

map_to_hstore

Zusammenführen von Zuordnungselementen in eine hstore-formatierte Zeichenkette.

Syntax	map_to_hstore(map)
Argumente	• **map** - die Eingabe-Zuordnung
Beispiele	• `map_to_hstore(map('1','one','2','two'))` → „„1“=>“one““,„„2“=>“two““

map_to_html_dl

Zusammenführen von Zuordnungselementen in einen HTML-Definitionslisten-String.

Syntax	map_to_html_dl(map)
Argumente	• **map** - die Eingabe-Zuordnung
Beispiele	• `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

Zusammenführen von Zuordnungselementen in einen HTML-Tabellenstring.

Syntax	map_to_html_table(map)
Argumente	• **map** - die Eingabe-Zuordnung
Beispiele	• `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

Erstellen einer JSON-formatierten Zeichenkette aus einer Zuordnung, einem Array oder einem anderen Wert.

Syntax	to_json(value)
Argumente	• **value** - der Eingabewert
Beispiele	• `to_json(map('1','one','2','two'))` → {„1“:“one“,“2“:“two“} • `to_json(array(1,2,3))` → [1,2,3]

url_encode

Gibt eine URL-kodierte Zeichenkette aus einer Zuordnung zurück. Wandelt alle Zeichen in ihre korrekt kodierte Form um und erzeugt eine vollständig konforme Abfragezeichenfolge.

Beachten Sie, dass das Pluszeichen ‚+‘ nicht umgewandelt wird..

Syntax	url_encode(map)
Argumente	• **map** - eine Zuordnung
Beispiele	• `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 Mathematische Funktionen

Diese Gruppe enthält mathematische Funktionen (z.B. sqrt, sin und cos).

abs

Gibt den absoluten Wert einer Zahl zurück.

Syntax	abs(value)
Argumente	• **value** - eine Zahl
Beispiele	• `abs(-2)` → 2

acos

Gibt den inversen Kosinus eines Wertes im Bogenmaß zurück.

Syntax	acos(value)
Argumente	• **value** - Kosinus eines Winkels im Bogenmaß
Beispiele	• `acos(0.5)` → 1.0471975511966

asin

Gibt den inversen Sinus eines Wertes im Bogenmaß zurück.

Syntax	asin(value)
Argumente	• **value** - Sinus eines Winkels im Bogenmaß
Beispiele	• `asin(1.0)` → 1.5707963267949

atan

Gibt den inversen Tangens eines Wertes im Bogenmaß zurück.

Syntax	atan(value)
Argumente	• **value** - Tangens eines Winkels im Bogenmaß
Beispiele	• `atan(0.5)` → 0.463647609000806

atan2

Gibt den inversen Tangens von dy/dx zurück, indem die Vorzeichen der beiden Argumente verwendet werden, um den Quadranten des Ergebnisses zu bestimmen.

Syntax	atan2(dy, dx)
Argumente	• **dy** - Differenz der Y-Koordinaten • **dx** - Differenz der X–Koordinate
Beispiele	• `atan2(1.0, 1.732)` → 0.523611477769969

ceil

Rundet eine Zahl auf.

Syntax	ceil(value)
Argumente	• **value** - eine Zahl
Beispiele	• `ceil(4.9)` → 5 • `ceil(-4.9)` → -4

clamp

Begrenzt einen Eingabewert auf einen bestimmten Bereich.

Syntax	clamp(minimum, input, maximum)
Argumente	• **minimum** - der kleinste Wert, den *input* annehmen darf • **input** - einen Wert, der auf den durch *minimum* und *maximum* angegebenen Bereich beschränkt wird • **maximum** - der größte Wert, den *input* annehmen darf
Beispiele	• `clamp(1,5,10)` → 5 *input* liegt zwischen 1 und 10 und wird daher unverändert zurückgegeben • `clamp(1,0,10)` → 1 *input* ist kleiner als der Mindestwert von 1, also gibt die Funktion 1 zurück • `clamp(1,11,10)` → 10 *input* ist größer als der Maximalwert von 10, also gibt die Funktion 10 zurück

cos

Gibt den Kosinus eines Winkels zurück.

Syntax	cos(angle)
Argumente	• **angle** - Winkel im Bogenmaß
Beispiele	• `cos(1.571)` → 0.000796326710733263

degrees

Konvertiert von Bogenmaß in Grad.

Syntax	degrees(radians)
Argumente	• **radians** - numerischer Wert
Beispiele	• `degrees(3.14159)` → 180 • `degrees(1)` → 57.2958

exp

Gibt den Exponentialwert eines Wertes zurück.

Syntax	exp(value)
Argumente	• **value** - Basis zum Exponentialwert
Beispiele	• `exp(1.0)` → 2.71828182845905

floor

Rundet eine Zahl ab.

Syntax	floor(value)
Argumente	• **value** - eine Zahl
Beispiele	• `floor(4.9)` → 4 • `floor(-4.9)` → -5

ln

Gibt den natürlichen Logarithmus eines Wertes zurück.

Syntax	ln(value)
Argumente	• **value** - numerischer Wert
Beispiele	• `ln(1)` → 0 • `ln(2.7182818284590452354)` → 1

log

Gibt den Wert des Logarithmus des übergebenen Wertes und der Basis zurück.

Syntax	log(base, value)
Argumente	• **base** - jede positive Zahl • **value** - jede positive Zahl
Beispiele	• `log(2, 32)` → 5 • `log(0.5, 32)` → -5

log10

Gibt den Wert des Logarithmus zur Basis 10 des übergebenen Ausdrucks zurück.

Syntax	log10(value)
Argumente	• **value** - jede positive Zahl
Beispiele	• `log10(1)` → 0 • `log10(100)` → 2

max

Gibt den größten Wert aus einer Menge von Werten zurück.

Syntax	max(value1, value2, …)
Argumente	• **value** - eine Zahl
Beispiele	• `max(2,10.2,5.5)` → 10.2 • `max(20.5,NULL,6.2)` → 20.5

min

Gibt den kleinsten Wert aus einer Menge von Werten zurück.

Syntax	min(value1, value2, …)
Argumente	• **value** - eine Zahl
Beispiele	• `min(20.5,10,6.2)` → 6.2 • `min(2,-10.3,NULL)` → -10.3

pi

Gibt den Wert von pi für Berechnungen zurück.

Syntax	pi()
Beispiele	• `pi()` → 3.14159265358979

radians

Konvertiert von Grad in Bogenmaß.

Syntax	radians(degrees)
Argumente	• **degrees** - numerischer Wert
Beispiele	• `radians(180)` → 3.14159 • `radians(57.2958)` → 1

rand

Gibt eine zufällige ganze Zahl innerhalb des durch das Minimum- und Maximum-Argument (einschließlich) angegebenen Bereichs zurück. Wenn ein Seed angegeben wird, ist der zurückgegebene Wert immer derselbe, abhängig vom Seed.

Syntax	rand(min, max, [seed=NULL]) [] kennzeichnet optionale Argumente
Argumente	• **min** - eine ganze Zahl, die die kleinstmögliche gewünschte Zufallszahl darstellt • **max** - eine ganze Zahl, die die größtmögliche gewünschte Zufallszahl darstellt • **seed** - beliebiger Wert, der als Seed verwendet wird
Beispiele	• `rand(1, 10)` → 8

randf

Gibt eine zufällige Fließkommazahl innerhalb des durch das Minimum- und Maximum-Argument (einschließlich) angegebenen Bereichs zurück. Wenn ein Seed angegeben wird, ist der zurückgegebene Wert immer derselbe, abhängig vom Seed.

Syntax	randf([min=0.0], [max=1.0], [seed=NULL]) [] kennzeichnet optionale Argumente
Argumente	• **min** - ein Float, der die kleinstmögliche gewünschte Zufallszahl darstellt • **max** - ein Float, der die größtmögliche gewünschte Zufallszahl darstellt • **seed** - beliebiger Wert, der als Seed verwendet wird
Beispiele	• `randf(1, 10)` → 4.59258286403147

round

Rundet eine Zahl auf die Anzahl der Dezimalstellen.

Syntax	round(value, [places=0]) [] kennzeichnet optionale Argumente
Argumente	• **value** - zu rundende Dezimalzahl • **places** - Optionale Ganzzahl, die die Anzahl der Stellen angibt, auf die Dezimalzahlen gerundet werden sollen. Kann negativ sein, um Vorkommastellen, zu runden.
Beispiele	• `round(1234.567, 2)` → 1234.57 • `round(1234.567)` → 1235 • `round(1234.567, -1)` → 1230

scale_exponential

Transformiert einen gegebenen Wert aus einem Eingabebereich in einen Ausgabebereich unter Verwendung einer Exponentialkurve. Diese Funktion kann verwendet werden, um Werte im angegebenen Ausgabebereich ansteigen oder auslaufen zu lassen.

Syntax	scale_exponential(value, domain_min, domain_max, range_min, range_max, exponent)
Argumente	• **value** - Ein Wert im Eingabebereich. Die Funktion gibt einen entsprechenden skalierten Wert im Ausgabebereich zurück. • **domain_min** - Gibt den Mindestwert im Eingabebereich an, den kleinsten Wert, den der Eingabewert annehmen soll. • **domain_max** - Gibt den Höchstwert im Eingabebereich an, den größten Wert, den der Eingabewert annehmen soll. • **range_min** - Gibt den Minimalwert im Ausgabebereich an, also den kleinsten Wert, der von der Funktion ausgegeben werden soll. • **range_max** - Gibt den Maximalwert im Ausgabebereich an, also den größten Wert, der von der Funktion ausgegeben werden soll. • **exponent** - Ein positiver Wert (größer als 0), der vorgibt, wie die Eingabewerte auf den Ausgabebereich abgebildet werden. Große Exponenten bewirken, dass die Ausgabewerte langsam ansteigen, bevor sie sich beschleunigen, wenn sich die Eingabewerte dem Maximum des Bereichs nähern. Kleinere Exponenten (kleiner als 1) bewirken, dass die Ausgabewerte „auslaufen“, d. h. die Zuordnung beginnt schnell, wird aber langsamer, wenn sie sich dem Bereichsmaximum nähert.
Beispiele	• `scale_exponential(5,0,10,0,100,2)` → 3.030 ansteigend, mit einem Exponenten von 2 • `scale_exponential(3,0,10,0,100,0.5)` → 87.585 absteigend, mit einem Exponenten von 0,5

scale_linear

Transformiert einen gegebenen Wert aus einem Eingabebereich in einen Ausgabebereich durch lineare Interpolation.

Syntax	scale_linear(value, domain_min, domain_max, range_min, range_max)
Argumente	• **value** - Ein Wert im Eingabebereich. Die Funktion gibt einen entsprechenden skalierten Wert im Ausgabebereich zurück. • **domain_min** - Gibt den Mindestwert im Eingabebereich an, den kleinsten Wert, den der Eingabewert annehmen soll. • **domain_max** - Gibt den Höchstwert im Eingabebereich an, den größten Wert, den der Eingabewert annehmen soll. • **range_min** - Gibt den Minimalwert im Ausgabebereich an, also den kleinsten Wert, der von der Funktion ausgegeben werden soll. • **range_max** - Gibt den Maximalwert im Ausgabebereich an, also den größten Wert, der von der Funktion ausgegeben werden soll.
Beispiele	• `scale_linear(5,0,10,0,100)` → 50 • `scale_linear(0.2,0,1,0,360)` → 72 Skalierung eines Wertes zwischen 0 und 1 auf einen Winkel zwischen 0 und 360 • `scale_linear(1500,1000,10000,9,20)` → 9.6111111 Skalierung einer Population, die zwischen 1.000 und 10.000 schwankt, auf eine Schriftgröße zwischen 9 und 20

scale_polynomial

Transformiert einen gegebenen Wert aus einem Eingabebereich in einen Ausgabebereich unter Verwendung einer Polynomkurve. Diese Funktion kann verwendet werden, um Werte im angegebenen Ausgabebereich ansteigen oder auslaufen zu lassen.

Syntax	scale_polynomial(value, domain_min, domain_max, range_min, range_max, exponent)
Argumente	• **value** - Ein Wert im Eingabebereich. Die Funktion gibt einen entsprechenden skalierten Wert im Ausgabebereich zurück. • **domain_min** - Gibt den Mindestwert im Eingabebereich an, den kleinsten Wert, den der Eingabewert annehmen soll. • **domain_max** - Gibt den Höchstwert im Eingabebereich an, den größten Wert, den der Eingabewert annehmen soll. • **range_min** - Gibt den Minimalwert im Ausgabebereich an, also den kleinsten Wert, der von der Funktion ausgegeben werden soll. • **range_max** - Gibt den Maximalwert im Ausgabebereich an, also den größten Wert, der von der Funktion ausgegeben werden soll. • **exponent** - Ein positiver Wert (größer als 0), der vorgibt, wie die Eingabewerte auf den Ausgabebereich abgebildet werden. Große Exponenten bewirken, dass die Ausgabewerte langsam ansteigen, bevor sie sich beschleunigen, wenn sich die Eingabewerte dem Maximum des Bereichs nähern. Kleinere Exponenten (kleiner als 1) bewirken, dass die Ausgabewerte „auslaufen“, d. h. die Zuordnung beginnt schnell, wird aber langsamer, wenn sie sich dem Bereichsmaximum nähert.
Beispiele	• `scale_polynomial(5,0,10,0,100,2)` → 25 ansteigend, mit einem Exponenten von 2 • `scale_polynomial(3,0,10,0,100,0.5)` → 54.772 absteigend, mit einem Exponenten von 0,5

sin

Gibt den Sinus eines Winkels zurück.

Syntax	sin(angle)
Argumente	• **angle** - Winkel im Bogenmaß
Beispiele	• `sin(1.571)` → 0.999999682931835

sqrt

Gibt die Quadratwurzel eines Wertes zurück.

Syntax	sqrt(value)
Argumente	• **value** - eine Zahl
Beispiele	• `sqrt(9)` → 3

tan

Gibt den Tangens eines Winkels zurück.

Syntax	tan(angle)
Argumente	• **angle** - Winkel im Bogenmaß
Beispiele	• `tan(1.0)` → 1.5574077246549

9.2.19 Netz-Layer-Funktionen

Diese Gruppe enthält Funktionen, die Werte für Netz-Layer berechnen oder zurückgeben.

$face_area

Gibt die Fläche des aktuellen Netz-Layers zurück. Die von dieser Funktion berechnete Fläche berücksichtigt sowohl die Ellipsoid- als auch die Flächeneinheit-Einstellungen des aktuellen Projekts. Wenn zum Beispiel ein Ellipsoid für das Projekt festgelegt wurde, ist die berechnete Fläche ellipsoidisch, und wenn kein Ellipsoid festgelegt wurde, ist die berechnete Fläche planimetrisch.

Syntax	$face_area
Beispiele	• `$face_area` → 42

$face_index

Gibt den Index der aktuellen Netzfläche zurück.

Syntax	$face_index
Beispiele	• `$face_index` → 4581

$vertex_as_point

Gibt den aktuellen Knotenpunkt als eine Punktgeometrie zurück.

Syntax	$vertex_as_point
Beispiele	• `geom_to_wkt( $vertex_as_point )` → ‚POINT(800 1500 41)‘

$vertex_index

Gibt den Index des aktuellen Netzknotens zurück.

Syntax	$vertex_index
Beispiele	• `$vertex_index` → 9874

$vertex_x

Gibt die X-Koordinate des aktuellen Netzknotens zurück.

Syntax	$vertex_x
Beispiele	• `$vertex_x` → 42.12

$vertex_y

Gibt die Y-Koordinate des aktuellen Netzknotens zurück.

Syntax	$vertex_y
Beispiele	• `$vertex_y` → 12.24

$vertex_z

Gibt den Z-Wert des aktuellen Netzknotens zurück.

Syntax	$vertex_z
Beispiele	• `$vertex_z` → 42

9.2.20 Operatoren

Diese Gruppe enthält Operatoren (z. B. +, -, *). Beachten Sie, dass bei den meisten der folgenden mathematischen Funktionen das Ergebnis NULL ist, wenn eine der Eingaben NULL ist.

%

Rest der Division. Nimmt das Vorzeichen der Dividende.

Syntax	a % b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `9 % 2` → 1 • `9 % -2` → 1 • `-9 % 2` → -1 • `5 % NULL` → NULL

*

Multiplikation von zwei Werten

Syntax	a * b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `5 * 4` → 20 • `5 * NULL` → NULL

+

Addition von zwei Werten. Wenn einer der Werte NULL ist, ist das Ergebnis auch NULL.

Syntax	a + b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `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

Siehe auch: *concat*, ||

-

Subtraktion von zwei Werten. Wenn einer der Werte NULL ist, ist das Ergebnis auch NULL.

Syntax	a - b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `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

/

Division von zwei Werten

Syntax	a / b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `5 / 4` → 1.25 • `5 / NULL` → NULL

//

Abrundungs-Teilung von zwei Werten

Syntax	a // b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `9 // 2` → 4

<

Vergleicht zwei Werte und gibt 1 aus, wenn der linke Wert kleiner als der rechte Wert ist.

Syntax	a < b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `5 < 4` → FALSE • `5 < 5` → FALSE • `4 < 5` → TRUE

<=

Vergleicht zwei Werte und gibt 1 aus, wenn der linke Wert kleiner oder gleich dem rechten Wert ist.

Syntax	a <= b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `5 <= 4` → FALSE • `5 <= 5` → TRUE • `4 <= 5` → TRUE

<>

Vergleicht zwei Werte und wertet 1 aus, wenn sie nicht gleich sind.

Syntax	a <> b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `5 <> 4` → TRUE • `4 <> 4` → FALSE • `5 <> NULL` → NULL • `NULL <> NULL` → NULL

=

Vergleicht zwei Werte und gibt den Wert 1 zurück, wenn sie gleich sind.

Syntax	a = b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `5 = 4` → FALSE • `4 = 4` → TRUE • `5 = NULL` → NULL • `NULL = NULL` → NULL

>

Vergleicht zwei Werte und gibt 1 zurück, wenn der linke Wert größer als der rechte Wert ist.

Syntax	a > b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `5 > 4` → TRUE • `5 > 5` → FALSE • `4 > 5` → FALSE

>=

Vergleicht zwei Werte und gibt 1 zurück, wenn der linke Wert größer oder gleich dem rechten Wert ist.

Syntax	a >= b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `5 >= 4` → TRUE • `5 >= 5` → TRUE • `4 >= 5` → FALSE

AND

Gibt TRUE zurück, wenn die Bedingungen a und b erfüllt sind.

Syntax	a AND b
Argumente	• **a** - Bedingung • **b** - Bedingung
Beispiele	• `TRUE AND TRUE` → TRUE • `TRUE AND FALSE` → FALSE • `4 = 2+2 AND 1 = 1` → TRUE • `4 = 2+2 AND 1 = 2` → FALSE

BETWEEN

Gibt TRUE zurück, wenn der Wert innerhalb des angegebenen Bereichs liegt. Der Bereich wird als einschließlich der Grenzen betrachtet. Zum Testen auf Ausschluss kann NOT BETWEEN verwendet werden.

Syntax	Wert BETWEEN untere_Grenze AND obere_Grenze
Argumente	• **value** - der Wert, der mit einem Bereich verglichen werden soll. Es kann eine Zeichenkette, eine Zahl oder ein Datum sein. • **untere_Grenze AND obere_Grenze** - Bereichsgrenzen
Beispiele	• `'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

Bemerkung

value BETWEEN lower_bound AND higher_bound ist das gleiche wie „*value* >= *lower_bound AND value* <= *higher_bound*“.

Siehe auch: *NOT BETWEEN*

ILIKE

Gibt TRUE zurück, wenn der erste Parameter ohne Berücksichtigung der Groß-/Kleinschreibung mit dem angegebenen Muster übereinstimmt. LIKE kann anstelle von ILIKE verwendet werden, um die Groß- und Kleinschreibung zu berücksichtigen. Funktioniert auch mit Zahlen.

Syntax	Zeichenfolge/Zahl ILIKE Muster
Argumente	• **string/number** - zu suchende Zeichenfolge • **pattern** - zu suchendes Muster, Sie können ‚%' als Platzhalter, ‚_' als einzelnes Zeichen und ‚\\' verwenden, um diese Sonderzeichen zu umgehen.
Beispiele	• `'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

Gibt TRUE zurück, wenn der Wert in einer Liste von Werten gefunden wurde.

Syntax	a IN b
Argumente	• **a** - ein Wert • **b** - Werteliste
Beispiele	• `'A' IN ('A','B')` → TRUE • `'A' IN ('C','B')` → FALSE

IS

Gibt TRUE zurück, wenn a dasselbe ist wie b.

Syntax	a IS b
Argumente	• **a** - beliebiger Wert • **b** - beliebiger Wert
Beispiele	• `'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, wenn Ihre Geometrie nicht NULL ist

IS NOT

Gibt TRUE zurück, wenn a nicht dasselbe ist wie b.

Syntax	a IS NOT b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `'a' IS NOT 'b'` → TRUE • `'a' IS NOT 'a'` → FALSE • `4 IS NOT 2+2` → FALSE

LIKE

Gibt TRUE zurück, wenn der erste Parameter mit dem angegebenen Muster übereinstimmt. Funktioniert auch mit Zahlen.

Syntax	Zeichenkette/Zahl LIKE Muster
Argumente	• **string/number** - Wert • **pattern** - Muster, mit dem der Wert verglichen werden soll, Sie können ‚%' als Platzhalter, ‚_' als einzelnes Zeichen und ‚\\' verwenden, um diese Sonderzeichen zu umgehen.
Beispiele	• `'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

Negiert eine Bedingung.

Syntax	NOT a
Argumente	• **a** - Bedingung
Beispiele	• `NOT 1` → FALSE • `NOT 0` → TRUE

NOT BETWEEN

Gibt TRUE zurück, wenn der Wert nicht innerhalb des angegebenen Bereichs liegt. Der Bereich wird inklusive der Grenzen betrachtet.

Syntax	Wert NOT BETWEEN untere_Grenze AND obere_Grenze
Argumente	• **value** - der Wert, der mit einem Bereich verglichen werden soll. Es kann eine Zeichenkette, eine Zahl oder ein Datum sein. • **untere_Grenze AND obere_Grenze** - Bereichsgrenzen
Beispiele	• `'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

Bemerkung

value NOT BETWEEN lower_bound AND higher_bound ist das gleiche wie“*value* < *lower_bound OR value* > *higher_bound*“.

Siehe auch: *BETWEEN*

OR

Gibt TRUE zurück, wenn Bedingung a oder b wahr ist.

Syntax	a OR b
Argumente	• **a** - Bedingung • **b** - Bedingung
Beispiele	• `4 = 2+2 OR 1 = 1` → TRUE • `4 = 2+2 OR 1 = 2` → TRUE • `4 = 2 OR 1 = 2` → FALSE

[]

Index-Operator. Gibt ein Element aus einem Array oder einem Map-Wert zurück.

Syntax	[index]
Argumente	• **index** - Array-Index oder Map-Schlüsselwert
Beispiele	• `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

Siehe auch: *array_get*, *map_get*

^

Potenz aus zwei Werten.

Syntax	a ^ b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `5 ^ 4` → 625 • `5 ^ NULL` → NULL

||

Verbindet zwei Werte zu einer Zeichenkette.

Wenn einer der Werte NULL ist, ist das Ergebnis NULL. Siehe die Funktion CONCAT für ein anderes Ergebnis.

Syntax	a || b
Argumente	• **a** - ein Wert • **b** - ein Wert
Beispiele	• `'Here' || ' and ' || 'there'` → ‚Here and there‘ • `'Nothing' || NULL` → NULL • `'Dia: ' || "Diameter"` → ‚Dia: 25‘ • `1 || 2` → ‚12‘

Siehe auch: *concat*, *+*

~

Führt eine Übereinstimmung eines regulären Ausdrucks mit einer Zeichenkette durch. Backslash-Zeichen müssen mit doppeltem Escape-Zeichen versehen werden (z. B. „\\s“ für eine Übereinstimmung mit einem Leerzeichen).

Syntax	string ~ regex
Argumente	• **string** - Ein String-Wert • **regex** - Ein regulärer Ausdruck. Schrägstriche müssen mit einem Escape versehen werden, z. B. \\d.
Beispiele	• `'hello' ~ 'll'` → TRUE • `'hello' ~ '^ll'` → FALSE • `'hello' ~ 'llo$'` → TRUE • `'abc123' ~ '\\d+'` → TRUE

Siehe auch: *regexp_match*

9.2.21 Verarbeitungs-Funktionen

Diese Gruppe enthält Funktionen, die auf Verarbeitungsalgorithmen angewendet werden.

parameter

Gibt den Wert eines Eingabeparameters eines Verarbeitungsalgorithmus zurück.

Syntax	parameter(name)
Argumente	• **name** - Name des entsprechenden Eingabeparameters
Beispiele	• `parameter('BUFFER_SIZE')` → 5.6

9.2.22 Raster-Funktionen

Diese Gruppe enthält Funktionen zur Verarbeitung von Rasterlayern.

raster_attributes

Liefert eine Zuordnung mit den Feldnamen als Schlüsseln und den Rasterattribut-Tabellenwerten als Werten aus dem Attribut-Tabelleneintrag, der dem angegebenen Rasterwert entspricht.

Syntax	raster_attributes(layer, band, value)
Argumente	• **layer** - der Name oder die ID eines Raster-Layers • **band** - die Bandnummer für die zugehörige Attributtabellenabfrage. • **value** - Rasterwert
Beispiele	• `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

Liefert Statistiken von einem Raster Layer.

Syntax	raster_statistic(layer, band, property)
Argumente	• **layer** - eine Zeichenkette, die entweder einen Namen für einen Raster-Layer oder eine Layer-ID darstellt • **band** - Ganzzahl, die die Bandnummer des Raster-Layers angibt, beginnend bei 1 • **property** - eine Zeichenkette, die der zurückzugebenden Eigenschaft entspricht. Gültige Optionen sind: – min: Mindestwert – max: Höchstwert – avg: Durchschnittswert (Mittelwert) – stdev: Standardabweichung der Werte – range: Wertebereich (max - min) – sum: Summe aller Werte aus dem Raster
Beispiele	• `raster_statistic('lc',1,'avg')` → Durchschnittswert von Band 1 aus dem ‚lc' Raster Layer • `raster_statistic('ac2010',3,'min')` → Mindestwert von Band 3 aus dem Raster-Layer ‚ac2010'.

raster_value

Gibt den Rasterwert zurück, der an dem angegebenen Punkt gefunden wurde..

Syntax	raster_value(layer, band, point)
Argumente	• **layer** - der Name oder die ID eines Raster-Layers • **band** - die Bandnummer, aus der der Wert entnommen werden soll. • **point** - Punktgeometrie (bei mehrteiligen Geometrien mit mehr als einem Teil wird ein NULL-Wert zurückgegeben)
Beispiele	• `raster_value('dem', 1, make_point(1,1))` → 25 • `raster_value('ndvi', 2, @layer_cursor_point)` → 25

9.2.23 Funktionen für Datensätze und Attribute

Diese Gruppe enthält Funktionen, die mit Datensatzkennungen arbeiten.

attribute

Gibt ein Attribut von einem Objekt zurück.

Variante 1

Gibt den Wert eines Attributs des aktuellen Objekts zurück.

Syntax	attribute(attribute_name)
Argumente	• **attribute_name** - Name des zurückzugebenden Attributs
Beispiele	• `attribute( 'name' )` → im Attribut „name“ gespeicherter Wert für das aktuelle Objekt

Variante 2

Ermöglicht die Angabe des Zielobjekts und des Attributnamens.

Syntax	attribute(feature, attribute_name)
Argumente	• **feature** - ein Objekt • **attribute_name** - Name des zurückzugebenden Attributs
Beispiele	• `attribute( @atlas_feature, 'name' )` → im Attribut „name“ gespeicherter Wert für das aktuelle Objekt im Atlas

attributes

Gibt eine Zuordnung zurück, die alle Attribute eines Objekts enthält, mit Feldnamen als Zuordnungsschlüssel.

Variante 1

Gibt eine Zuordnung aller Attribute des aktuellen Objekts zurück.

Syntax	attributes()
Beispiele	• `attributes()['name']` → im Attribut „name“ gespeicherter Wert für das aktuelle Objekt

Variante 2

Ermöglicht die Angabe des Zielobjekts.

Syntax	attributes(feature)
Argumente	• **feature** - ein Objekt
Beispiele	• `attributes( @atlas_feature )['name']` → im Attribut „name“ gespeicherter Wert für das aktuelle Objekt im Atlas

Siehe auch: *Zuordnungs-Funktionen*

$currentfeature

Gibt das aktuelle Objekt zurück, das ausgewertet wird. Dies kann zusammen mit der Funktion ‚attribute‘ verwendet werden, um Attributwerte des aktuellen Objekts auszuwerten. **WARNUNG: Diese Funktion ist veraltet. Es wird empfohlen, stattdessen die Ersatzvariable @feature zu verwenden.**

Syntax	$currentfeature
Beispiele	• `attribute( $currentfeature, 'name' )` → im Attribut „name“ gespeicherter Wert für das aktuelle Objekt

display_expression

Gibt den Anzeigeausdruck für ein bestimmtes Objekt in einem Layer zurück. Der Ausdruck wird standardmäßig ausgewertet. Kann mit null, einem oder mehreren Argumenten verwendet werden, siehe unten für Details.

Keine Parameter

Wenn die Funktion ohne Parameter aufgerufen wird, wertet sie den Anzeigeausdruck des aktuellen Objekts im aktuellen Layer aus.

Syntax	display_expression()
Beispiele	• `display_expression()` → Der Anzeigeausdruck des aktuellen Objekts im aktuellen Layer.

Ein „Objekt“-Parameter

Wenn die Funktion nur mit dem Parameter ‚feature‘ aufgerufen wird, wertet sie das angegebene Objekt aus dem aktuellen Layer aus..

Syntax	display_expression(feature)
Argumente	• **feature** - das Objekt, das ausgewertet werden soll.
Beispiele	• `display_expression(@atlas_feature)` → Der Anzeigeausdruck des aktuellen Atlasobjekts.

Layer- und Objekt-Parameter

Wenn die Funktion mit einem Layer und einem Objekt aufgerufen wird, wird das angegebene Objekt aus dem angegebenen Layer ausgewertet.

Syntax	display_expression(layer, feature, [evaluate=true]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - Der Layer (oder seine ID oder sein Name) • **feature** - das Objekt, das ausgewertet werden soll. • **evaluate** - Wenn der Ausdruck ausgewertet werden muss. Bei false wird der Ausdruck nur als String-Literal zurückgegeben (das möglicherweise später mit der Funktion ‚eval' ausgewertet werden kann).
Beispiele	• `display_expression( 'streets', get_feature_by_id('streets', 1))` → Der Anzeigeausdruck des Objekts mit der ID 1 auf dem Layer „streets". • `display_expression('a_layer_id', @feature, 'False')` → Der Anzeigeausdruck des angegebenen Objekts wurde nicht ausgewertet.

feature_id

Gibt die eindeutige ID eines Objekts zurück, oder NULL, wenn das Objekt nicht gültig ist..

Syntax	feature_id(feature)
Argumente	• **feature** - ein Objekt
Beispiele	• `feature_id( @feature )` → die ID des aktuellen Objekts

Siehe auch: *get_feature_by_id*

get_feature

Gibt das erste Objekt eines Layers zurück, das einem bestimmten Attributwert entspricht.

Einzelwert-Variante

Zusammen mit der Layer-ID werden eine einzelne Spalte und ein Wert angegeben.

Syntax	get_feature(layer, attribute, value)
Argumente	• **layer** - Layer-Name oder ID • **attribute** - Name des Attributs, das für die Übereinstimmung verwendet werden soll • **value** - Attributwert, der überprüft werden soll
Beispiele	• `get_feature('streets','name','main st')` → erstes Objekt im Layer „streets" mit dem Wert „main st" im Feld „name" gefunden

Map-Variante

Zusammen mit der Layer-ID eine Zuordnung, die die zu verwendenden Spalten (Schlüssel) und ihren jeweiligen Wert enthält.

Syntax	get_feature(layer, attribute)
Argumente	• **layer** - Layer-Name oder ID • **attribute** - Zuordnung, die die zu verwendenden Spalten- und Wertepaare enthält
Beispiele	• `get_feature('streets',map('name','main st','lane_num','4'))` → erstes Objekt im Layer „streets“ mit dem Wert „main st“ im Feld „name“ und dem Wert „4“ im Feld „lane_num“

get_feature_by_id

Gibt das Objekt mit einer ID aus einem Layer zurück.

Syntax	get_feature_by_id(layer, feature_id)
Argumente	• **layer** - Layer, Layer-Name oder Layer-ID • **feature_id** - die ID des Objekts, das zurückgegeben werden soll
Beispiele	• `get_feature_by_id('streets', 1)` → das Objekt mit der ID 1 aus dem Layer „streets“

Siehe auch: *feature_id*

$id

Gibt die Objekt-ID der aktuellen Zeile zurück. **WARNUNG: Diese Funktion ist veraltet. Es wird empfohlen, stattdessen die Ersatzvariable @id zu verwenden.**

Syntax	$id
Beispiele	• `$id` → 42

Siehe auch: *feature_id*, *get_feature_by_id*

is_attribute_valid

Gibt TRUE zurück, wenn ein bestimmtes Objekt-Attribut alle Bedingungen erfüllt.

Syntax	is_attribute_valid(attribute, [feature], [layer], [strength]) [] kennzeichnet optionale Argumente
Argumente	• **attribute** - ein Attributname • **feature** - Ein Objekt. Wenn nicht festgelegt, wird das aktuelle Objekt verwendet.. • **layer** - Ein Vektor Layer. Wenn nicht festgelegt, wird der aktuelle Layer verwendet.. • **strength** - Setzen Sie diesen Wert auf ‚hard' oder ‚soft', um eine bestimmte Art von Bedingung einzugrenzen. Ist dieser Wert nicht gesetzt, gibt die Funktion FALSE zurück, wenn entweder eine harte oder eine weiche Beschränkung fehlschlägt..
Beispiele	• `is_attribute_valid('HECTARES')` → TRUE, wenn der Wert des aktuellen Objekts im Feld „HECTARES" alle Bedingungen erfüllt. • `is_attribute_valid('HOUSES',get_feature('my_layer', 'FID', 10), 'my_layer')` → FALSE, wenn der Wert im Feld „HOUSES" des Objekts mit „FID"=10 in „my_layer" nicht alle Bedingungen erfüllt.

Siehe auch: *Restriktionen*

is_feature_valid

Gibt TRUE zurück, wenn ein Objekt alle Feldbeschränkungen erfüllt.

Syntax	is_feature_valid([feature], [layer], [strength]) [] kennzeichnet optionale Argumente
Argumente	• **feature** - Ein Objekt. Wenn nicht festgelegt, wird das aktuelle Objekt verwendet.. • **layer** - Ein Vektor Layer. Wenn nicht festgelegt, wird der aktuelle Layer verwendet.. • **strength** - Setzen Sie diesen Wert auf ‚hard' oder ‚soft', um eine bestimmte Art von Bedingung einzugrenzen. Ist dieser Wert nicht gesetzt, gibt die Funktion FALSE zurück, wenn entweder eine harte oder eine weiche Beschränkung fehlschlägt..
Beispiele	• `is_feature_valid(strength:='hard')` → TRUE, wenn alle Felder des aktuellen Objekts ihre harten Bedingungen erfüllen. • `is_feature_valid(get_feature('my_layer', 'FID', 10), 'my_layer')` → FALSE, wenn alle Felder des Objekts mit „FID"=10 in „my_layer" nicht alle Bedingungen erfüllen.

Siehe auch: *Restriktionen*

is_selected

Gibt TRUE zurück, wenn ein Objekt ausgewählt ist. Kann mit null, einem oder zwei Argumenten verwendet werden, siehe unten für Details.

Keine Parameter

Wird die Funktion ohne Parameter aufgerufen, gibt sie TRUE zurück, wenn das aktuelle Objekt im aktuellen Layer ausgewählt ist.

Syntax	is_selected()
Beispiele	• `is_selected()` → TRUE, wenn das aktuelle Objekt im aktuellen Layer ausgewählt ist.

Ein „Objekt“-Parameter

Wird die Funktion nur mit dem Parameter ‚feature‘ aufgerufen, gibt sie TRUE zurück, wenn das angegebene Objekt aus dem aktuellen Layer ausgewählt ist.

Syntax	is_selected(feature)
Argumente	• **feature** - das Objekt, das für die Auswahl geprüft werden soll.
Beispiele	• `is_selected(@atlas_feature)` → TRUE, wenn das aktuelle Objekt im Atlas ausgewählt ist. • `is_selected(get_feature('streets', 'name', 'Main St.'))` → TRUE, wenn das eindeutig benannte Objekt „Main St.“ auf dem aktiven Layer „streets“ ausgewählt ist. • `is_selected(get_feature_by_id('streets', 1))` → TRUE, wenn das Objekt mit der ID 1 auf dem aktiven Layer „streets“ ausgewählt ist.

Zwei Parameter

Wenn die Funktion sowohl mit einem Layer als auch mit einem Objekt aufgerufen wird, gibt sie TRUE zurück, wenn das angegebene Objekt aus dem angegebenen Layer ausgewählt ist.

Syntax	is_selected(layer, feature)
Argumente	• **layer** - Der Layer (seine ID oder sein Name), für den die Auswahl geprüft werden soll. • **feature** - das Objekt, das für die Auswahl geprüft werden soll.
Beispiele	• `is_selected( 'streets', get_feature('streets', 'name', "street_name"))` → TRUE, wenn die Straße des aktuellen Gebäudes ausgewählt ist (unter der Annahme, dass der Gebäude-Layer ein Feld namens „street_name“ und der Layer „streets“ ein Feld namens „name“ mit eindeutigen Werten hat). • `is_selected( 'streets', get_feature_by_id('streets', 1))` → TRUE, wenn das Objekt mit der ID 1 aus dem Layer „streets“ ausgewählt ist.

maptip

Gibt den Maptip für ein bestimmtes Objekt in einem Layer zurück. Der Ausdruck wird standardmäßig ausgewertet. Kann mit null, einem oder mehreren Argumenten verwendet werden, siehe unten für Details.

Keine Parameter

Wenn die Funktion ohne Parameter aufgerufen wird, wertet sie den Maptip des aktuellen Objekts im aktuellen Layer aus.

Syntax	maptip()
Beispiele	• `maptip()` → Der Maptip des aktuellen Objekts im aktuellen Layer.

Ein „Objekt“-Parameter

Wenn die Funktion nur mit dem Parameter ‚feature‘ aufgerufen wird, wertet sie das angegebene Objekt aus dem aktuellen Layer aus..

Syntax	maptip(feature)
Argumente	• **feature** - das Objekt, das ausgewertet werden soll.
Beispiele	• `maptip(@atlas_feature)` → Der Maptip des aktuellen Objekts im Atlas.

Layer- und Objekt-Parameter

Wenn die Funktion mit einem Layer und einem Objekt aufgerufen wird, wird das angegebene Objekt aus dem angegebenen Layer ausgewertet.

Syntax	maptip(layer, feature, [evaluate=true]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - Der Layer (oder seine ID oder sein Name) • **feature** - das Objekt, das ausgewertet werden soll. • **evaluate** - Wenn der Ausdruck ausgewertet werden muss. Bei false wird der Ausdruck nur als String-Literal zurückgegeben (das möglicherweise später mit der Funktion „eval_template“ ausgewertet werden kann).
Beispiele	• `maptip('streets', get_feature_by_id('streets', 1))` → Der Maptip des Objekts mit der ID 1 aus dem Layer ‚streets“. • `maptip('a_layer_id', @feature, 'False')` → Der Maptip des angegebenen Objekts wurde nicht ausgewertet.

num_selected

Gibt die Anzahl der ausgewählten Objekte auf einem bestimmten Layer zurück. Funktioniert standardmäßig auf dem Layer, auf dem der Ausdruck ausgewertet wird.

Syntax	num_selected([layer=current layer]) [] kennzeichnet optionale Argumente
Argumente	• **layer** - Der Layer (bzw. seine ID oder sein Name), für den die Auswahl überprüft werden soll.
Beispiele	• `num_selected()` → Die Anzahl der ausgewählten Objekte im aktuellen Layer. • `num_selected('streets')` → Die Anzahl der ausgewählten Objekte im Layer „streets“

represent_attributes

Gibt eine Zuordnung mit den Attributnamen als Schlüssel und den konfigurierten Darstellungswerten als Werte zurück. Der Darstellungswert für die Attribute hängt von dem konfigurierten Widget-Typ für jedes Attribut ab. Kann mit null, einem oder mehreren Argumenten verwendet werden, siehe unten für Details.

Keine Parameter

Wird die Funktion ohne Parameter aufgerufen, liefert sie die Darstellung der Attribute des aktuellen Objekts im aktuellen Layer.

Syntax	represent_attributes()
Beispiele	• `represent_attributes()` → Die Darstellung der Attribute für das aktuelle Objekt.

Ein „Objekt"-Parameter

Wenn die Funktion nur mit dem Parameter ‚feature' aufgerufen wird, liefert sie die Darstellung der Attribute des angegebenen Objekts aus dem aktuellen Layer.

Syntax	represent_attributes(feature)
Argumente	• **feature** - das Objekt, das ausgewertet werden soll.
Beispiele	• `represent_attributes(@atlas_feature)` → Die Darstellung der Attribute für das angegebene Objekt aus dem aktuellen Layer.

Layer- und Objekt-Parameter

Wird die Funktion mit einem ‚layer' und einem ‚feature'-Parameter aufgerufen, gibt sie die Darstellung der Attribute des angegebenen Objekts aus dem angegebenen Layer zurück.

Syntax	represent_attributes(layer, feature)
Argumente	• **layer** - Der Layer (oder seine ID oder sein Name). • **feature** - das Objekt, das ausgewertet werden soll.
Beispiele	• `represent_attributes('atlas_layer', @atlas_feature)` → die Darstellung der Attribute für das angegebene Objekt aus dem angegebenen Layer

Siehe auch: *represent_value*

represent_value

Gibt den konfigurierten Darstellungswert für einen Feldwert zurück. Er hängt vom konfigurierten Widgettyp ab. Dies ist oft nützlich für „Value Map"-Widgets.

Syntax	represent_value(value, [fieldName]) [] kennzeichnet optionale Argumente
Argumente	• **value** - Der Wert, der aufgelöst werden soll. Höchstwahrscheinlich ein Feld. • **fieldName** - Der Feldname, für den die Widget-Konfiguration geladen werden soll.
Beispiele	• `represent_value("field_with_value_map")` → Beschreibung für den Wert • `represent_value('static value', 'field_name')` → Beschreibung für den statischen Wert

Siehe auch: *widget types*, *represent_attributes*

sqlite_fetch_and_increment

Verwalten von autoinkrementierenden Werten in Sqlite-Datenbanken

SQlite-Standardwerte können nur beim Einfügen angewendet und nicht vorab abgerufen werden.

Dies macht es unmöglich, einen inkrementierten Primärschlüssel über AUTO_INCREMENT zu erhalten, bevor die Zeile in der Datenbank erstellt wird. Anmerkung: Bei Postgres funktioniert dies über die Option *evaluate default values*.

Wenn man neue Objekte mit Beziehungen hinzufügt, ist es sehr praktisch, wenn man bereits Kinder für ein übergeordnetes Objekt hinzufügen kann, während das Formular der Eltern noch offen ist und das übergeordnete Objekt somit noch nicht gebunden ist.

Um diese Einschränkung zu umgehen, kann diese Funktion verwendet werden, um Sequenzwerte in einer separaten Tabelle auf Sqlite-basierten Formaten wie gpkg zu verwalten.

Die Sequenztabelle wird nach einer Sequenz-ID gefiltert (filter_attribute und filter_value), und der aktuelle Wert des id_field wird um 1 erhöht und der erhöhte Wert zurückgegeben.

Wenn für zusätzliche Spalten Werte angegeben werden müssen, kann die Zuordnung default_values für diesen Zweck verwendet werden.

Hinweis

Diese Funktion modifiziert die Ziel-Sqlite-Tabelle. Sie ist für die Verwendung mit Standardwertkonfigurationen für Attribute vorgesehen.

Wenn der Datenbankparameter ein Layer ist und der Layer sich im Transaktionsmodus befindet, wird der Wert nur einmal während der Lebensdauer einer Transaktion abgerufen, zwischengespeichert und inkrementiert. Dies macht es unsicher, von mehreren Prozessen aus parallel an der gleichen Datenbank zu arbeiten.

Syntax	sqlite_fetch_and_increment(database, table, id_field, filter_attribute, filter_value, [default_values]) [] kennzeichnet optionale Argumente
Argumente	• **database** - Pfad zur Sqlite-Datei oder zum GeoPackage Layer • **table** - Name der Tabelle, in der die Sequenzen verwaltet werden • **id_field** - Name des Feldes, das den aktuellen Wert enthält • **filter_attribute** - Name des Feldes, das einen eindeutigen Bezeichner für diese Sequenz enthält. Muss einen UNIQUE-Index haben. • **filter_value** - Name der zu verwendenden Sequenz. • **default_values** - Zuordnung mit Standardwerten für zusätzliche Spalten in der Tabelle. Die Werte müssen vollständig in Anführungszeichen gesetzt werden. Funktionen sind erlaubt.
Beispiele	• `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

Siehe auch: *Datenquellen*, *Festlegung von Beziehungen zwischen mehreren Layern*

uuid

Erzeugt einen Universally Unique Identifier (UUID) für jede Zeile mit der Qt-Methode QUuid::createUuid.

Syntax	uuid([format='WithBraces']) [] kennzeichnet optionale Argumente
Argumente	• **format** - Das Format, in dem die UUID formatiert werden soll. ‚WithBraces', ‚WithoutBraces' oder ‚Id128'.
Beispiele	• `uuid()` → ‚{0bd2f60f-f157-4a6d-96af-d4ba4cb366a1}' • `uuid('WithoutBraces')` → ‚0bd2f60f-f157-4a6d-96af-d4ba4cb366a1' • `uuid('Id128')` → ‚0bd2f60ff1574a6d96afd4ba4cb366a1'

9.2.24 Beziehungen

Diese Gruppe enthält die Liste der im aktuellen Projekt verfügbaren Beziehungen) oder ein Formular anzupassen.

9.2.25 Sensoren-Funktionen

Diese Gruppe enthält Funktionen zur Interaktion mit Sensoren.

sensor_data

Gibt den letzten erfassten Wert (oder Werte als Zuordnung zu Sensoren, die mehrere Werte melden) von einem registrierten Sensor zurück.

Syntax	sensor_data(name, [expiration]) [] kennzeichnet optionale Argumente
Argumente	• **name** - den Sensornamen • **expiration** - maximal zulässige Millisekunde seit dem letzten erfassten Wert
Beispiele	• `sensor_data('geiger_1')` → ‚2000‘

9.2.26 Zeichenketten-Funktionen

Diese Gruppe enthält Funktionen für Zeichenketten (z.B. Ersetzen, in Großbuchstaben umwandeln).

ascii

Gibt den Unicode-Code zurück, der dem ersten Zeichen einer Zeichenkette zugeordnet ist.

Syntax	ascii(string)
Argumente	• **string** - die in Unicode-Code zu konvertierende Zeichenkette
Beispiele	• `ascii('Q')` → 81

char

Gibt das mit einem Unicode-Code verbundene Zeichen zurück.

Syntax	char(code)
Argumente	• **code** - eine Unicode-Kennzahl
Beispiele	• `char(81)` → ‚Q‘

concat

Verkettet mehrere Zeichenketten zu einer einzigen. NULL-Werte werden in leere Zeichenketten umgewandelt. Andere Werte (wie Zahlen) werden in Zeichenketten umgewandelt

Syntax	concat(string1, string2, …)
Argumente	• **string** - einen String-Wert
Beispiele	• `concat('sun', 'set')` → ‚sunset‘ • `concat('a','b','c','d','e')` → ‚abcde‘ • `concat('Anno ', 1984)` → ‚Anno 1984‘ • `concat('The Wall', NULL)` → ‚The Wall‘

Verkettung von Feldern

Sie können auch Zeichenketten oder Feldwerte verketten, indem Sie die Operatoren `||` oder + verwenden, mit einigen besonderen Eigenschaften:

- Der Operator + bedeutet auch Summenausdruck. Wenn Sie also einen ganzzahligen Operanden (Feld oder numerischer Wert) haben, kann dies fehleranfällig sein und Sie sollten besser die anderen Operatoren verwenden:

```
'My feature id is: ' + "gid" => triggers an error as gid returns an integer
```

- Wenn eines der Argumente einen NULL-Wert hat, gibt entweder „||“ oder „+“ einen NULL-Wert zurück. Um die anderen Argumente unabhängig vom NULL-Wert zurückzugeben, können Sie die Funktionen „concat“ oder „array_to_string“ verwenden:

```
'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: '
```

Siehe auch: *array_to_string*, ||, +

format

Formatieren einer Zeichenkette mit den angegebenen Argumenten.

Syntax	format(string, arg1, arg2, …)
Argumente	• **string** - Eine Zeichenkette mit den Platzhaltern %1, %2, usw. für die Argumente. Platzhalter können wiederholt werden. Der Platzhalter mit der niedrigsten Nummer wird durch arg1 ersetzt, der nächste durch arg2, usw.. • **arg** - beliebiger Typ. Beliebige Anzahl von Argumenten.
Beispiele	• `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

Formatiert einen Datumstyp oder einen String in ein benutzerdefiniertes Stringformat. Verwendet Qt Datum/Zeit Format Strings. Siehe QDateTime::toString.

Syntax	format_date(datetime, format, [language]) [] kennzeichnet optionale Argumente
Argumente	• **datetime** - Datum, Uhrzeit oder DatumZeit-Wert • **format** - Formatvorlage, die zur Formatierung der Zeichenkette verwendet werden soll.

Ausdruck	gibt zurück
d	den Tag als Zahl ohne führende Null (1 bis 31)
dd	den Tag als Zahl mit führender Null (01 bis 31)
ddd	den abgekürzten lokalisierten Tagesnamen (z. B. ‚Mon' bis ‚Sun')
dddd	den langen lokalisierten Tagesnamen (z. B. ‚Monday' bis ‚Sunday')
M	den Monat als Zahl ohne führende Null (1-12)
MM	den Monat als Zahl mit führender Null (01-12)
MMM	den abgekürzten lokalisierten Monatsnamen (z. B. ‚Jan' bis ‚Dec')
MMMM	den langen lokalisierten Monatsnamen (z. B. ‚January' bis ‚December')
yy	das Jahr als zweistellige Zahl (00-99)
yyyy	das Jahr als vierstellige Zahl

Diese Ausdrücke können für den Zeitteil des Formatstrings verwendet werden:

Ausdruck	gibt zurück
h	die Stunde ohne führende Null (0 bis 23 oder 1 bis 12 bei AM/PM-Anzeige)
hh	die Stunde mit führender Null (00 bis 23 oder 01 bis 12 bei AM/PM-Anzeige)
H	die Stunde ohne führende Null (0 bis 23, auch bei AM/PM-Anzeige)
HH	die Stunde mit einer führenden Null (00 bis 23, auch bei AM/PM-Anzeige)
m	die Minute ohne führende Null (0 bis 59)
mm	die Minute mit führender Null (00 bis 59))
s	die Sekunde ohne führende Null (0 bis 59)
ss	die Sekunde mit führender Null (00 bis 59)
z	die Millisekunden ohne nachgestellte Nullen (0 bis 999)
zzz	die Millisekunden mit nachgestellten Nullen (000 bis 999)
AP oder A	als AM/PM-Zeit interpretieren. *AP* muss entweder ‚AM' oder ‚PM' sein.
ap oder a	Interpretation als AM/PM-Zeit. *ap* muss entweder ‚am' oder ‚pm' sein.

- **language** - Sprache (Kleinbuchstaben, zwei- oder dreibuchstabig, ISO 639 Sprachcode), die verwendet wird, um das Datum in eine benutzerdefinierte Zeichenfolge zu formatieren. Standardmäßig wird das aktuelle Gebietsschema des QGIS-Benutzers verwendet.

Beispiele

- `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', wenn das aktuelle Gebietsschema eine englische Variante ist
- `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

Gibt eine Zahl zurück, die mit dem Gebietsschema-Trennzeichen für Tausender formatiert ist. Standardmäßig wird das aktuelle QGIS-Benutzergebietsschema verwendet. Außerdem werden die Dezimalstellen auf die Anzahl der angegebenen Stellen gekürzt.

Syntax	format_number(number, [places=0], [language], [omit_group_separators=false], [trim_trailing_zeroes=false]) [] kennzeichnet optionale Argumente
Argumente	• **number** - Nummer, die formatiert werden soll • **places** - Ganzzahl, die die Anzahl der Dezimalstellen angibt, auf die die Zeichenkette gekürzt werden soll. • **language** - Sprache (Kleinbuchstaben, zwei- oder dreibuchstabig, ISO 639 Sprachcode), die zur Formatierung der Zahl in eine Zeichenkette verwendet wird. Standardmäßig wird das aktuelle Gebietsschema des QGIS-Benutzers verwendet. • **omit_group_separators** - wenn auf true gesetzt, werden Gruppentrennzeichen nicht in die Zeichenkette aufgenommen • **trim_trailing_zeroes** - wenn auf true gesetzt, werden nachstehende Nullen nach dem Dezimalpunkt aus der Zeichenkette entfernt
Beispiele	• `format_number(10000000.332,2)` → ‚10,000,000.33‘ wenn z. B. das aktuelle Gebietsschema eine englischsprachige Variante ist • `format_number(10000000.332,2,'fr')` → ‚10 000 000,33‘

left

Gibt eine Teilzeichenkette zurück, die die *n* am weitesten links stehenden Zeichen der Zeichenkette enthält.

Syntax	left(string, length)
Argumente	• **string** - eine Zeichenkette • **length** - Ganzzahl. Die Anzahl der Zeichen von links in der Zeichenkette, die zurückgegeben werden sollen.
Beispiele	• `left('Hello World',5)` → ‚Hello‘

Siehe auch: *right*

length

Gibt die Anzahl der Zeichen in einer Zeichenkette oder die Länge eines geometrischen Linienstring zurück.

Zeichenketten-Variante

Gibt die Anzahl der Zeichen in einer Zeichenkette zurück.

Syntax	length(string)
Argumente	• **string** - Zeichenkette, von welcher die Länge gezählt werden soll
Beispiele	• `length('hello')` → 5

Geometrie-Variante

Berechnet die Länge eines geometrischen Linienobjekts. Die Berechnungen erfolgen immer planimetrisch im räumlichen Bezugssystem (SRS) dieser Geometrie, und die Einheiten der zurückgegebenen Länge entsprechen den Einheiten für das SRS. Dies unterscheidet sich von den Berechnungen, die von der Funktion $length durchgeführt werden, die ellipsoidische Berechnungen basierend auf den Einstellungen für Ellipsoid und Entfernungseinheiten des Projekts vornimmt.

Syntax	length(geometry)
Argumente	• **geometry** - Liniengeometrie-Objekt
Beispiele	• `length(geom_from_wkt('LINESTRING(0 0, 4 0)'))` → 4.0

lower

Konvertiert eine Zeichenkette in Kleinbuchstaben.

Syntax	lower(string)
Argumente	• **string** - die in Kleinbuchstaben zu konvertierende Zeichenkette
Beispiele	• `lower('HELLO World')` → ‚hello world‘

Siehe auch: *upper*, *title*

lpad

Gibt eine Zeichenkette zurück, die links auf die angegebene Breite aufgefüllt wird, wobei ein Füllzeichen verwendet wird. Wenn die Zielbreite kleiner ist als die Länge der Zeichenkette, wird die Zeichenkette abgeschnitten.

Syntax	lpad(string, width, fill)
Argumente	• **string** - Zeichenkette zum Auffüllen • **width** - Länge der neuen Zeichenkette • **fill** - Zeichen, mit dem der verbleibende Platz ausgefüllt werden soll
Beispiele	• `lpad('Hello', 10, 'x')` → ‚xxxxxHello‘ • `lpad('Hello', 3, 'x')` → ‚Hel‘

Siehe auch: *rpad*

ltrim

Entfernt die längste Zeichenfolge, die nur die angegebenen Zeichen (standardmäßig ein Leerzeichen) enthält, vom Anfang der Zeichenfolge.

Syntax	ltrim(string, [characters=' ']) [] kennzeichnet optionale Argumente
Argumente	• **string** - Zeichenfolge zum Trimmen • **characters** - zu trimmende Zeichen
Beispiele	• `ltrim(' hello world ')` → ‚hello world ' • `ltrim('zzzytest', 'xyz')` → ‚test'

Siehe auch: *rtrim*, *trim*

regexp_match

Gibt die erste übereinstimmende Position eines regulären Ausdrucks innerhalb einer Unicode-Zeichenkette zurück, oder 0, wenn die Teilzeichenkette nicht gefunden wird.

Syntax	regexp_match(input_string, regex)
Argumente	• **input_string** - die mit dem regulären Ausdruck zu prüfende Zeichenfolge • **regex** - Der reguläre Ausdruck, gegen den getestet werden soll. Backslash-Zeichen müssen doppelt escaped werden (z. B. „\\s" für ein Leerzeichen oder „\\b" für eine Wortgrenze).
Beispiele	• `regexp_match('QGIS ROCKS','\\sROCKS')` → 5 • `regexp_match('Budač','udač\\b')` → 2

Siehe auch: *strpos*

regexp_replace

Gibt eine Zeichenkette zurück, die durch den angegebenen regulären Ausdruck ersetzt wurde.

Syntax	regexp_replace(input_string, regex, replacement)
Argumente	• **input_string** - Zeichenfolge, in der die zu ersetzende Muster gesucht werden sollen • **regex** - Der zu ersetzende reguläre Ausdruck. Backslash-Zeichen müssen doppelt maskiert werden (z. B. „\\s", um ein Leerzeichen zu ersetzen). • **replacement** - Die Zeichenkette, die alle übereinstimmenden Vorkommen des übergebenen regulären Ausdrucks ersetzen soll. Eingebettete Gruppen können in die Ersetzungszeichenkette eingefügt werden, indem man \\1, \\2, etc..
Beispiele	• `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'

Siehe auch: *replace*, *array_replace*

regexp_substr

Gibt den Teil einer Zeichenkette zurück, der mit einem angegebenen regulären Ausdruck übereinstimmt.

Syntax	regexp_substr(input_string, regex)
Argumente	• **input_string** - die Zeichenkette, in der nach Übereinstimmungen gesucht werden soll • **regex** - Der reguläre Ausdruck, der abgeglichen werden soll. Backslash-Zeichen müssen mit doppeltem Backslash versehen werden (z. B. „\\s“, um ein Leerzeichen zu finden).
Beispiele	• `regexp_substr('abc123','(\\d+)')` → ‚123‘

Siehe auch: *substr*, *regexp_matches*

repeat

Repeats a string a specified number of times.

Syntax	repeat(string, number)
Argumente	• **string** - string to repeat • **number** - number of times to repeat string
Beispiele	• `repeat('Hello', 3)` → ‚HelloHelloHello‘

replace

Gibt eine Zeichenkette zurück, die durch die übergebene Zeichenkette, das Array oder die Map von Zeichenketten ersetzt wurde.

String- und Array-Variante

Gibt eine Zeichenkette zurück, bei der die übergebene Zeichenkette oder ein Array von Zeichenketten durch eine Zeichenkette oder ein Array von Zeichenketten ersetzt wurde.

Syntax	replace(string, before, after)
Argumente	• **string** - die Eingabezeichenkette • **before** - die zu ersetzende Zeichenkette oder ein Array von Zeichenketten • **after** - die Zeichenkette oder das Array von Zeichenketten, die als Ersatz verwendet werden sollen
Beispiele	• `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-Variante

Gibt eine Zeichenkette zurück, in der die übergebenen Map-Schlüssel durch gepaarte Werte ersetzt sind. Längere Map-Schlüssel werden zuerst ausgewertet.

Syntax	replace(string, map)
Argumente	• **string** - die Eingabezeichenkette • **map** - die Zuordnung von Schlüsseln und Werten
Beispiele	• `replace('APP SHOULD ROCK',map('APP','QGIS','SHOULD', 'DOES'))` → ‚QGIS DOES ROCK' • `replace('forty two',map('for','4','two','2','forty two', '42'))` → ‚42'

Siehe auch: *regexp_replace*, *array_replace*

reverse

Reverses the direction of a line string or reverses a string of text.

Zeichenketten-Variante

Reverses the order of characters in a string.

Syntax	reverse(string)
Argumente	• **string** - string to reverse
Beispiele	• `reverse('hello')` → ‚olleh'

Geometrie-Variante

Kehrt die Richtung einer Linienkette um, indem es die Reihenfolge ihrer Knotenpunkte umkehrt.

Syntax	reverse(geometry)
Argumente	• **geometry** - eine Geometrie
Beispiele	• `geom_to_wkt(reverse(geom_from_wkt('LINESTRING(0 0, 1 1, 2 2)')))` → ‚LINESTRING(2 2, 1 1, 0 0)'

right

Gibt eine Teilzeichenkette zurück, die die *n* äußersten rechten Zeichen der Zeichenkette enthält.

Syntax	right(string, length)
Argumente	• **string** - eine Zeichenkette • **length** - Ganzzahl. Die Anzahl der Zeichen, die von der rechten Seite der Zeichenkette zurückgegeben werden sollen.
Beispiele	• `right('Hello World',5)` → ‚World‘

Siehe auch: *left*

rpad

Gibt eine Zeichenkette zurück, die rechts auf die angegebene Breite aufgefüllt wurde, wobei ein Füllzeichen verwendet wird. Wenn die Zielbreite kleiner ist als die Länge der Zeichenkette, wird die Zeichenkette abgeschnitten.

Syntax	rpad(string, width, fill)
Argumente	• **string** - Zeichenkette zum Auffüllen • **width** - Länge der neuen Zeichenkette • **fill** - Zeichen, mit dem der verbleibende Platz ausgefüllt werden soll
Beispiele	• `rpad('Hello', 10, 'x')` → ‚Helloxxxxx‘ • `rpad('Hello', 3, 'x')` → ‚Hel‘

Siehe auch: *lpad*

rtrim

Entfernt die längste Zeichenfolge, die nur die angegebenen Zeichen (standardmäßig ein Leerzeichen) enthält, vom Ende der Zeichenfolge.

Syntax	rtrim(string, [characters=‘ ,]) [] kennzeichnet optionale Argumente
Argumente	• **string** - Zeichenfolge zum Trimmen • **characters** - zu trimmende Zeichen
Beispiele	• `rtrim(' hello world ')` → , hello world‘ • `rtrim('testxxzx', 'xyz')` → ‚test‘

Siehe auch: *ltrim*, *trim*

strpos

Gibt die erste übereinstimmende Position einer Teilzeichenkette innerhalb einer anderen Zeichenkette zurück, oder 0, wenn die Teilzeichenkette nicht gefunden wird.

Syntax	strpos(haystack, needle)
Argumente	• **haystack** - zu durchsuchende Zeichenkette • **needle** - Zeichenkette, nach der gesucht werden soll
Beispiele	• `strpos('HELLO WORLD','WORLD')` → 7 • `strpos('HELLO WORLD','GOODBYE')` → 0

Siehe auch: *regexp_match*

substr

Gibt einen Teil einer Zeichenkette zurück.

Syntax	substr(string, start, [length]) [] kennzeichnet optionale Argumente
Argumente	• **string** - die vollständige Eingabezeichenkette • **start** - Ganzzahl, die die zu extrahierende Startposition angibt, beginnend mit 1; wenn start negativ ist, beginnt die Rückgabezeichenkette am Ende der Zeichenkette abzüglich des Startwerts • **length** - Ganzzahl, die die Länge der zu extrahierenden Zeichenkette angibt; wenn length negativ ist, wird die Rückgabezeichenkette die angegebene Länge der Zeichen am Ende der Zeichenkette auslassen
Beispiele	• `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‘

Siehe auch: *regexp_substr*, *regexp_matches*

title

Konvertiert alle Wörter einer Zeichenkette in Groß- und Kleinschreibung (alle Wörter klein mit führendem Großbuchstaben).

Syntax	title(string)
Argumente	• **string** - die Zeichenkette, die in Groß- und Kleinschreibung umgewandelt werden soll
Beispiele	• `title('hello WOrld')` → ‚Hello World‘

Siehe auch: *lower*, *upper*

to_string

Converts a number to string. The conversion is not locale-aware, see ‚format_number' for a locale-aware alternative.

Syntax	to_string(number)
Argumente	• **number** - Ganzzahl oder reeller Wert, der in eine Zeichenkette konvertiert werden soll
Beispiele	• `to_string(1.23)` → ‚1.23'

Siehe auch: *format_number*

trim

Entfernt alle führenden und nachfolgenden Whitespaces (Leerzeichen, Reiter, etc.) aus einer Zeichenkette.

Syntax	trim(string)
Argumente	• **string** - Zeichenfolge zum Trimmen
Beispiele	• `trim(' hello world ')` → ‚hello world'

Siehe auch: *ltrim*, *rtrim*

upper

Konvertiert eine Zeichenkette in Großbuchstaben.

Syntax	upper(string)
Argumente	• **string** - die Zeichenkette, die in Großbuchstaben umgewandelt werden soll
Beispiele	• `upper('hello WOrld')` → ‚HELLO WORLD'

Siehe auch: *lower*, *title*

wordwrap

Gibt eine Zeichenkette zurück, die nach einer maximalen/minimalen Anzahl von Zeichen mit einem Absatz-Tag umgebrochen ist.

Syntax	wordwrap(string, wrap_length, [delimiter_string]) [] kennzeichnet optionale Argumente
Argumente	• **string** - die Zeichenkette, die umgebrochen werden soll • **wrap_length** - eine ganze Zahl. Wenn wrap_length positiv ist, stellt die Zahl die ideale maximale Anzahl der umzubrechenden Zeichen dar; wenn sie negativ ist, stellt die Zahl die minimale Anzahl der umzubrechenden Zeichen dar. • **delimiter_string** - Optionale Begrenzungszeichenfolge zum Umbruch in eine neue Zeile.
Beispiele	• `wordwrap('UNIVERSITY OF QGIS',13)` → ‚UNIVERSITY OF QGIS‘ • `wordwrap('UNIVERSITY OF QGIS',-3)` → ‚UNIVERSITY OF QGIS‘

9.2.27 Benutzerdefinierte Ausdrücke

Diese Gruppe enthält die als *Benutzerausdrücke* gespeicherten Ausdrücke.

9.2.28 Variablen

Diese Gruppe enthält dynamische Variablen im Zusammenhang mit der Anwendung, der Projektdatei und anderen Einstellungen. Die Verfügbarkeit der Variablen hängt vom Kontext ab:

- dem Objekte über Ausdruck wählen Dialog
- dem Feldrechner Dialog
- dem Layereigenschaften Dialog
- dem Drucklayout

Um diese Variablen in einem Ausdruck zu verwenden, sollte ihnen das Zeichen `@` vorangestellt werden (z.B. `@row_number`).

Variable	Beschreibung
algorithm_id	Die eindeutige ID eines Algorithmus
animation_end_time	Ende des gesamten zeitlichen Bereichs der Animation (als Datetime-Wert)
animation_interval	Dauer des gesamten zeitlichen Bereichs der Animation (als Intervallwert)
animation_start_time	Beginn des gesamten zeitlichen Bereichs der Animation (als Datumswert)
atlas_feature	Das aktuelle Atlas-Objekt (als Feature-Objekt)
atlas_featureid	Die ID des aktuellen Atlas-Objekts
atlas_featurenumber	Die Nummer des aktuellen Atlas-Objekts im Layout
atlas_filename	Der Name der aktuellen Atlas-Datei
atlas_geometry	Die Geometrie des aktuellen Atlas-Objektes
atlas_layerid	Die ID des aktuellen Atlas-Coverage-Layers
atlas_layername	Der Name des aktuellen Atlas-Coverage-Layers
atlas_pagename	Der Name der aktuellen Atlas-Seite
atlas_totalfeatures	Die Gesamtzahl der Objekte im Atlas
band	Die Nummer des Bandes im Rasterlayer
band_description	Die Beschreibung des Bandes im Rasterlayer
band_name	Der Name des Bandes im Rasterlayer
canvas_cursor_point	Die letzte Position des Cursors auf der Kartenansicht in geografischen Koordinaten des Projekts
cluster_color	Die Farbe der Symbole innerhalb eines Clusters, oder NULL, wenn die Symbole gemischte Farben haben

Fortsetzung auf der nächsten Seite

Tab. 9.1 – Fortsetzung der vorherigen Seite

Variable	Beschreibung
cluster_size	Die Anzahl der Symbole innerhalb eines Clusters
current_feature	Das Objekt, das gerade im Attributformular oder in der Tabellenzeile bearbeitet wird
current_geometry	Die Geometrie des Objekts, das gerade im Formular oder in der Tabellenzeile bearbeitet wird
current_parent_feature	steht für das Objekt, das gerade im übergeordneten Formular bearbeitet wird. Nur im Kontext eines eingebetteten Formulars verwendbar.
current_parent_geometry	stellt die Geometrie des Objekts dar, das gerade im übergeordneten Formular bearbeitet wird. Nur im Kontext eines eingebetteten Formulars verwendbar.
form_mode	Wofür das Formular verwendet wird, wie AddFeatureMode, SingleEditMode, MultiEditMode, SearchMode, AggregateSearchMode oder IdentifyMode als String.
feature	Das aktuelle Objekt, das ausgewertet wird. Dies kann mit der Funktion „attribute“ verwendet werden, um Attributwerte des aktuellen Objekts auszuwerten.
frame_duration	Zeitliche Dauer der einzelnen Animationsbilder (als Intervallwert)
frame_number	Aktuelle Bildnummer während der Wiedergabe der Animation
frame_rate	Anzahl der Bilder pro Sekunde bei der Wiedergabe der Animation
fullextent_maxx	Maximaler x-Wert der gesamten Kartenansicht (einschließlich aller Layer)
fullextent_maxy	Maximaler y-Wert der gesamten Kartenansicht (einschließlich aller Layer)
fullextent_minx	Minimaler x-Wert aus der gesamten Kartenansicht (einschließlich aller Layer)
fullextent_miny	Minimaler y-Wert der gesamten Kartenansicht (einschließlich aller Layer)
geometry	Die Geometrie des aktuell bewerteten Objekts
geometry_part_count	Die Anzahl der Teile in der Geometrie des gerenderten Objekts
geometry_part_num	Die aktuelle Geometrie-Teilenummer für das darzustellende Objekt
geometry_point_count	Die Anzahl der Punkte im gerenderten Teil der Geometrie
geometry_point_num	Die aktuelle Punktnummer im Teil der gerenderten Geometrie
geometry_ring_num	Aktuelle Geometrieringnummer für das zu rendernde Objekt (nur für Polygonfeatures). Der äußere Ring hat einen Wert von 0.
grid_axis	Die aktuelle Achse der Gitterbeschriftung (z. B. ‚x‘ für Längengrad, ‚y‘ für Breitengrad)
grid_number	Der aktuelle Wert der Gitter-Anmerkung
id	Die ID des aktuell betrachteten Objekts
item_id	Die Benutzer-ID des Layoutelements (nicht unbedingt eindeutig)
item_uuid	Die eindeutige ID des Layoutelements
layer	Der aktuelle Layer
layer_crs	Die Koordinatenbezugssystem-ID des aktuellen Layers
layer_crs_ellipsoid	Die Ellipsoid ID des aktuellen Layers KBS
layer_cursor_point	Punktgeometrie unter der Mausposition im Kartenbereich (oder die GetFeatureInfo-Position im Kontext von QGIS Server) im KBS des aktiven Layers
layer_id	Die ID des aktuellen Layers
layer_ids	Die IDs aller Karten-Layer im aktuellen Projekt als Liste
layer_name	Der Name des aktuellen Layers
layer_vertical_crs	Die Kennung für das vertikale Koordinatenreferenzsystem des Layers (z. B. „EPSG:5703“)
layer_vertical_crs_definition	Die vollständige Definition des vertikalen Koordinatenreferenzsystems des Layers
layer_vertical_crs_description	Der Name des vertikalen Koordinatenreferenzsystems des Layers
layer_vertical_crs_wkt	Die WKT-Definition des vertikalen Koordinatenreferenzsystems des aktuellen Layers
layers	Alle Karten-Layer des aktuellen Projekts in einer Liste
layout_dpi	Die Auflösung der Layout-Darstellung (DPI)
layout_name	Der Name des Layouts
layout_numpages	Die Anzahl der Seiten im Layout

Fortsetzung auf der nächsten Seite

Tab. 9.1 – Fortsetzung der vorherigen Seite

Variable	Beschreibung
layout_page	Die Seitenzahl des aktuellen Elements im Layout
layout_pageheight	Die aktive Seitenhöhe im Layout (in mm für Standardpapierformate oder in der Einheit, die für benutzerdefinierte Papierformate verwendet wurde)
layout_pageoffsets	Array der Y-Koordinaten des oberen Endes jeder Seite. Ermöglicht die dynamische Positionierung von Elementen auf Seiten in einem Kontext, in dem sich die Seitengrößen ändern können
layout_pagewidth	Die aktive Seitenbreite im Layout (in mm für Standardpapierformate oder in der Einheit, die für benutzerdefinierte Papierformate verwendet wurde)
legend_column_count	Die Anzahl der Spalten in der Legende
legend_filter_by_map	Gibt an, ob der Inhalt der Legende durch die Karte gefiltert wird
legend_filter_out_atlas	Gibt an, ob der Atlas aus der Legende herausgefiltert wird
legend_split_layers	Gibt an, ob Layer in der Legende geteilt werden können
legend_title	Der Titel der Legende
legend_wrap_string	Das/die Zeichen, mit denen der Legendentext umbrochen wird
map_crs	Das Koordinatenbezugssystem der aktuellen Karte
map_crs_acronym	Das Akronym des Koordinatenbezugssystems der aktuellen Karte
map_crs_definition	Die vollständige Definition des Koordinatenbezugssystems der aktuellen Karte
map_crs_description	Der Name des Koordinatenbezugssystems der aktuellen Karte
map_crs_ellipsoid	Das Akronym des Ellipsoids des Koordinatenbezugssystems der aktuellen Karte
map_crs_proj4	Die Proj4-Definition des Koordinatenbezugssystems der aktuellen Karte
map_crs_projection	Der beschreibende Name der vom Koordinatenbezugssystem der Karte verwendeten Projektionsmethode (z. B. „Albers Equal Area“)
map_crs_wkt	Die WKT-Definition des Koordinatenbezugssystems der aktuellen Karte
map_end_time	Das Ende des zeitlichen Bereichs der Karte (als DatumZeitwert)
map_extent	Die Geometrie, die die aktuelle Ausdehnung der Karte darstellt
map_extent_center	Das Punkt-Objekt in der Mitte der Karte
map_extent_height	Die aktuelle Höhe der Karte
map_extent_width	Die aktuelle Breite der Karte
map_id	Die ID des aktuellen Kartenziels. Bei Kartenansichten ist dies „canvas“, bei Layout-Kartenansichten die Element-ID.
map_interval	Die Dauer des zeitlichen Bereichs der Karte (als Intervallwert)
map_layer_ids	Die Liste der IDs der in der Karte sichtbaren Layer
map_layers	Die Liste der in der Karte sichtbaren Layer
map_rotation	Die aktuelle Drehung der Karte
map_scale	Der aktuelle Maßstab der Karte
map_start_time	Der Beginn des zeitlichen Bereichs der Karte (als DatumZeitwert)
map_units	Die Maßeinheiten der Karte
map_z_range_lower	Untere Höhe des Höhenbereichs der Karte
map_z_range_upper	Obere Höhe des Höhenbereichs der Karte
model_path	Vollständiger Pfad (einschließlich Dateiname) des aktuellen Modells (oder Projektpfad, wenn das Modell in ein Projekt eingebettet ist)
model_folder	Ordner mit dem aktuellen Modell (oder Projektordner, wenn das Modell in ein Projekt eingebettet ist)
model_name	Name des aktuellen Modells
model_group	Gruppe für aktuelles Modell
notification_message	Inhalt der vom Anbieter gesendeten Benachrichtigung (nur für Aktionen verfügbar, die durch Anbieterbenachrichtigungen ausgelöst werden)
parent	Verweist auf das aktuelle Objekt im übergeordneten Layer und ermöglicht den Zugriff auf dessen Attribute und Geometrie beim Filtern einer *aggregate*-Funktion
plot_axis	Die zugehörige Plotachse, z. B. „x“ oder „y“.
plot_axis_value	Der aktuelle Wert für die Plotachse.
project_abstract	Die Projektzusammenfassung, entnommen aus den Projekt-Metadaten

Fortsetzung auf der nächsten Seite

Tab. 9.1 – Fortsetzung der vorherigen Seite

Variable	Beschreibung
project_area_units	Die Flächeneinheit für das aktuelle Projekt, die bei der Berechnung der Flächen von Geometrien verwendet wird
project_author	Der Projektautor, entnommen aus den Projekt-Metadaten
project_basename	Der Basisname des Dateinamens des aktuellen Projekts (ohne Pfad und Erweiterung)
project_creation_date	Das Erstellungsdatum des Projekts, das den Projekt-Metadaten entnommen wird
project_crs	Kennung für das Koordinatenreferenzsystem des Projekts (z. B. „EPSG:4326“)
project_crs_arconym	Das Akronym des Koordinatenbezugssystems des Projekts
project_crs_definition	Die vollständige Definition des Koordinatenbezugssystems des Projekts
project_crs_description	Die Beschreibung des Koordinatenbezugssystems des Projekts
project_crs_ellipsoid	Das Ellipsoid des Koordinatenbezugssystems des Projekts
project_crs_proj4	Die Proj4-Darstellung des Koordinatenbezugssystems des Projekts
project_crs_wkt	Die WKT-Darstellung des Koordinatenbezugssystems des Projekts
project_distance_units	Die Entfernungseinheit für das aktuelle Projekt, die bei der Berechnung der Länge von Geometrien und Entfernungen verwendet wird
project_ellipsoid	Der Name des Ellipsoids des aktuellen Projekts, der bei der Berechnung von geodätischen Flächen oder Längen von Geometrien verwendet wird
project_filename	Der Dateiname des aktuellen Projekts
project_folder	Der Ordner des aktuellen Projekts
project_home	Der Home-Pfad des aktuellen Projekts
project_identifier	Die Projektkennung, die aus den Metadaten des Projekts stammt
project_keywords	Die Schlüsselwörter des Projekts, die den Metadaten des Projekts entnommen wurden
project_last_saved	Datum/Uhrzeit, zu der das Projekt zuletzt gespeichert wurde.
project_path	Der vollständige Pfad (einschließlich Dateiname) des aktuellen Projekts
project_title	Der Titel des aktuellen Projekts
project_units	Die Einheiten des KBS des Projekts
project_vertical_crs	Kennung für das vertikale Koordinatenreferenzsystem des Projekts (z. B. „EPSG:5703“)
project_vertical_crs_definition	Die vollständige Definition des vertikalen Koordinatenreferenzsystems des Projekts
project_vertical_crs_description	Die Beschreibung des vertikalen Koordinatenreferenzsystems des Projekts
project_vertical_crs_wkt	Die WKT-Darstellung (Well Known Text) des vertikalen Koordinatenreferenzsystems des Projekts
qgis_locale	Die aktuelle Sprache von QGIS
qgis_os_name	Der Name des aktuellen Betriebssystems, z. B. „windows“, „linux“ oder „osx“
qgis_platform	Die QGIS-Plattform, z.B. ‚Desktop‘ oder ‚Server‘
qgis_release_name	Der Name der aktuellen QGIS-Version
qgis_short_version	Die aktuelle QGIS-Version als Shortstring
qgis_version	Die aktuelle QGIS Versionsbezeichnung
qgis_version_no	Die aktuelle QGIS-Versionsnummer
row_number	Speichert die Zahl der aktuellen Reihe
snapping_results	Ermöglicht den Zugriff auf die Ergebnisse des Einrastens während der Digitalisierung eines Objekts (nur in der Funktion „Hinzufügen“ verfügbar)
scale_value	Der aktuelle Wert für den Maßstabsleistenabstand
selected_file_path	Ausgewählter Dateipfad vom Datei-Widget-Selektor beim Hochladen einer Datei mit einem externen Speichersystem
symbol_angle	Der Winkel des Symbols, das zur Darstellung des Objekts verwendet wird (gilt nur für Markierungssymbole)
symbol_color	Die Farbe des Symbols, das zur Darstellung des Objekts verwendet wird
symbol_count	Die Anzahl der Objekte, die durch das Symbol dargestellt werden (in der Layout-Legende)
symbol_frame	Die Bildnummer (nur bei animierten Symbolen)
symbol_id	Die interne ID des Symbols (in der Layoutlegende)

Fortsetzung auf der nächsten Seite

Tab. 9.1 – Fortsetzung der vorherigen Seite

Variable	Beschreibung
symbol_label	Die Beschriftung für das Symbol (entweder eine benutzerdefinierte Beschriftung oder die automatisch generierte Standardbeschriftung) in der Layoutlegende
symbol_layer_count	Gesamtzahl der Layer im Symbol
symbol_layer_index	Aktueller Layer-Index des Symbols
symbol_marker_column	Spaltennummer für die Markierung (gilt nur für Punktmusterfüllungen)
symbol_marker_row	Zeilennummer für die Markierung (gilt nur für Punktmusterfüllungen)
user_account_name	Der Name des Betriebssystemkontos des aktuellen Benutzers
user_full_name	Der Benutzername des aktuellen Benutzers für das Betriebssystem
value	Der aktuelle Wert
vector_tile_zoom	Genaue Vektor-Kachel-Zoomstufe der Karte, die gerendert wird (abgeleitet vom aktuellen Kartenmaßstab). Normalerweise im Intervall [0, 20]. Im Gegensatz zu @zoom_level ist diese Variable ein Fließkommawert, der zur Interpolation von Werten zwischen zwei ganzzahligen Zoomstufen verwendet werden kann.
with_variable	Ermöglicht das Setzen einer Variablen zur Verwendung innerhalb eines Ausdrucks und vermeidet die wiederholte Berechnung desselben Werts
zoom_level	Vektor-Tile Zoomlevel der Karte, die gerendert wird (abgeleitet vom aktuellen Kartenmaßstab). Normalerweise im Intervall [0, 20].

Beispiele:

- Liefert die X-Koordinate des Mittelpunkts eines Kartenelements im Layout:

```
x( map_get( item_variables( 'map1'), 'map_extent_center' ) )
```

- Gibt für jedes Objekt im aktuellen Layer die Anzahl der sich überschneidenden Flughafenobjekte zurück:

```
aggregate( layer:='airport', aggregate:='count', expression:="code",
               filter:=intersects( $geometry, geometry( @parent ) ) )
```

- Ermittelt die object_id des ersten erfassten Punkts einer Linie:

```
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 Kürzlich verwendete Funktionen

Diese Gruppe enthält die zuletzt verwendeten Funktionen. Je nach Verwendungskontext (Objektauswahl, Feldrechner, allgemein) werden die zuletzt verwendeten Ausdrücke der entsprechenden Liste hinzugefügt (bis zu zehn Ausdrücke), sortiert von mehr bis weniger aktuell. Dies erleichtert das schnelle Auffinden und erneute Anwenden von zuvor verwendeten Ausdrücken.

KAPITEL 10

Die Stilbibliothek

10.1 Die Stilverwaltung

10.1.1 Der Stilverwaltung-Dialog

Die *Stilverwaltung* ist der Ort, an dem Sie grundlegende Stilelemente verwalten und erstellen können. Dies sind Symbole, Farbrampen, Textformate oder Etiketteneinstellungen, die zur Symbolisierung von Objekten, Layern oder Drucklayouts verwendet werden können. Sie werden in der Datenbank `symbology-style.db` unter dem aktiven *Benutzerprofil* gespeichert und für alle mit diesem Profil geöffneten Projektdateien freigegeben. Stilelemente können dank der Export-/Importfunktionen des Dialogs *Stilverwaltung* auch mit anderen gemeinsam genutzt werden.

Sie können diesen Dialog öffnen über:

- das Menü *Einstellungen* ► *Stilverwaltung…*
- mit der Schaltfläche Stilverwaltung in der Werkzeugleiste des Projekts
- mit der Schaltfläche Stilverwaltung aus dem Menü eines Vektor-Layers *Layereigenschaften* ► *Konfiguration eines Symbols* oder *Formatierung eines Textes*).

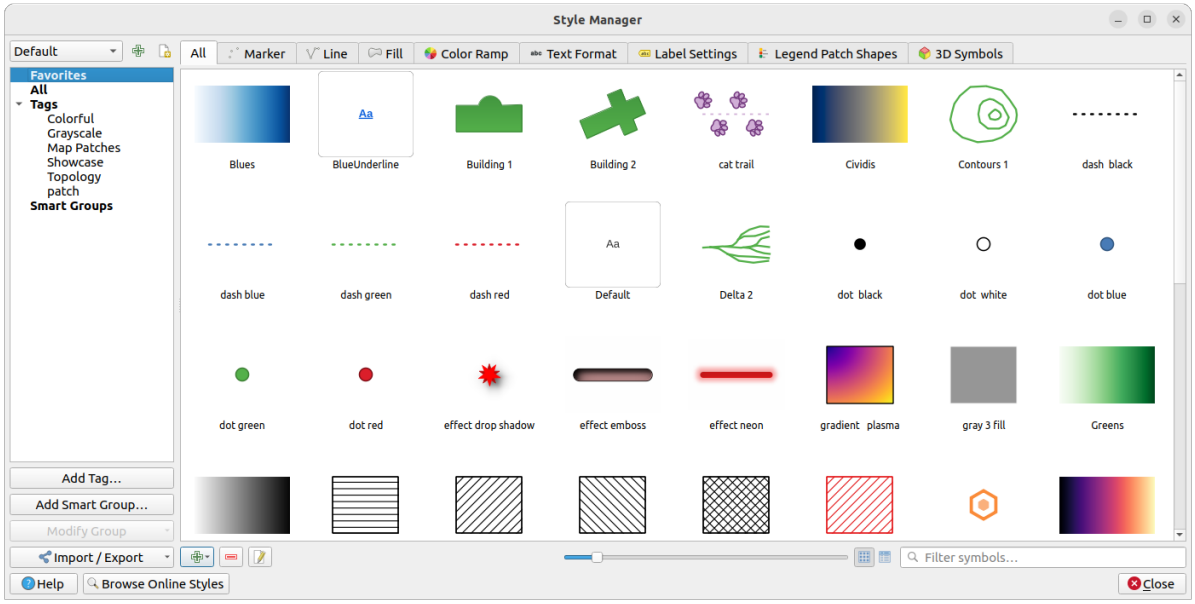


Abb. 10.1: Die Stilverwaltung

Organisieren von Stil-Elementen

In der oberen linken Ecke des Dialogs *Stilverwaltung* können Sie im Dropdown-Menü die Stil-Datenbank auswählen, mit der Sie sich verbinden möchten. Wenn Sie *Standard* wählen, werden Sie mit der Standard-Stil-Datenbank verbunden, in der Sie alle verfügbaren Standard-Stil-Elemente und die von Ihnen in dieser Datenbank gespeicherten finden können. Wenn Sie *Projektstile* wählen, werden Sie mit der Projektstil-Datenbank verbunden, in der Sie nur die Symbole finden können, die Sie in dieser Datenbank gespeichert haben. Wenn Sie weitere Stil-Datenbanken erstellt haben, werden diese im Dropdown-Menü aufgelistet. Es gibt auch die Optionen Vorhandene Stildatenbank zum Projekt hinzufügen und Neue Stildatenbank erstellen (siehe mehr *Stildatenbank*).

Für jede Stildatenbank können Sie die Elemente in verschiedene Kategorien einteilen, die im Bedienfeld auf der linken Seite aufgeführt sind:

- **Favoriten**: wird standardmäßig angezeigt, wenn ein Element konfiguriert wird; zeigt eine erweiterbare Gruppe von Elementen an;
- **Alle**: listet alle verfügbaren Elemente für den aktiven Typ auf;
- **Tags**: zeigt eine Liste von Tags an, die Sie zur Identifizierung der Elemente verwenden können. Ein Element kann mehr als einmal gekennzeichnet werden. Wählen Sie ein Tag in der Liste aus, und die Reiter werden aktualisiert, um nur die zugehörigen Elemente anzuzeigen. Um einen neuen Tag zu erstellen, den Sie später an eine Reihe von Elementen anhängen können, verwenden Sie die Schaltfläche *Tag hinzufügen…* oder wählen Sie den Befehl *Tag hinzufügen…* aus einem beliebigen Tag-Kontextmenü;
- **Schlaue Gruppe**: eine schlaue Gruppe holt sich ihre Symbole dynamisch gemäß den festgelegten Bedingungen (siehe z.B. Abb. 10.2). Klicken Sie auf die Schaltfläche *Schlau-Gruppe hinzufügen…*, um intelligente Gruppen zu erstellen. Im Dialogfeld können Sie einen Ausdruck eingeben, um die auszuwählenden Elemente zu filtern (hat ein bestimmtes Tag, hat eine Zeichenkette im Namen, usw.). Jedes Symbol, jede Farbrampe, jedes Textformat und jede Etiketteneinstellung, die die eingegebene(n) Bedingung(en) erfüllen, werden automatisch der intelligenten Gruppe hinzugefügt.

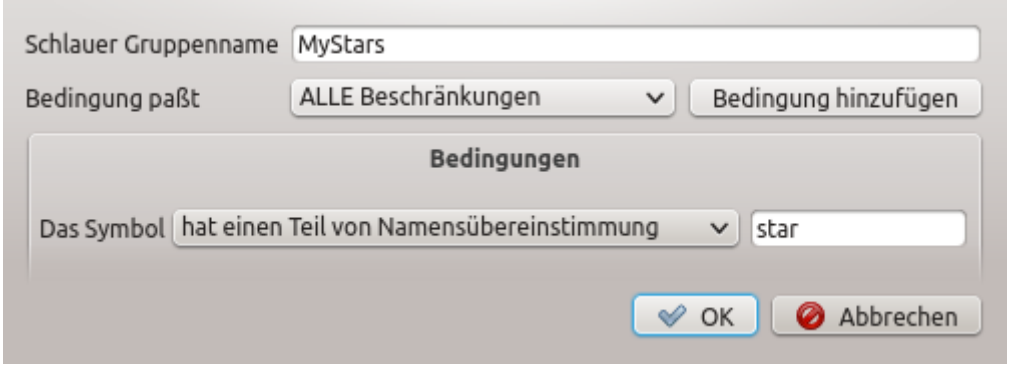


Abb. 10.2: Erstellen einer schlauen Gruppe

Tags und schlaue Gruppen schließen sich nicht gegenseitig aus: Sie sind einfach zwei verschiedene Möglichkeiten, Ihre Stilelemente zu organisieren. Im Gegensatz zu den intelligenten Gruppen, die ihre zugehörigen Elemente automatisch auf der Grundlage der eingegebenen Einschränkungen abrufen, werden Tags vom Benutzer gefüllt. Um eine dieser Kategorien zu bearbeiten, können Sie entweder:

- Wählen Sie die Elemente aus, klicken Sie mit der rechten Maustaste und wählen Sie *Hinzufügen zu Tag* ► und wählen Sie dann den Tag-Namen oder erstellen Sie ein neues Tag;
- Wählen Sie das Tag aus und drücken Sie *Gruppe ändern…* ► *Ausgewähltes Tag an Symbole anhängen*. Neben jedem Element wird ein Kontrollkästchen angezeigt, mit dessen Hilfe Sie es auswählen bzw. die Auswahl aufheben können. Wenn die Auswahl abgeschlossen ist, drücken Sie *Gruppe ändern…* ► *Markieren beenden*.
- wählen Sie die intelligente Gruppe aus, drücken Sie die Taste *Gruppe ändern…* ► *Intelligente Gruppe bearbeiten…* und konfigurieren Sie einen neuen Satz von Einschränkungen im Dialog *Smart Group Editor*. Diese Option ist auch über das Kontextmenü der intelligenten Gruppe verfügbar.

Um einen Tag oder eine schlaue Gruppe zu entfernen, klicken Sie mit der rechten Maustaste darauf und wählen Sie die Schaltfläche *Entfernen*. Beachten Sie, dass dadurch die in der Kategorie gruppierten Elemente nicht gelöscht werden.

Das Dialogfeld *Stilverwaltung* zeigt in der Mitte einen Rahmen mit Vorschauelementen, die in Reitern organisiert sind:

- *Alle* für die vollständige Auflistung von Punkt-, Linien- und Flächensymbolen und Beschriftungseinstellungen sowie vordefinierte Farbrampen und Textformaten;
- *Markierung* nur für Punktsymbole;
- *Linie* nur für lineare Symbole;
- *Füllung* nur für Flächensymbole;
- *Farbverlauf*;
- abc *Textformat*, um *Textformate* zu verwalten, die die Schriftart, Farbe, Puffer, Schatten und Hintergründe von Texten speichern (d.h. alle Formatierungsteile der Beschriftungseinstellungen, die zum Beispiel in Layouts verwendet werden können);
- *Beschriftungseinstellungen* zur Verwaltung von *Beschriftungseinstellungen*, die die Textformate und einige Layer-spezifische Einstellungen wie Platzierung der Beschriftung, Priorität, Beschriftung, Rendering, usw. beinhalten.
- *Legendensymbole* zur Verwaltung von benutzerdefinierten Legendensymbolen, die *Markierung*, *Linie* und *Füllung* umfassen.

- *3D-Symbole* zur Konfiguration von Symbolen mit *3D-Eigenschaften* (Extrusion, Schattierung, Höhe, …) für die Objekte, die in einer *3D-Kartenansicht* dargestellt werden sollen.

Sie können die Stile in *Icon-Ansicht* oder in *Listen-Ansicht* auf der unteren rechten Seite anordnen. In beiden Ansichten zeigt der Tooltip eine größere Darstellung des Stils. Mit dem Schieberegler für die Miniaturgröße links neben den Symbolen können Sie die Größe der Miniaturansichten im Dialog anpassen, um eine bessere Vorschau der Symbole zu erhalten.

Hinzufügen, Bearbeiten oder Entfernen eines Elements

Wie Sie bereits gesehen haben, werden die Stilelemente in verschiedenen Reitern aufgelistet, deren Inhalt von der aktiven Kategorie (Tag, intelligente Gruppe, Favoriten…) abhängt. Wenn ein Reiter aktiviert ist, können Sie:

- Ein neues Element hinzufügen: Drücken Sie die Taste Element hinzufügen und konfigurieren Sie das Element mit einer der folgenden Beschreibungen: *Symbol*, *Farbverlauf* oder *Textformat und Beschriftung*.
- Ein bestehendes Element ändern: Wählen Sie ein Element aus und drücken Sie die Taste Element bearbeiten und konfigurieren Sie es wie oben beschrieben.
- Ein vorhandenes Element löschen: Um ein nicht mehr benötigtes Element zu löschen, markieren Sie es und klicken Sie auf Symbol entfernen (auch mit der rechten Maustaste möglich). Das Element wird aus der lokalen Datenbank gelöscht.

Beachten Sie, dass die Registerkarte *Alle* Zugang zu allen Elementen in der Stilverwaltung bietet.

Wenn Sie mit der rechten Maustaste auf ein oder mehrere Objekte klicken, haben Sie folgende Optionen:

- *Zu Favoriten hinzufügen*;
- *Aus Favoriten entfernen*;
- *Zu Tag hinzufügen* ► wählen Sie den entsprechenden Tag aus oder erstellen Sie einen neuen; die aktuell zugewiesenen Tags sind markiert;
- *Tags leeren*: alle Tags des Symbols werden gelöscht
- *Element(e) löschen…*;
- *Element bearbeiten…*: gilt auch bei mehreren ausgewählten Objekten nur für das jeweils direkt angeklickte;
- *Element kopieren*;
- *Element einfügen…*: Fügt das Element in eine der Kategorien des Stilmanagers oder an anderer Stelle in QGIS (Symbol- oder Farbbuttons) ein;
- *Gewählte(s) Symbol(e) als PNG exportieren…* (nur für Symbole verfügbar);
- *Gewählte(s) Symbol(e) als SVG exportieren…* (nur für Symbole verfügbar);

Gemeinsame Nutzung von Stilen

Das *Import/Export* Werkzeug, links unten im Dialog der Stilverwaltung, bietet Optionen, um Symbole, Farbrampen, Textformate und Beschriftungseinstellungen mit anderen zu teilen. Diese Optionen sind auch durch Rechtsklick auf die Elemente verfügbar.

Elemente exportieren

Sie können eine Reihe von Elementen in eine `.XML`-Datei exportieren:

1. Erweitern Sie das Dropdown-Menü *Import/Export* und wählen Sie *Elemente exportieren…*
2. Wählen Sie die Elemente, die Sie einbinden möchten. Die Auswahl kann mit der Maus oder über ein zuvor festgelegtes Tag oder eine Gruppe erfolgen.
3. Drücken Sie *Exportieren*, wenn Sie bereit sind. Sie werden aufgefordert, das Ziel der gespeicherten Datei anzugeben. Der XML-Export erzeugt eine einzige Datei, die alle ausgewählten Elemente enthält. Diese Datei kann dann in die Stilbibliothek eines anderen Benutzers importiert werden.

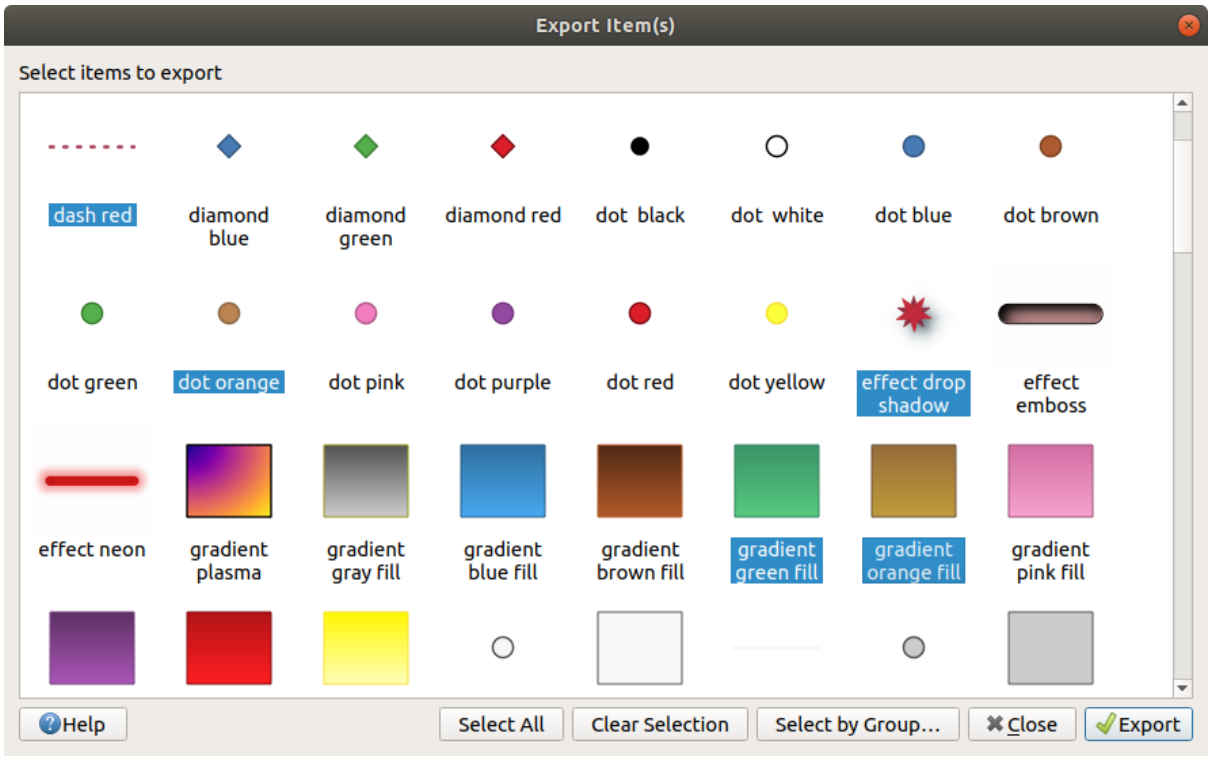


Abb. 10.3: Stil-Elemente exportieren

Wenn Symbole ausgewählt sind, können Sie sie auch als `.PNG` oder `.SVG` exportieren. Der Export nach `.PNG` oder `.SVG` (beides nicht verfügbar für andere Stilelementtypen) erzeugt eine Datei für jedes ausgewählte Symbol in einem bestimmten Ordner. Der SVG-Ordner kann dem *SVG-Pfade* im *Einstellungen* ► *Optionen* ► *System*-Menü eines anderen Benutzers hinzugefügt werden, sodass dieser direkten Zugriff auf alle diese Symbole hat.

Elemente importieren

Sie können Ihre Stilbibliothek erweitern, indem Sie neue Elemente importieren:

1. Erweitern Sie das Dropdown-Menü *Import/Export* und wählen Sie *Import Item(s)* am linken unteren Rand des Dialogs.
2. Geben Sie in dem neuen Dialogfeld die Quelle der Stilelemente an (dies kann eine `.xml`-Datei auf der Festplatte oder eine URL sein).
3. Legen Sie für die zu importierenden Elemente die Option *Zu Favoriten hinzufügen* fest.
4. Markieren Sie **|unmarkiert|** *Eingebettete Tags nicht importieren*, um den Import von Tags zu vermeiden, die mit den zu importierenden Elementen verbunden sind.

5. Geben Sie den Namen eines beliebigen *Zusätzlichen Tags* an, das auf die neuen Elemente Anwendung finden soll.
6. Wählen Sie in der Vorschau die Symbole aus, die Sie zu Ihrer Bibliothek hinzufügen möchten.
7. Drücken Sie *Import*.

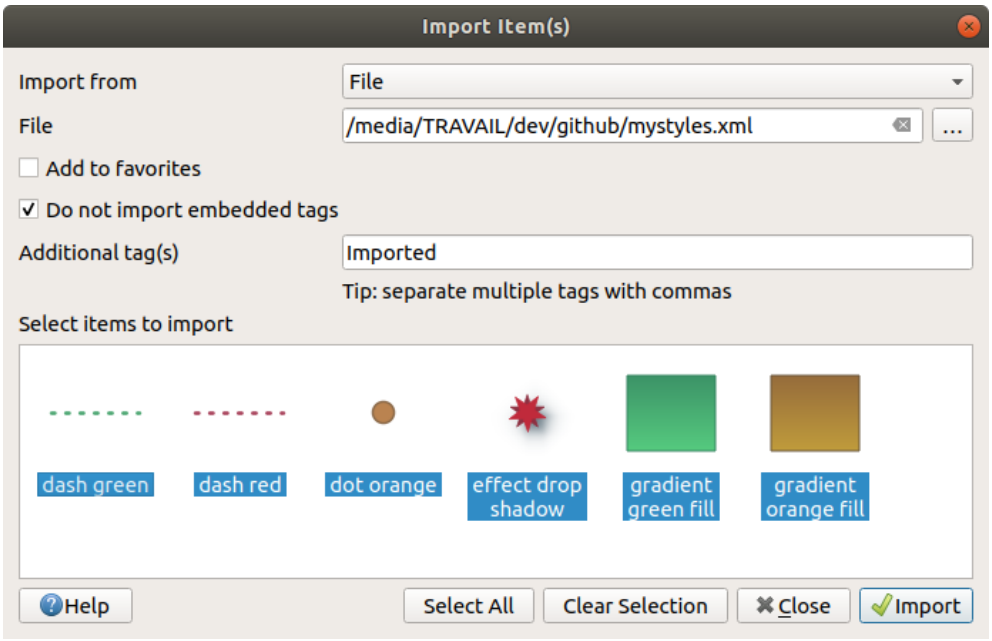


Abb. 10.4: Stil-Elemente importieren

Verwendung des Browser-Bedienfelds

Es ist auch möglich, Stilelemente direkt aus dem *Browser*-Bedienfeld in die Stildatenbank des aktiven Benutzerprofils zu importieren:

1. Wählen Sie die `.xml`-Stildatei im Browser
2. Ziehen Sie sie auf die Kartenansicht oder klicken Sie mit der rechten Maustaste und wählen Sie *Style importieren…*
3. Füllen Sie den *Elemente importieren*-Dialog wie in *Elemente importieren* aus
4. Drücken Sie *Import* und die ausgewählten Stilelemente werden der Stildatenbank hinzugefügt

Ein Doppelklick auf die Stildatei im Browser öffnet den Dialog *Stilverwaltung*, der die Elemente in der Datei anzeigt. Sie können diese auswählen und mit *Kopieren in Standardstil…* in die aktive Stildatenbank importieren. Den Elementen können Tags zugewiesen werden. Der Befehl *Stil öffnen…* ist auch über die rechte Maustaste verfügbar.

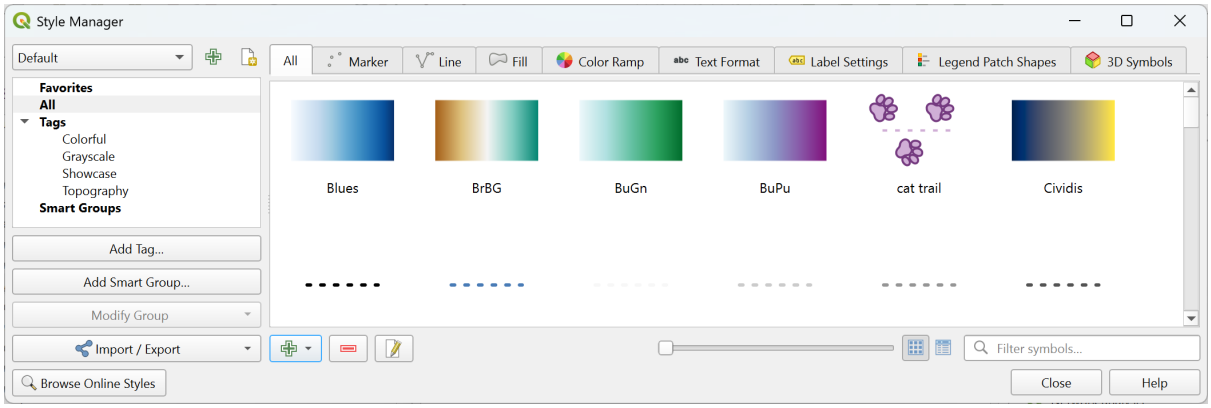


Abb. 10.5: Öffnen einer Datei mit Stilelementen

Der Dialog erlaubt auch den Export einzelner Symbole als `.PNG` oder `.SVG` Dateien.

Nutzung des Online-Repository

Das QGIS-Projekt unterhält ein Repositorium mit einer Sammlung von Stilen, die von QGIS-Benutzern genutzt werden können. Dieses ist unter https://hub.qgis.org/styles verfügbar und kann über den Dialog *Stilverwaltung* aufgerufen werden, indem Sie unten auf die Schaltfläche *Onlinestile durchsuchen* klicken.

Von diesem Repository aus können Sie:

1. Blättern und suchen Sie nach beliebigen Stilelementen, basierend auf deren Typ oder Namen
2. Laden Sie die Stildatei herunter und entpacken Sie sie
3. Laden Sie die `.xml`-Datei in Ihre Stildatenbank in QGIS, indem Sie eine der oben genannten Importmethoden verwenden.

10.1.2 Einstellen einer Farbrampe

Der Reiter „Farbverlauf“ im Dialogfeld *Stilverwaltung* hilft Ihnen bei der Vorschau verschiedener Farbrampen auf der Grundlage der im linken Bedienfeld ausgewählten Kategorie.

Um eine benutzerdefinierte Farbrampe zu erstellen, aktivieren Sie den Reiter Farbrampe und klicken Sie auf die Schaltfläche Element hinzufügen. Die Schaltfläche öffnet eine Dropdown-Liste zur Auswahl des Rampentyps:

- *Gradient*: gibt eine Start- und Endfarbe an und erzeugt eine Farbrampe, die **fortlaufend** oder **diskret** sein kann. Mit einem Doppelklick auf die Rampenvorschau können Sie beliebig viele Zwischenfarben hinzufügen. Klicken Sie auf den Farbstop-Indikator und unter *Gradientenstop* können Sie:
  - seine *Relative Position* vom Startpunkt der Farbrampe aus einstellen. Es ist auch möglich, den Indikator mit der Maus zu ziehen oder die Pfeiltasten zu drücken (für eine größere Verschiebung mit der Taste `Shift` kombinieren).
  - das Farbmodell angeben, das bei der Interpolation zwischen Farben verwendet werden soll: Es kann *RGB*, *HSL* oder *HSV* sein. Unter bestimmten Umständen kann diese Option dazu beitragen, entsättigte Mitteltöne zu vermeiden, was zu visuell ansprechenderen Verläufen führt.
  - legt die Richtung fest, der die Interpolation für die **Ton**-Komponente einer *HSL*- oder *HSV*-Farbangabe folgen soll. Sie kann *Mit dem Uhrzeigersinn* oder *Gegen den Uhrzeigersinn* sein.
  - die *Farbeigenschaften* setzen
  - mit *Stopp löschen* oder `DEL` einen Gradientenstopp entfernen

Der Bereich *Plot* bietet eine weitere, grafische Möglichkeit, die Farbrampe zu gestalten, indem die Position oder die Deckkraft und die HSL-Komponenten der Farbstopps geändert werden.

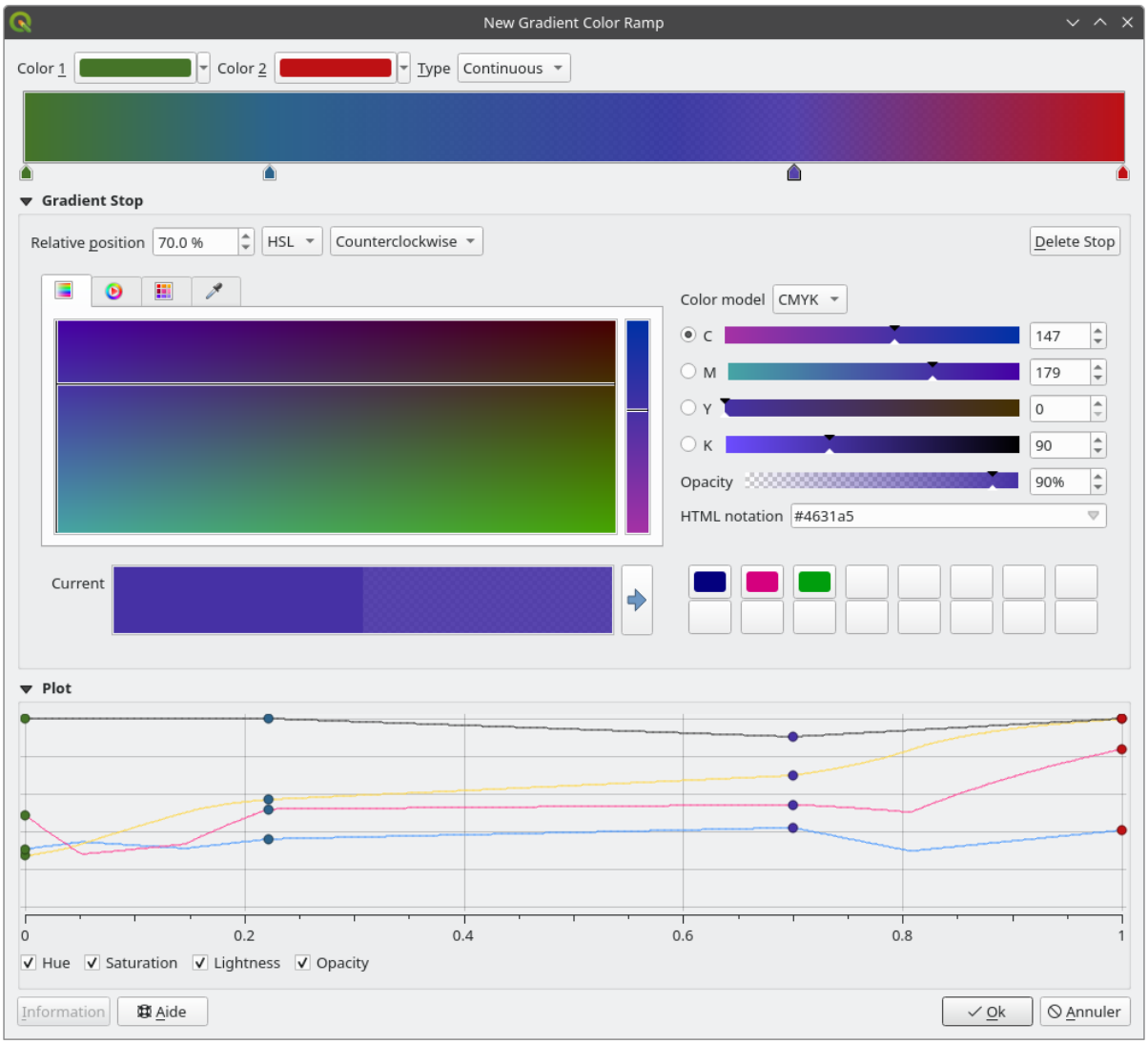


Abb. 10.6: Beispiel einer benutzerdefinierten Farbverlaufsrampe mit mehreren Stopps

> **Hinweis**
>
> Durch Ziehen und Ablegen einer Farbe aus einem Farbpunkt auf die Farbverlaufsrampenvorschau wird ein neuer Farbstopp hinzugefügt.

- *Farbvorgaben*: ermöglicht die Erstellung einer Farbrampe, die aus einer Liste von Farben besteht, die vom Benutzer ausgewählt wurden;
- *Zufällig*: erzeugt eine zufällige Auswahl von Farben auf der Grundlage des Wertebereichs für *Ton*, *Sättigung*, *Wert* und *Deckkraft* und einer Anzahl von Farben (*Klassen*);
- *Katalog: ColorBrewer*: eine Reihe vordefinierter diskreter Farbverläufe, bei denen Sie die Anzahl der Farben in der Rampe anpassen können;
- *Katalog: cpt-city*: Zugriff auf einen ganzen Katalog von Farbverläufen, diese können Sie lokal *als Standardgradiente speichern*. Die Option cpt-city öffnet einen neuen Dialog mit Hunderten von Themen, die bereits im Lieferumfang enthalten sind.

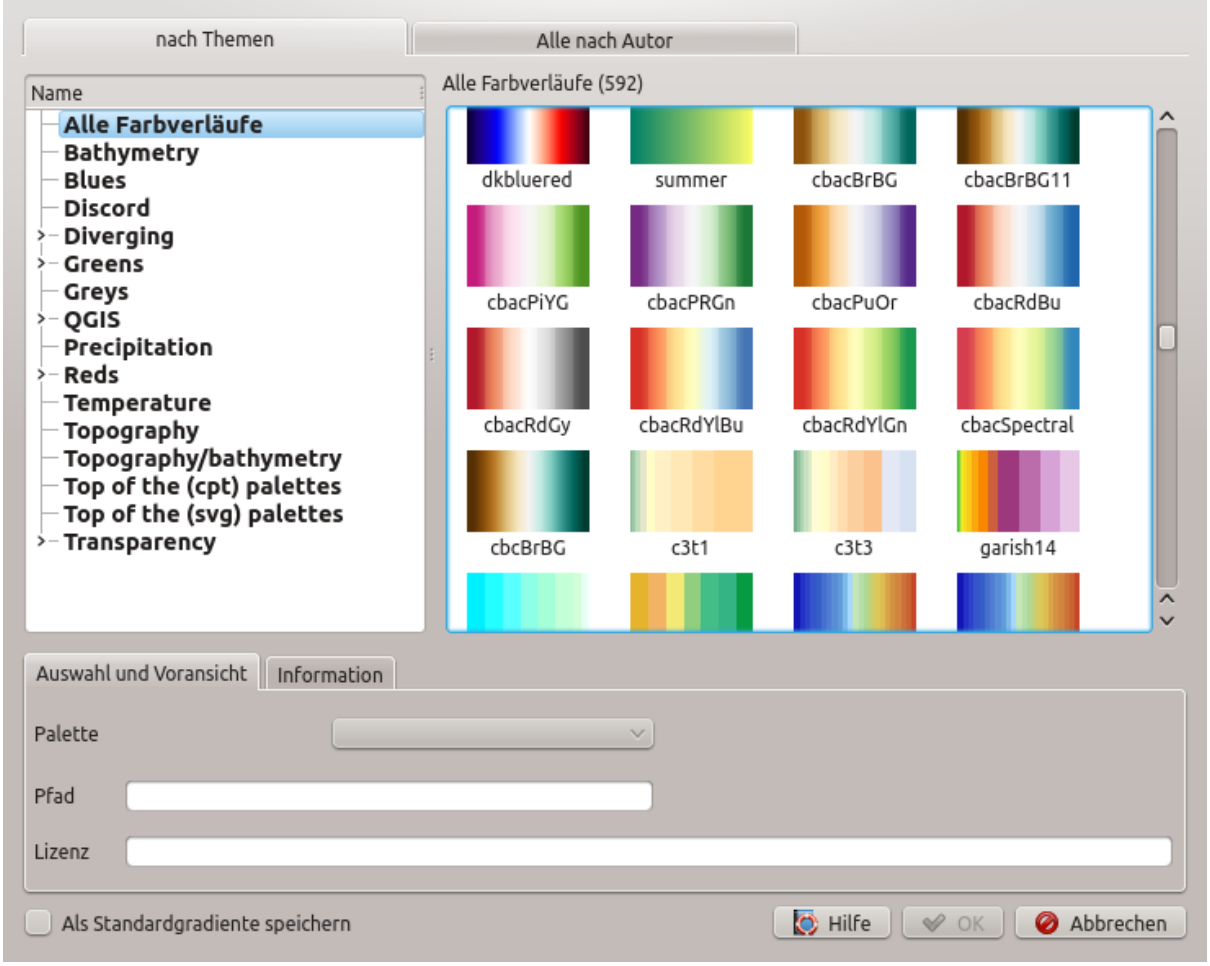


Abb. 10.7: Der cpt-city-Dialog mit hunderten von Farbrampen

10.1.3 Ein Legendensymbol erstellen

Um ein neues Legendensymbol zu erstellen, aktivieren Sie den Reiter *Legendensymbol* und klicken auf die Schaltfläche Element hinzufügen. Die Schaltfläche öffnet eine Dropdown-Liste zur Auswahl des Geometrietyps:

- *Markierungslegendensymbol…*: zur Verwendung mit Punktgeometrien.
- *Linienlegendensymbol…*: zur Verwendung mit Liniengeometrien.
- *Legendensymbol füllen…*: zur Verwendung mit Polygongeometrien.

Bei allen drei Optionen wird derselbe Dialog angezeigt.

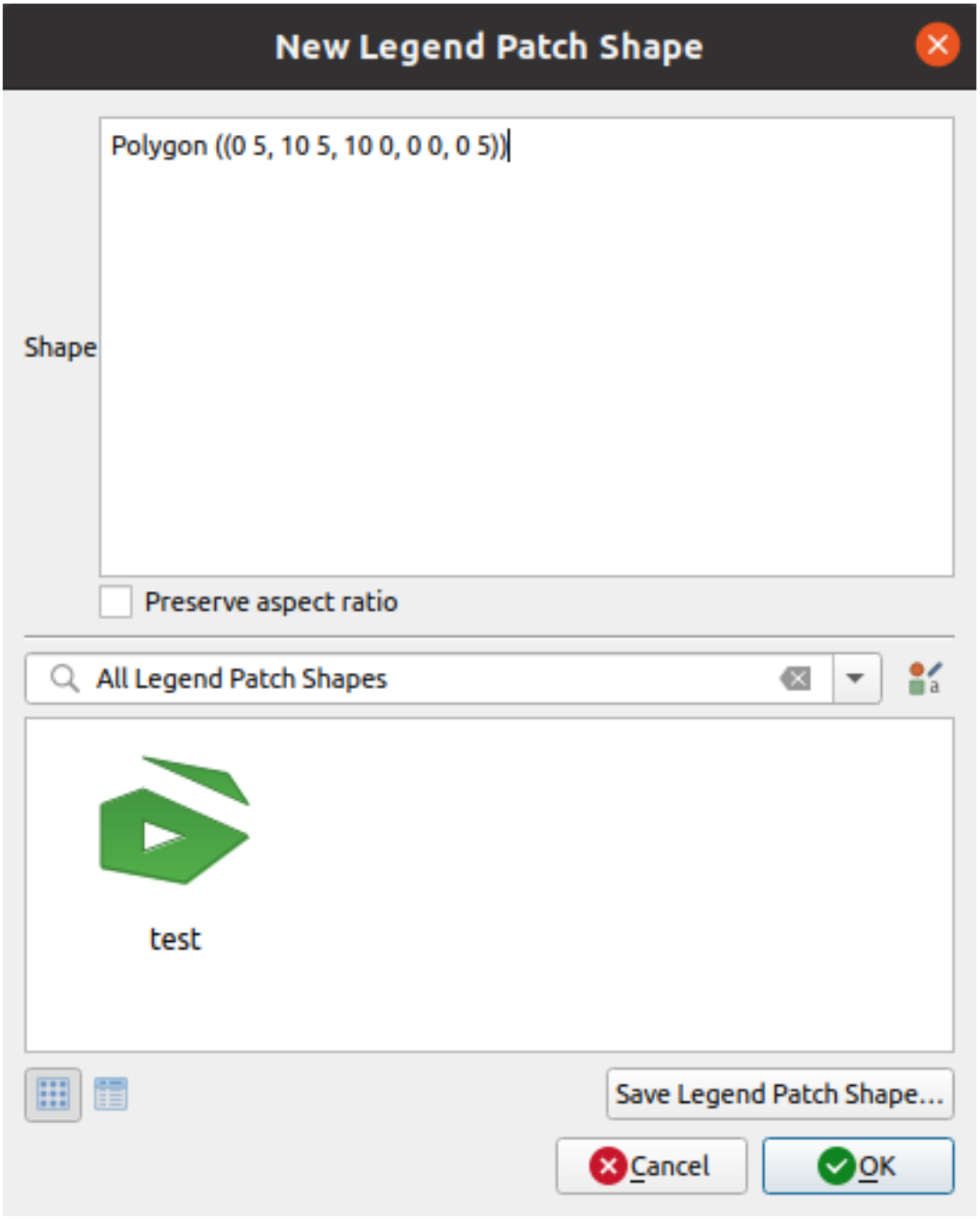


Abb. 10.8: Erstellen Sie eine neues Legendensymbol

Nur der Formtyp und die angezeigten Legendenfelder unterscheiden sich je nach gewähltem Geometrietyp. Die folgenden Optionen sind verfügbar:

- *Form*: hier ist die Form des Legendenfeldes als WKT-String zu definieren. Es können ein- und mehrteilige Geometrien verwendet werden, aber nicht GeometryCollection.
- *Seitenverhältnis beibehalten*
- *Icon Ansicht* oder *Listen Ansicht* der verfügbaren Formen von Legendensymbolen, gefiltert nach Tags.

Wenn das neue Legendensymbol definiert ist, können Sie *Legendensymbol speichern…* oder *OK* drücken, die beide zum gleichen Dialog führen.

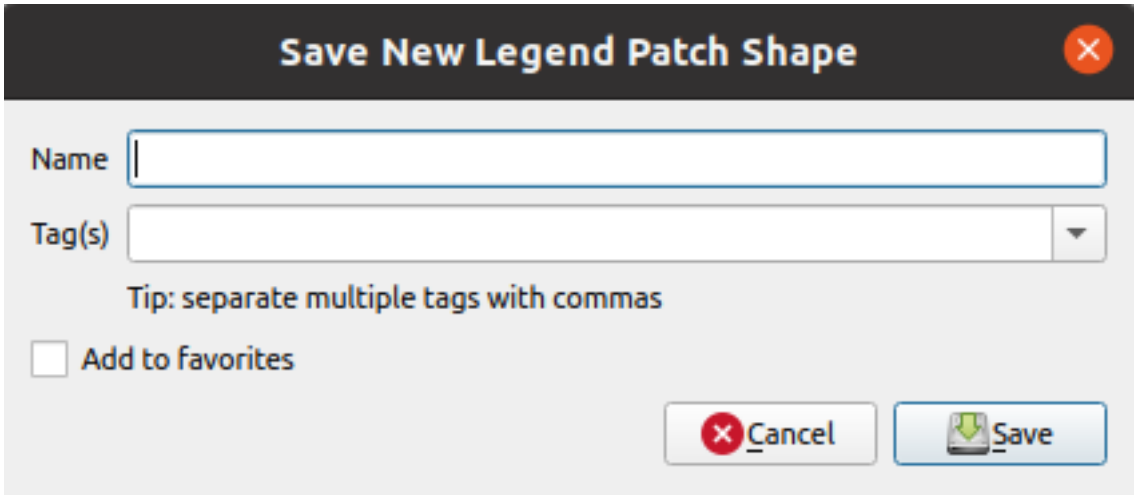


Abb. 10.9: Neues Legendensymbol speichern

Hier müssen Sie einen Namen und Tags wählen, um das Symbol zu beschreiben und ob sie zu den Favoriten hinzugefügt werden soll.

Wenn Sie *Speichern…* drücken, wird die Form der Liste hinzugefügt und Sie werden zurück zum Dialog *Neues Legendensymbol* geleitet, und können weitere, neue Formen zu erstellen.

10.2 Die Symbolauswahl

Die Symbolauswahl ist das Hauptdialogfeld zur Gestaltung eines Symbols. Sie können Markierungs-, Linien- oder Füllsymbole erstellen oder bearbeiten.

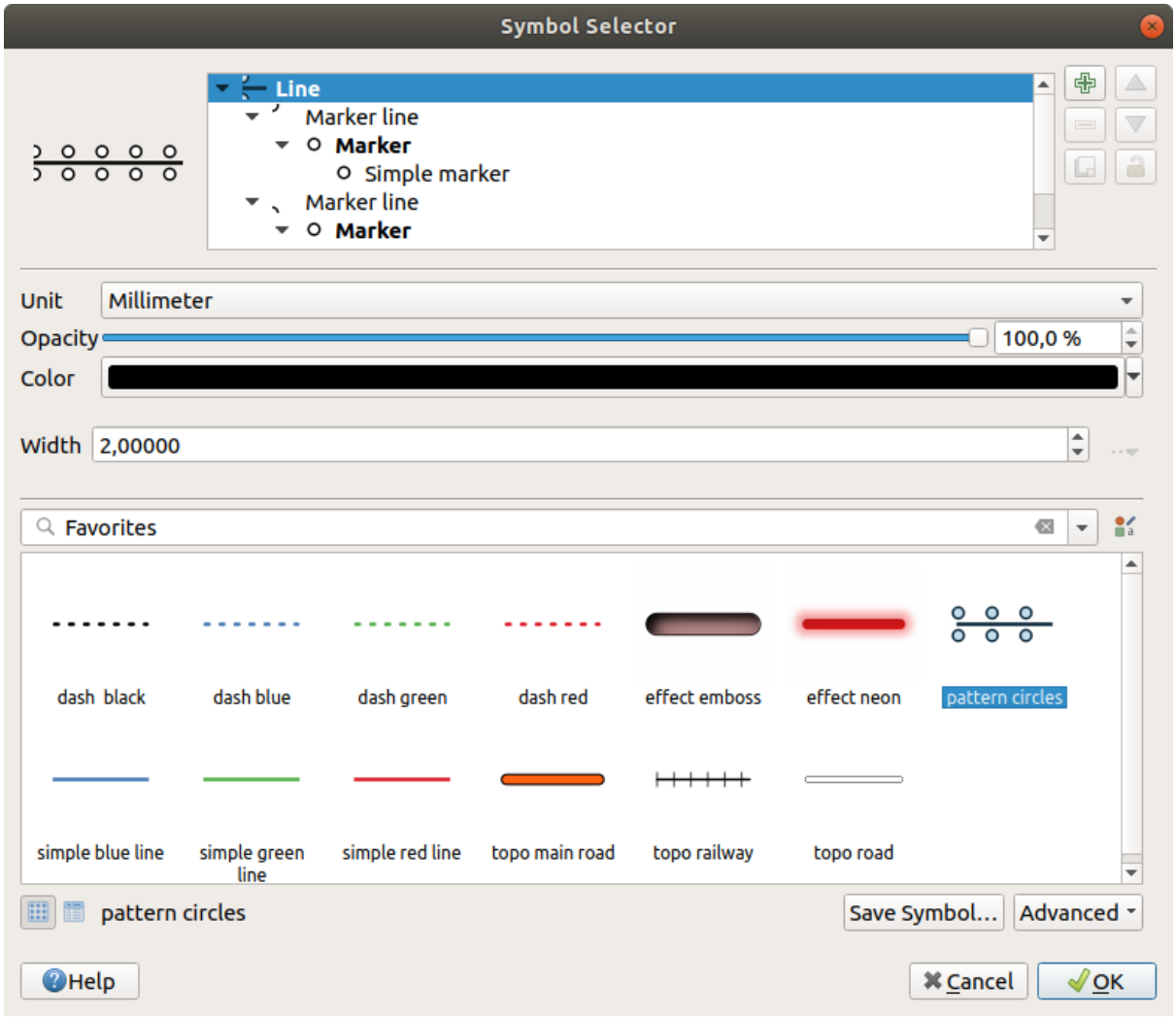


Abb. 10.10: Entwerfen eines Liniensymbols

Zwei Hauptkomponenten strukturieren den Symbolauswahldialog:

- der Symbolbaum mit den Symbollayern, die anschließend zu einem neuen gemeinsamen Symbol kombiniert werden
- und Einstellungen zur Konfiguration des ausgewählten Layers in der Baumstruktur.

10.2.1 Der Symbollayer-Baum

Ein Symbol kann aus mehreren *Symbollayern* bestehen. Der Symbolbaum zeigt die Überlagerung dieser Layer, die anschließend zu einem neuen globalen Symbol kombiniert werden. Außerdem wird eine dynamische Symboldarstellung aktualisiert, sobald sich Symboleigenschaften ändern.

Je nach ausgewählter Ebene in den Symbolbaumelementen werden verschiedene Werkzeuge zur Verfügung gestellt, die Sie bei der Verwaltung des Baums unterstützen:

- Symbollayer hinufügen: Sie können so viele Symbole übereinander legen, wie Sie möchten.
- den ausgewählten Symbollayer entfernen
- Farben des Layer sperren: eine gesperrte Farbe bleibt unverändert, wenn der Benutzer die Farbe auf der globalen (oder oberen) Symbolebene ändert

- eine(n) (Gruppe von) Symbollayer(n) duplizieren
- den Symbollayer nach oben oder unten verschieben

10.2.2 Ein Symbol konfigurieren

In QGIS kann die Konfiguration eines Symbols auf zwei Ebenen vorgenommen werden: auf Ebene des Symbol und auf Ebene des Symbollayers.

Das Symbol

Auf der obersten Ebene des Baums ist die Layer-Geometrie festgelegt, die vom Typ **Markierung**, **Linie** oder **Füllung** sein kann. Jedes Symbol kann ein oder mehrere Symbole (auch eines anderen Typs) oder Symbol-Layer enthalten.

Sie können einige Parameter einstellen, die für das gesamte Symbol gelten:

- *Einheit*: kann **Millimeter**, **Punkte**, **Pixel**, **Meter im Maßstab**, **Karteneinheiten** oder **Zoll** sein (siehe *Einheiten Auswahl* für weitere Einzelheiten)
- *Deckkraft*
- *Farbe*: Wenn dieser Parameter vom Benutzer geändert wird, wird sein Wert an alle nicht gesperrten Untersymbole weitergegeben.
- *Größe* and *Drehung* für Markierungssymbole
- *Breite* für Liniensymbole

> **Tipp**
>
> Verwenden Sie die Eigenschaften *Größe* (für Marker-Symbole) oder die *Breite* (für Linien-Symbole) auf der Symbolebene, um die Größe aller eingebetteten *symbol layers* Ebenen proportional zu ändern.

> **Bemerkung**
>
> Die Schaltfläche Datendefinierte Übersteuerung neben den Parametern Breite, Größe oder Drehung ist inaktiv, wenn Sie das Symbol über den Dialog der Stilverwaltung einstellen. Wenn das Symbol mit einem Karten Layer verbunden ist, hilft Ihnen diese Schaltfläche bei der Erstellung der *proportionalen oder multivariaten Analyse* Darstellung.

- Eine Vorschau auf die Symbolbibliothek gefiltert werden. Sie können die Liste der Symbole auch über die Schaltfläche Stilverwaltung aktualisieren und den Namensgeberdialog öffnen. Dort können Sie alle Möglichkeiten nutzen, die im Abschnitt *Die Stilverwaltung* beschrieben sind.

  Die Symbole werden entweder angezeigt:

  - in einer Icon-Liste (mit Miniaturbild, Name und zugehörigen Tags) mit Hilfe der Schaltfläche Listen-Ansicht unterhalb des Rahmens;
  - oder als Icon-Vorschau über die Schaltfläche Icon-Ansicht.

- Drücken Sie die Schaltfläche *Symbol speichern*, um den Dialog *Neues Symbol speichern* zu öffnen. Hier können Sie den *Zielort* wählen, an dem Sie das zu bearbeitende Symbol hinzufügen möchten, ihm einen *Namen* geben und *Tag(s)* hinzufügen. Sie haben auch die Möglichkeit, Ihr neues Symbol mit Zu Favoriten hinzufügen zu markieren.
- Mit der Option *Erweitert* können Sie:

- für Linien und Füllungen, *Objekte auf Kartenausschnitt beschneiden*.
- für Füllungen, *Rechte-Hand-Regel erzwingen*: ermöglicht es, gerenderte Füllsymbole zu zwingen, die der Standard „Rechte-Hand-Regel“ folgen (z.B. bei Polygone, bei denen der äußere Ring im Uhrzeigersinn und der inneren Ringe gegen den Uhrzeigersinn ausgerichtet ist).

  Die Orientierungskorrektur wird nur beim Rendern angewendet, die ursprüngliche Geometrie des Objekts bleibt unverändert. Dies ermöglicht die Erstellung von Füllsymbolen mit einheitlichem Erscheinungsbild, unabhängig vom gerenderten Datensatz und der Ringausrichtung der einzelnen Objekte.
- Bei Markierungs-Symbolen ermöglicht die Option *Puffereinstellungen…* das Hinzufügen eines Halo-Effekts um das Markierungs-Symbol herum, um es vor unterschiedlichen Hintergründen besser lesbar zu machen. Sie können die *Größe*, den *Verbindungsstil* und das *Symbol* des Puffers anpassen.
- 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…* zur Festlegung der Reihenfolge der Symbolwiedergabe
  - *Datendefinierte Legendengröße*
  - *Übereinstimmung mit gespeicherten Symbolen…* and *Übereinstimmung mit Symbolen aus der Datei…* um automatisch *Symbole zu Klassen zuzuordnen*

Der Symbollayer

Auf einer niedrigeren Ebene der Baumstruktur können Sie die Symbollayer anpassen. Die verfügbaren Symbollayer-Typen hängen vom oberen Symbol Typ ab. Sie können auf den Symbollayer *Zeicheneffekte* anwenden, um seine Darstellung zu verbessern.

Da es nicht möglich wäre, alle Optionen aller Layer-Typen zu beschreiben, werden im Folgenden nur bestimmte und wichtige Optionen erwähnt.

Gemeinsame Parameter

Einige allgemeine Optionen und Widgets sind verfügbar, um einen Layer zu erstellen, unabhängig davon, ob es sich um einen Marker-, Linien- oder Füllsubtyp handelt:

- das *Farbauswahl* Werkzeug zur einfachen Farb-Anpassung
- *Einheiten*: Es können **Millimeter**, **Punkte**, **Pixel**, **Meter im Maßstab**, **Karteneinheiten** oder **Zoll** sein (siehe *Einheiten Auswahl* für weitere Einzelheiten)
- das Widget Datendefinierte Übersteuerung bei fast allen Optionen, welches die Möglichkeiten der Anpassung jedes Symbols erweitert (siehe *Einrichten von datendefinierter Übersteuerung* für weitere Informationen)
- Die Option *Symbollayer einschalten* steuert die Sichtbarkeit des Symbollayers. Deaktivierte Layer werden beim Rendern des Symbols nicht gezeichnet, sondern im Symbol gespeichert. Die Möglichkeit, Layer auszublenden, ist praktisch, wenn Sie nach dem besten Design Ihres Symbols suchen, da Sie zum Testen keine Layer entfernen müssen. Die datenbasierte Überschreibung ermöglicht es, verschiedene Layer auf der Grundlage von Ausdrücken (z. B. mit Hilfe von Objektattributen) aus- oder einzublenden.
- das *Zeicheneffekte* Kontrollkästchen, um *Zeicheneffekte* zu aktivieren.

Bemerkung

Die folgende Beschreibung geht davon aus, dass der Symbollayer-Typ an die Geometrie des Objekts gebunden ist. In diesem Fall kann der untergeordnete Layer-Parameter (Platzierung, Offset…) an das übergeordnete Symbol gebunden sein und nicht an die Objektgeometrie selbst.

Markierungs-Symbole

Für Punkt-Objekte gibt es mehrere *Symbollayertypen*:

Tab. 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		

- **Einfache Markierung** (Standard)

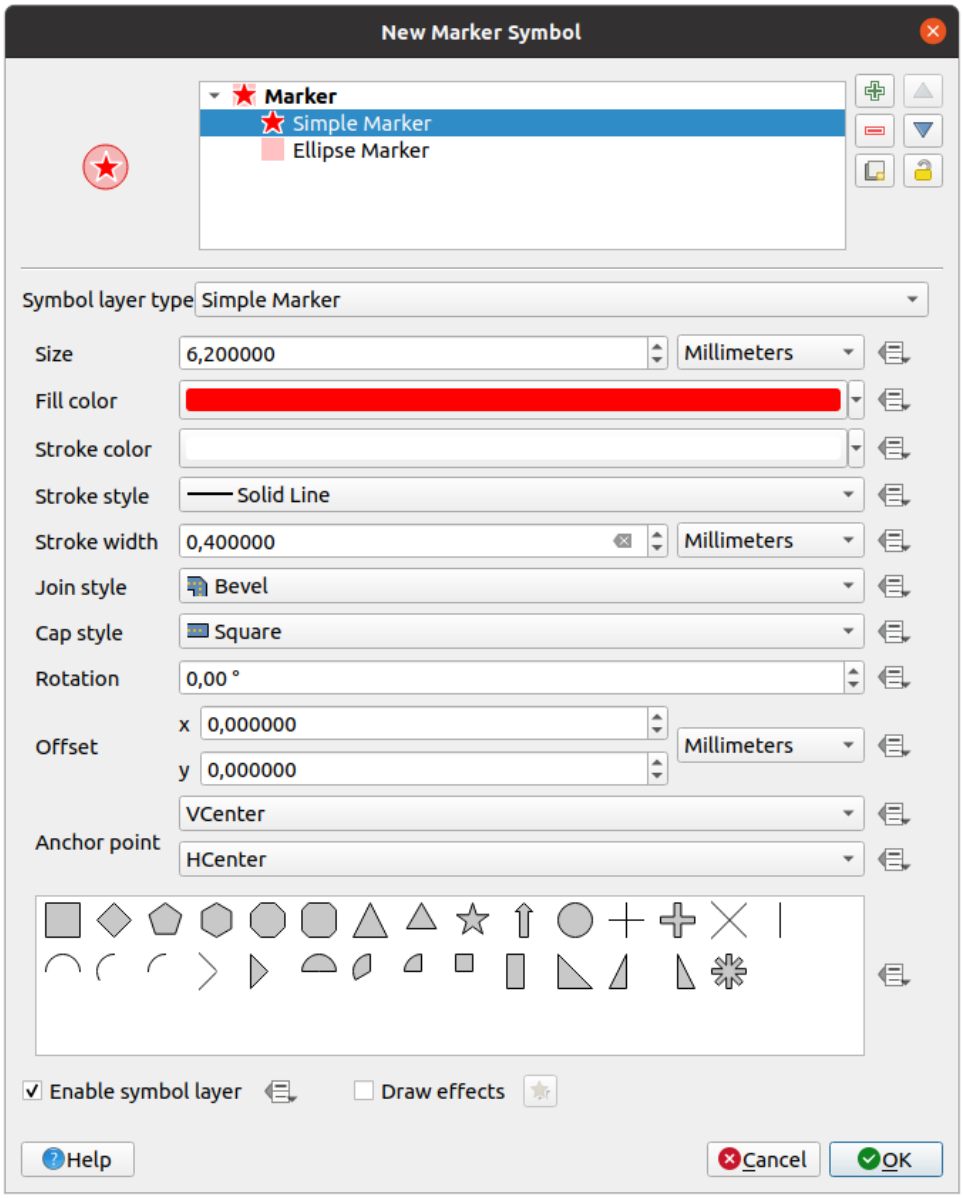


Abb. 10.11: Dialog zum Gestalten einer einfachen Markierung

Der Layer-Typ des einfachen Markierungs-Symbols hat die folgenden Eigenschaften:

- *Größe* in verschiedenen Einheiten
- *Füllfarbe*
- *Strichfarbe*, *Strichstil* (aus einer vordefinierten Liste) und *Strichbreite*
- *Verbindungsstil* hat die Optionen: **Abgefast**, **Eckig** und **Rund**
- *Endenstil* hat die Optionen **Quadratisch**, **Flach** und **Rund**
- *Drehung*
- *Versatz* in *X* und *Y* Ausdehnungen des Objekts
- *Ankerpunkt*: definiert den Quadrantenpunkt des Symbol, der als Platzierungsursprung festgelegt werden soll. Dies ist der Punkt, auf den der *Versatz* angewendet wird.

- **Animierte Markierung** (siehe *Animierte Markierung*)
- **Ellipsenmarkierung**: ein einfacher Layer für Markierungssymbole, mit anpassbarer Breite und Höhe

- **Gefüllte Markierung**: ähnelt dem Typ „einfache Markierung“, mit dem Unterschied, dass er ein *Füllungs-Symbol* zur Darstellung der Markierung verwendet. Dies ermöglicht die Verwendung aller vorhandenen QGIS-Füllungs- (und Strich-) Stile für die Darstellung von Markern, z.B. Farbverläufe oder Shapeburst-Füllungen.
- **Schriftmarkierung**: ähnelt dem Typ „einfache Markierung“ , mit dem Unterschied, dass er installierte Schriftarten zur Darstellung der Markierung verwendet. Hier gibt es zusätzlichen Eigenschaften:
  - *Schriftfamilie*
  - *Schriftartstil*
  - *Zeichen*, die den Text darstellen, der als Symbol angezeigt werden soll. Sie können eingegeben oder aus dem Schriftzeichen-Sammel-Widget ausgewählt werden, und Sie können sie mit den gewählten Einstellungen als *Vorschau* ansehen.
- **Geometriegenerator** (siehe *Der Geometrie-Generator*)
- **Maske**: dies Untersymbol definiert eine Maskenform, deren Farbeigenschaft ignoriert wird und nur die Deckkraft verwendet wird. Dies ist praktisch, wenn sich das Markierungssymbol mit Beschriftungen oder anderen Symbolen überschneidet, deren Farben nahe beieinander liegen, sodass sie schwer zu unterscheiden sind. Mehr Details unter *Eigenschaften Masken*.
- **Rasterbildmarkierung**: verwendet ein Bild (`PNG`, `JPG`, `BMP` …) als Markierungssymbol. Das Bild kann eine Datei auf der Festplatte, eine Remote-URL, eingebettet in die Style-Datenbank (*weitere Details*) oder als base64-String kodiert sein. Breite und Höhe des Bildes können unabhängig voneinander oder unter Verwendung von Seitenverhältnisse beibehalten festgelegt werden. Die Größe kann mit einer der *Gemeinsame Einheiten* oder als Prozentsatz der Originalgröße des Bildes (skaliert mit der Breite) angegeben werden.
- **Vektorfeldmarkierung** (siehe *Die Vektorfeld-Markierung*)
- **SVG-Markierung**: stellt Ihnen Bilder aus Ihren SVG-Pfaden (festgelegt im Menü *Einstellungen* ► *Optionen…* ► *System*) zum Rendern als Markierungssymbol zur Verfügung. Breite und Höhe des Symbols können unabhängig voneinander oder mithilfe von Seitenverhältnisse beibehalten eingestellt werden. Jede SVG-Datei kann auch in Farbe und Kontur angepasst werden. Das Bild kann eine Datei auf der Festplatte, eine Remote-URL, eingebettet in die Stildatenbank (*Weitere Details*) oder als base64-String kodiert sein.

  Das Symbol kann auch mit *Dynamische SVG-Parameter* gesetzt werden. Siehe *Parametrisierbare SVG* Abschnitt, um ein SVG-Symbol zu parametrisieren.

  Bemerkung

  Anforderungen an die SVG-Version

  QGIS rendert SVG-Dateien, die dem SVG Tiny 1.2-Profil folgen, das für die Implementierung auf einer Reihe von Geräten gedacht ist, von Mobiltelefonen und PDAs bis hin zu Laptops und Desktop-Computern, und enthält daher eine Teilmenge der Objekte, die in SVG 1.1 Full enthalten sind, zusammen mit neuen Objekten, die die Fähigkeiten von SVG erweitern.

  Einige Objekte, die nicht in diesen Spezifikationen enthalten sind, werden in QGIS möglicherweise nicht korrekt wiedergegeben.

Linien-Symbole

Liniensymbole eignen sich für liniengeometrische Objekte und verfügen über die folgenden Layer-Typen:

Tab. 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		

- **Einfache Linie** (Standard)

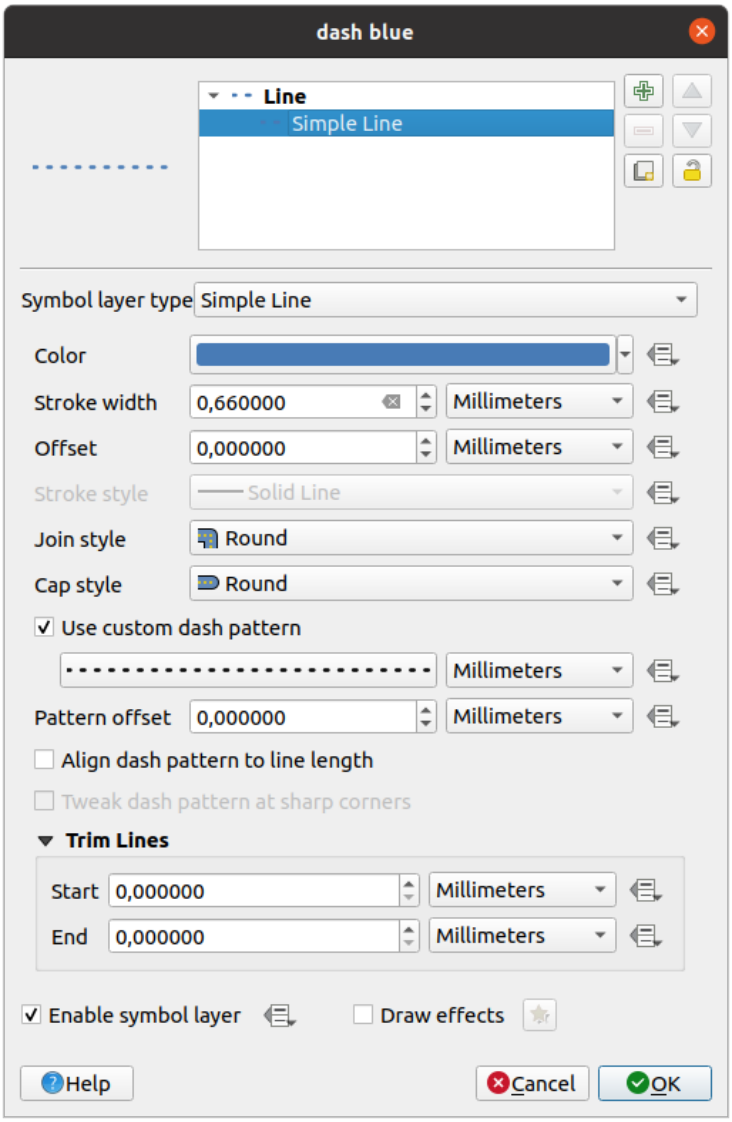


Abb. 10.12: Dialog zum Gestalten einer einfachen Linie

Der Layer-Typ „Einfache Linie“ hat viele der Eigenschaften des Layer-Typ *Einfache Markierung*, darüber hinaus aber die folgenden:

- *Benutzerdefinierte Strichlierung verwenden*: überschreibt die Einstellung *Strichstil* mit einem benutzerdefinierten Strichmuster. Sie müssen die Länge der aufeinanderfolgenden Striche und

Leerzeichen, die das Muster formen, in der gewählten Einheit definieren. Die Gesamtlänge des Musters wird im unteren Teil des Dialogs angezeigt.

  - *Musterversatz*: die Positionierung der Striche/Leerzeichen in der Zeile kann optimiert werden, sodass sie an passenderen Positionen platziert werden können, um Ecken in der Zeile zu berücksichtigen (kann auch verwendet werden, um benachbarte Strichmustergrenzen „auszurichten“).
  - *Strichlierung an Linienlänge ausrichten*: die Länge des Strichmusters wird so angepasst, dass die Zeile mit einem vollständigen Strichelement endet, anstatt mit einer Lücke.
  - *Strichelung an scharfe Kanten anpassen*: Passt die Platzierung des Strichmusters dynamisch an, sodass scharfe Ecken durch ein volles Strichelement dargestellt werden, das in die scharfe Ecke hinein- und aus ihr herausführt. Abhängig von *Strichlierung an Linienlänge ausrichten*.
  - *Linien abschneiden* vom *Beginn* und/oder vom *Ende*: ermöglicht es, bei der Liniendarstellung die ersten x mm und die letzten y mm der eigentlichen Linienfolge abzuschneiden. Es werden verschiedene *Einheiten* unterstützt, einschließlich eines prozentualen Anteils an der Gesamtlänge der Linie. Darüber hinaus kann über die datendefinierte Übersteuerung zusätzliche Kontrolle ausgeübt werden. Der Anfangs-/Endtrimmabstand kann z. B. bei der Erstellung komplexer Symbole verwendet werden, bei denen ein Layer nicht die am Anfang und Ende der Linie platzierten Layer für Markierungssymbole überlappen soll.

- **Pfeil**: Zeichnet Linien als gekrümmte oder nicht gekrümmte Pfeile mit einer einfachen oder doppelten Spitze, wobei die folgenden Details konfiguriert werden können:
  - *Kopftyp*
  - *Pfeiltyp*
  - *Pfeilbreite*
  - *Pfeilstärke am Anfang*
  - *Spitzenlänge*
  - *Spitzenstärke*
  - *Versatz*

  Darüber hinaus gibt es die Optionen *Gebogene Pfeile* (das Linienobjekt muss mindestens drei Scheitelpunkte haben) und *Pfeil auf jedem Segment wiederholen*. Für die Farbe wird eine *Füllung-Symbolebene* verwendet, was z.B. Farbverläufe oder Shapeburst erlaubt, um den Pfeilkörper darzustellen. In Kombination mit dem Geometriegenerator hilft diese Art von Layer-Symbolen bei der Darstellung von Flussdiagrammen.
- **Geometriegenerator** (siehe *Der Geometrie-Generator*)
- **Interpolierte Linie**: ermöglicht die Darstellung einer Linie, deren *Strichbreite* und/oder *Farbe* entlang der Geometrie variiert. Die folgenden Parameter sind einstellbar:
  - *Startwert* und *Endwert*: Werte, die für die Interpolation an den Endpunkten der Geometrie des Objekts verwendet werden. Es können feste Werte, Attribute des Objekts oder auf einem Ausdruck basierende Werte sein.
  - *Min. Wert* und *Max. Wert*: Werte, zwischen denen die Interpolation durchgeführt wird. Drücken Sie die Schaltfläche Laden, um sie automatisch auf der Grundlage der auf den Layer angewendeten minimalen und maximalen Start-/Endwerte zu füllen.
  - Nur verfügbar für die Strichbreite:
    * *Min. Breite* und *Max. Breite*: definieren den Bereich der variierenden Breite. *Min. Breite* ist mit dem *Min. Wert* verknüpft und *Max. Breite* dem *Max. Wert*‘. *Eine :ref:`Einheit <unit_selector>* kann zugeordnet werden.
    * *Absoluten Wert verwenden*: nur absoluten Wert für Interpolation berücksichtigen (negative Werte werden als positiv verwendet).

    * *Werte außerhalb des Bereichs ignorieren*: Wenn der `[Startwert - Endwert]`-Bereich eines Objekts nicht im `[Minimalwert - Maximalwert]`-Bereich enthalten ist, werden die außerhalb des Bereichs liegenden Teile der Geometrie des Objekts standardmäßig mit der minimalen oder maximalen Breite gerendert. Aktivieren Sie diese Option, um sie überhaupt nicht darzustellen.

  – Um die Farbe zu variieren, können Sie jede der Interpolationsmethoden von :ref:`Farbverlaufs-Klassifikationen <color_ramp_shader>`verwenden.

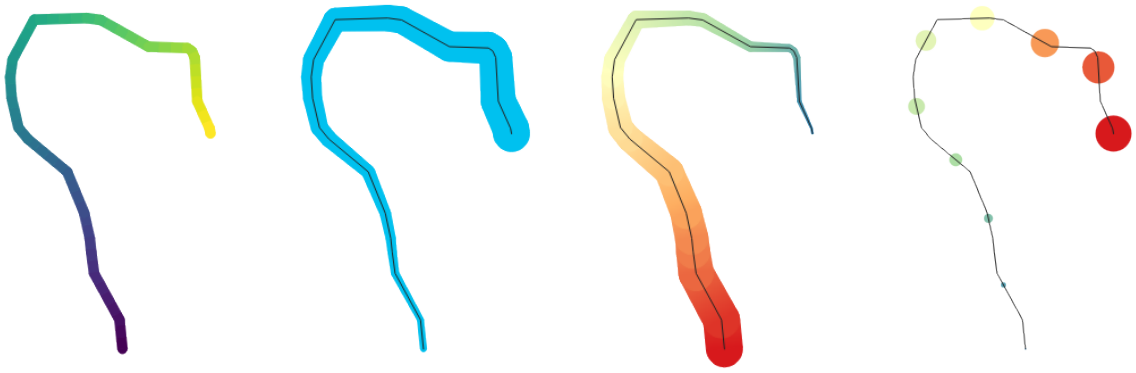


Abb. 10.13: Beispiele für interpolierte Linien

- **Markierungslinie**: Wiederholt ein *Markierungssymbol* über die Länge einer Linie.
  – Die *Markierungsplatzierung* kann mithilfe einer regelmäßigen Intervalleinstellung (ausgehend vom ersten Stützpunkt) und/oder der Liniengeometrieeigenschaft (am ersten oder letzten Stützpunkt, an inneren Stützpunkten, am zentralen Punkt der Linie oder jedes Segments oder an jedem Kurvenpunkt) festgelegt werden.
  – Wenn die Platzierung am ersten oder letzten Stützpunkt aktiviert ist, bewirkt die Option *Auf jedem Außenteil platzieren*, dass die Marker auch am ersten oder letzten Stützpunkt für jedes Teil einer mehrteiligen Geometrie gerendert werden.
  – *Entlang der Linie versetzen*: die Platzierung der Marker kann auch mit einem Versatz entlang der Linie versehen werden, wobei verschiedene ref:*Einheiten <unit_selector>* zur Wahl stehen (Millimeter, Punkte, Karteneinheit, Meter im Maßstab, Prozent, …):
    * Ein positiver Wert verschiebt die Markierungssymbole in Linienrichtung (bei den Platzierungen *Auf ersten Stützpunkt* und *Mit Intervall*) und nach hinten (bei der Platzierung *Auf letzten Stützpunkt*).
    * Ein negativer Wert auf einer nicht geschlossenen Linie führt zu keinem Versatz (für *Auf ersten Stützpunkt* und *Auf letzten Stützpunkt*) oder zu einem Rückwärtsversatz der Symbole (ab dem letzten Stützpunkt).
    * Bei einem geschlossenen Ring behandelt QGIS den Offset als eine Schleife um den Ring (vorwärts oder rückwärts). Setzt man z.B. den Offset auf 150% (bzw. -10% oder -110%), so wird der Offset als 50% (bzw. 90%) der Länge des geschlossenen Rings behandelt.
  – Die Option *Markierung der Linienrichtung folgend drehen* legt fest, ob jedes Markersymbol relativ zur Linienrichtung ausgerichtet werden soll oder nicht.

    Da eine Linie oft aus einer Folge von Segmenten mit unterschiedlichen Richtungen besteht, wird die Drehung der Markierung durch einen Mittelwert über eine bestimmte Strecke entlang der Linie berechnet. Wenn Sie zum Beispiel die Eigenschaft *Mittlerer Winkel über* auf `4mm` setzen, werden die beiden Punkte entlang der Linie, die `2mm` vor und nach der Platzierung des Symbols liegen, zur Berechnung des Linienwinkels für dieses Markierungssymbol verwendet. Dies hat den Effekt, dass kleine lokale Abweichungen von der Gesamtrichtung der Linie geglättet (oder entfernt) werden, was zu einer optisch ansprechenderen Ausrichtung der Markierungsliniensymbole führt.

  - *Linienversatz*: Die Markierungssymbole können auch gegenüber dem Linienobjekt versetzt werden.

- **Gehashte Line**: Wiederholt ein Liniensegment (ein Hash) über die Länge eines Liniensymbols, wobei ein Linien-Untersymbol zur Darstellung jedes einzelnen Segments verwendet wird. Mit anderen Worten, eine Hash-Linie ist wie eine Markierungslinie, bei der die Markierungssymbole durch Segmente ersetzt werden. Als solche haben die Hash-Linien die gleichen Eigenschaften wie Markierungsliniensymbole:
  - *Hashlänge*
  - *Hashdrehung*

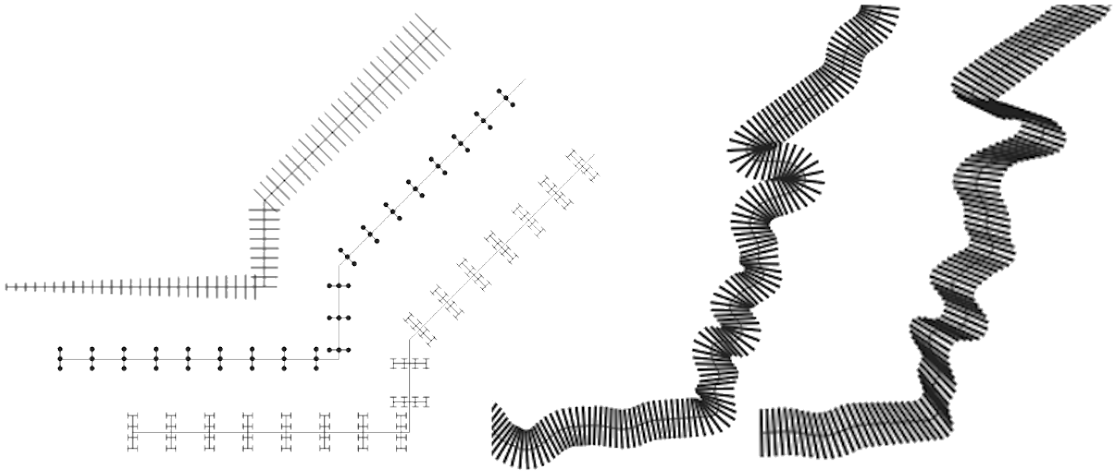


Abb. 10.14: Beispiele für gehashte Linien

- **Rasterlinie**: rendert und wiederholt ein Rasterbild entlang der Länge eines Objekts. Die *Strichbreite*, der *Versatz*, der *Verbindungsstil*, der *Endennstil* und die *Deckkraft* können angepasst werden.

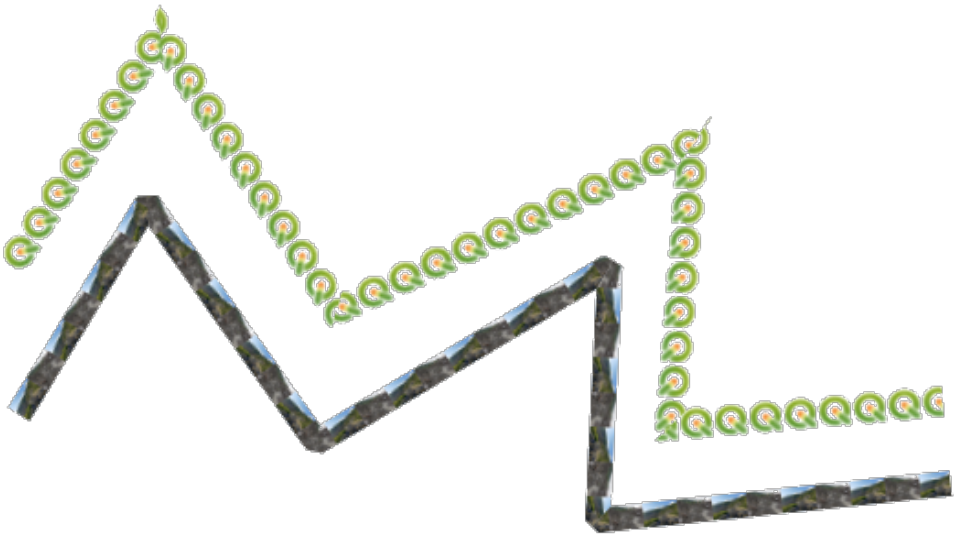


Abb. 10.15: Beispiele für Rasterlinien

- **Lineare Referenzierung**: Ermöglicht das Platzieren von Textbeschriftungen in regelmäßigen Abständen

entlang einer Linie oder an Positionen, die vorhandenen Eckpunkten entsprechen. Die Positionen können anhand kartesischer Abstände berechnet oder aus z/m-Werten interpoliert werden. Zu den Parametern gehören:

- *Maßplatzierung*: Beschriftungen können in regelmäßigen kartesischen 2D-Abständen, in regelmäßigen linear interpolierten Abständen, die anhand der Z- oder M-Werte in Geometrien berechnet werden, oder an vorhandenen Eckpunkten platziert werden. Es wird ein festes oder datendefiniertes *Intervall* angegeben.
- *Menge*: Die an jeder Position angezeigten Beschriftungen geben die zurückgelegte Strecke entlang der Linie oder den linear interpolierten Z- oder M-Wert an. Die Beschriftungen werden unter Verwendung des gesamten Spektrums der in QGIS verfügbaren Formatierungsfunktionen für *Text* und *Zahlen* gerendert.

> **Bemerkung**
>
> Bei Verwendung der entfernungsbasierten Platzierung werden die Entfernungen ausschließlich in 2D berechnet. Kartesische Berechnungen basieren auf dem ursprünglichen KBS des Layers, nicht dem KBS des Projekts.

- *Vielfaches überspringen von*: Wenn diese Option aktiviert ist, werden Beschriftungen, die ein Vielfaches dieses Wertes sind, übersprungen. Dies ermöglicht die Erstellung komplexer Referenzbeschriftungen, z. B. wenn ein Symbol zwei lineare Referenzsymbolebenen hat, von denen eine so eingestellt ist, dass sie alle 100 m in kleiner Schrift beschriftet, wobei Vielfache von 1000 übersprungen werden, und die zweite so eingestellt ist, dass sie alle 1000 m in großer, fetter Schrift beschriftet.
- *Mittlerer Winkel über*: Beschriftungen werden unter Verwendung eines Winkels gerendert, der durch Mittelung der Linienfolge berechnet wird, sodass scharfe kleine Zacken in der Linienführung nicht zu einer unschönen Drehung der Beschriftung führen.
- *Markierungssymbole anzeigen*: an referenzierten Punkten im Linienelement werden QGIS-Markierungs-Symbole angezeigt (die vollständigen Formatierungen für Markierungen stehen zur Verfügung). Dies ermöglicht z. B. die Anzeige einer Kreuzschraffur am beschrifteten Punkt für eine Linie im „Lineal-Stil“.

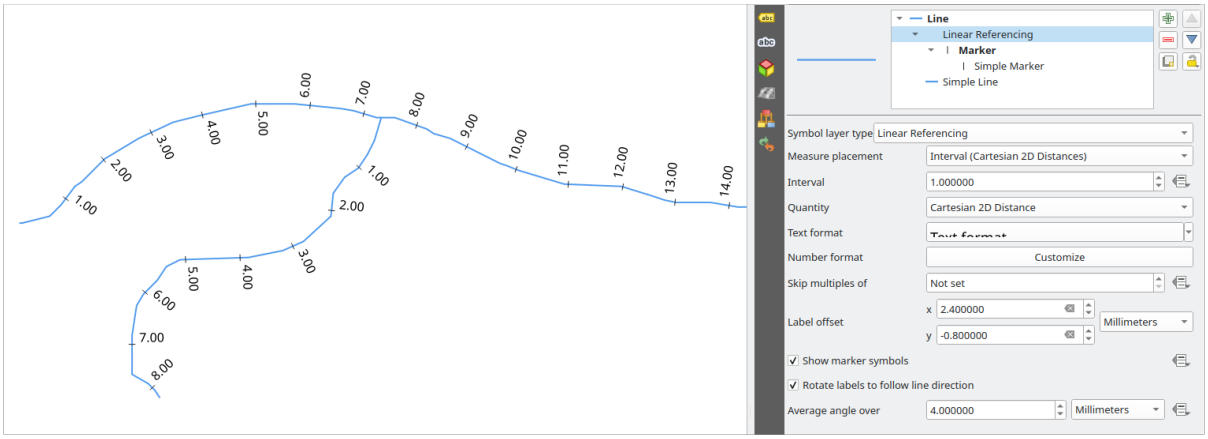


Abb. 10.16: Beschriftungen mittels linearer Referenzierung entlang einer Linie

- **Lineburst**: Rendert einen Farbverlauf entlang der Breite einer Linie. Sie können zwischen *Zweifarbig* oder *Farbverlauf* wählen und die *Strichbreite*, den *Versatz*, den *Verbindungsstil* und den *Endenstil* einstellen.

Abb. 10.17: Beispiele für lineburst-Linien

- **Gefüllte Linie**: Rendert das Innere der Linien mithilfe eines Füllsymbols, wodurch Linien mit Farbverläufen, Schraffuren usw. gefüllt werden können. Die Optionen *Strichbreite*, *Versatz*, *Verbindungsstil* und *Endstil* können angepasst werden.

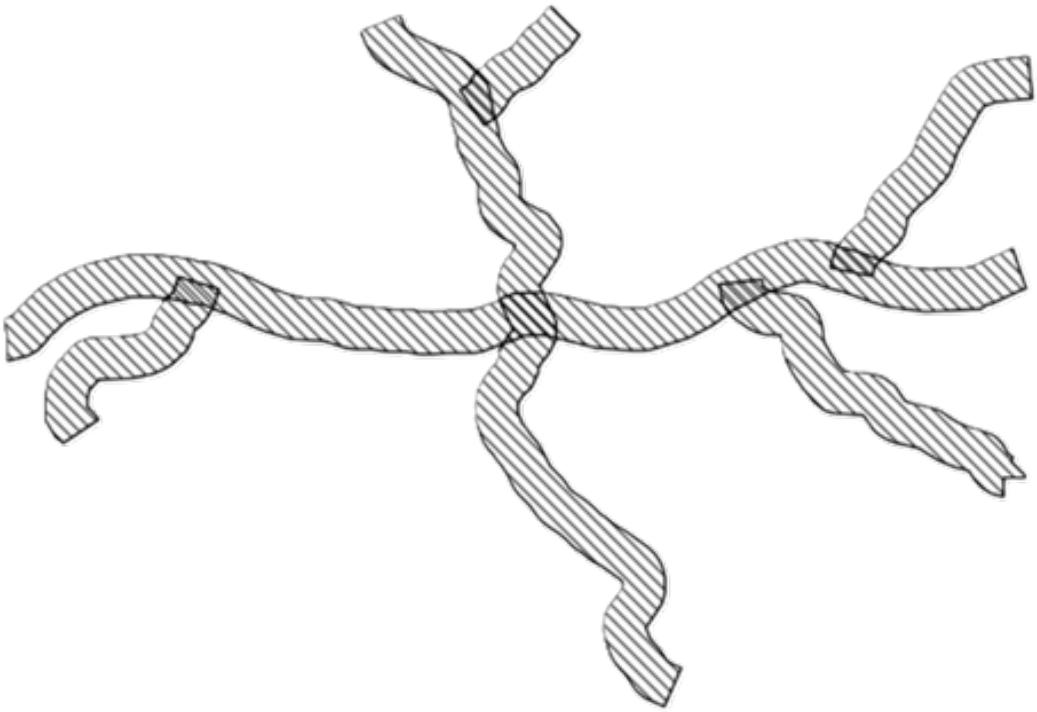


Abb. 10.18: Beispiele für gefüllte Linien

Füllungs-Symbole

Füllsymbole eignen sich für polygonale geometrische Objekte und verfügen über mehrere Layer-Typen:

Tab. 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	

- **Einfache Füllung** (Standard): füllt ein Polygon mit einer einheitlichen Farbe

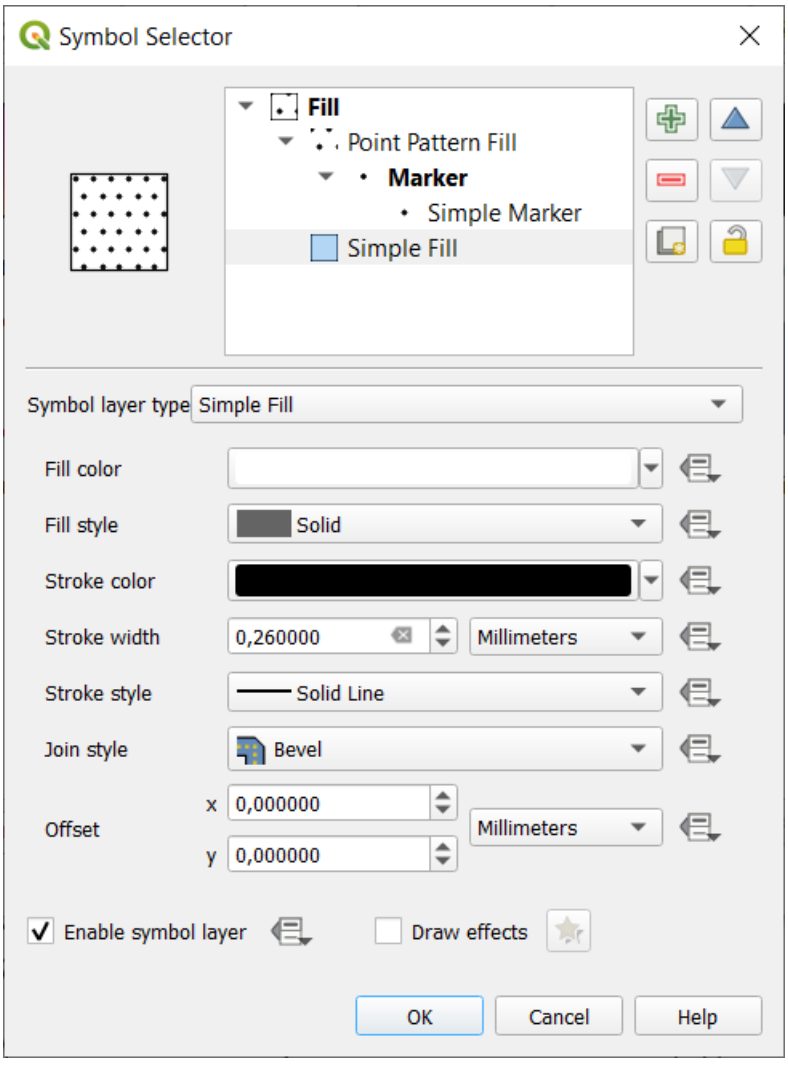


Abb. 10.19: Dialog zur Erstellung einer einfachen Füllung

- **Zentrierte Füllung**: setzt ein *Markierungs-Symbol* auf den Schwerpunkt des sichtbaren Objekts. Die Position

des Markers ist möglicherweise nicht der tatsächliche Schwerpunkt des Objekts, da die Berechnung das Polygon bzw. die Polygone berücksichtigt, die auf den in der Kartenansicht sichtbaren Bereich beschränkt sind, und Löcher ignoriert. Verwenden Sie ein *Geometrie-Generator-Symbol*, wenn Sie den exakten Flächenschwerpunkt wünschen.

Sie können:

- *Platzierung der Markierung in Polygon erzwingen*
- *Markierungen auf alle Teile mehrteiliger Objekte zeichnen* oder den Punkt nur auf dem größten Teil platzieren
- das/die Markierungssymbol(e) ganz oder teilweise anzeigen, wobei die Teile, die sich mit der aktuellen Objektgeometrie überschneiden (*Markierungen durch Polygone zuschneiden*) oder das Geometrieteil, zu dem das Symbol gehört (*Markierungen nur Rand des aktuellen Polygonteil abschneiden*)

- **Geometriegenerator** (siehe *Der Geometrie-Generator*)
- **Gradientenfüllung**: verwendet einen radialen, linearen oder konischen Farbverlauf, der entweder auf einfachen zweifarbigen Farbverläufen oder einer vordefinierten *Farbverlaufsrampe* basiert, um Polygone zu füllen. Der Farbverlauf kann gedreht und auf ein einzelnes Objekt oder auf die gesamte Kartenausdehnung angewendet werden. Auch Start- und Endpunkte können über Koordinaten oder über den Schwerpunkt (des Objekts oder der Karte) festgelegt werden. Ein datenabhängiger Offset kann definiert werden.
- **Linienmusterfüllung**: füllt das Polygon mit einem Schraffurmuster aus *Liniensymbol Layer*. Sie können einstellen:
  - *Ausrichtung*: legt fest, wie das Muster in Bezug auf das/die Objekt(e) positioniert wird:
    - *Muster an Objekt ausrichten*: Linien werden innerhalb jedes Objekts neu ausgerichtet
    - *Muster an Kartenausmaßen ausrichten*: ein Muster wird über die gesamte Kartenausdehnung gerendert, sodass die Linien über Objekte hinweg gleichbleibend ausgerichtet sind
  - *Drehung* der Linien, gegen den Uhrzeigersinn
  - *Zwischenräume*: Abstand zwischen aufeinanderfolgenden Linien
  - *Versatz* Abstand der Linien vom Rand des Objekts
  - *Ausschneiden*: erlaubt es zu steuern, wie Linien in der Füllung an die Polygonform angepasst werden sollen. Die Optionen sind:
    - *Nur beim Zeichnen zuschneiden*: Es werden Linien erzeugt, die den gesamten Begrenzungsrahmen des Objekts abdecken und dann beim Zeichnen abgeschnitten werden. Anfang und Ende der Linien sind nicht sichtbar.
    - *Linien vor dem Zeichnen zuschneiden*: Linien werden vor dem Rendern auf die genaue Form des Polygons beschnitten. Linienenden (einschließlich Kappenstile, Start-/Endmarkierungslinienobjekte, …) werden sichtbar sein und können manchmal über das Polygon hinausreichen (abhängig von den Einstellungen für das Liniensymbol).
    - *Kein Zuschneiden*: es wird überhaupt kein Zuschneiden durchgeführt - die Linien bedecken die gesamte Bounding Box des Objekts

> **Bemerkung**
>
> 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.

- **Punktmusterfüllung**: füllt das Polygon mit einem Gittermuster aus *Markersymbolen*. Sie können einstellen:
  - *Ausrichtung*: legt fest, wie das Muster in Bezug auf das/die Objekt(e) positioniert wird:

    * *Muster an Objekt ausrichten*: Linien für die Markierungen werden innerhalb jedes Objekts neu ausgerichtet
    * guilabel:*Ausrichtung des Musters an der Kartenausdehnung*: ein Muster wird über die gesamte Kartenausdehnung gerendert, sodass die Markierungen über Objekte hinweg gleichbleibend ausgerichtet sind

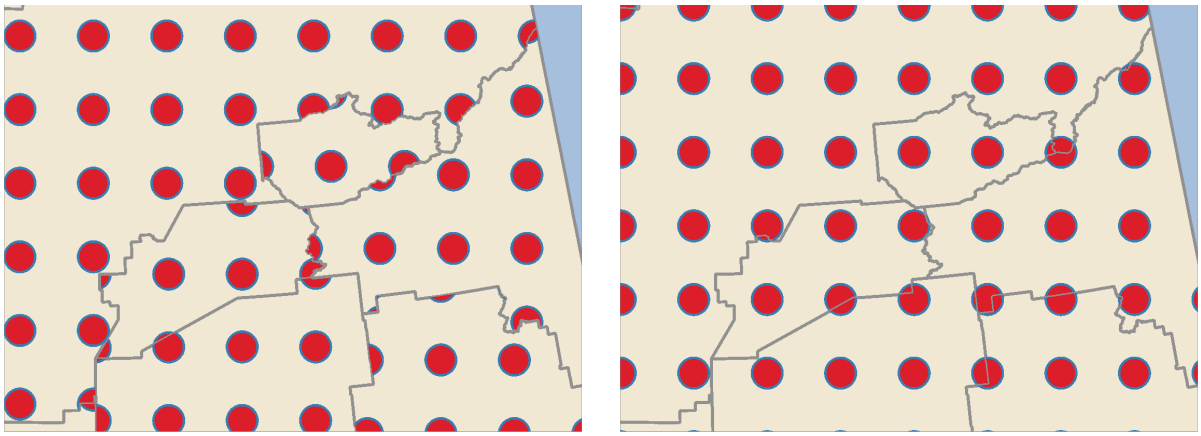


Abb. 10.20: Ausrichten von Punktmustern am Objekt (links) und an der Kartenausdehnung (rechts)

- *Entfernung*: *Horizontal* and *Vertikal* Abstände zwischen aufeinanderfolgenden Markern
- *Verlagerung*: ein *Horizontal* (bzw. *Vertikal*) Offset der Ausrichtung zwischen aufeinanderfolgenden Markern in einer Spalte (bzw. in einer Zeile)
- *Versatz*: *Horizontal* und *Vertikal* Entfernungen von der Grenze des Objekts
- *Auschneiden*: erlaubt es zu steuern, wie die Markierungen in der Füllung an die Polygonform angepasst werden sollen. Die Optionen sind:
    * *Auf Form zuschneiden*: Markierungen werden so beschnitten, dass nur die Teile innerhalb des Polygons sichtbar sind
    * *Markierung im Zentrum der Form*: nur Marker, deren Mittelpunkt innerhalb des Polygons liegt, werden gezeichnet, aber diese Marker werden nicht an der Außenseite des Polygons abgeschnitten
    * *Markierung ganz in der Form*: es werden nur Markierungen angezeigt, die vollständig innerhalb des Polygons liegen
    * *kein Zuschneiden*: jede Markierung, die sich mit dem Polygon schneidet, wird vollständig gerendert

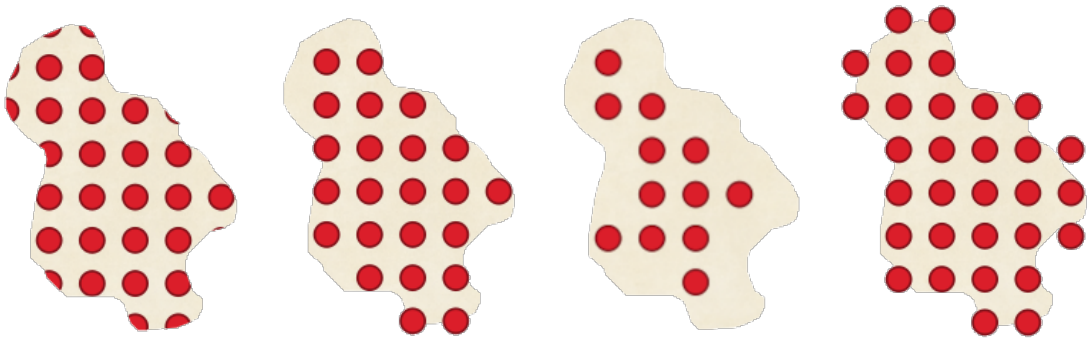


Abb. 10.21: Beschneiden von Markierungen in Füllungen - Von links nach rechts: Auf Form zuschneiden, Markierungsschwerpunkt innerhalb der Form, Markierung ganz in der Form, kein Zuschneiden

- *Drehung* des gesamten Musters, im Uhrzeigersinn

  - Im Bereich *Zufälliges Muster* eine Punktmusterfüllung mit zufällig verteilten Punkten bis zu einem einstellbaren maximalen Abstand (*Horizontal* oder *Vertikal*) erstellt werden. Sie können den maximalen Versatz in verschiednen Einheiten angeben, z. B. in Millimetern, Punkten, Karteneinheiten oder in „Prozent“ (wobei sich der Prozentwert auf die Breite oder Höhe des Musters bezieht).

    Sie können eine Ausganszufallszahl festlegen, um zu verhindern, dass die Symbolmuster zwischen den Aktualisierungen der Karte „herumspringen“. Es werden auch datenbasierte Überschreibungen unterstützt.

    Bemerkung

    Der Hauptunterschied zwischen dem Symboltyp *Zufälliges Muster* und dem Symboltyp *zufällige Markierungs-Füllung* besteht darin, dass der zufällige Versatz mit einem Punktmuster eine quasi-„regelmäßige“ Platzierung von Markern ermöglicht. Da die Punkte im Muster effektiv auf ein Raster beschränkt sind, ermöglicht dies die Erstellung von halb-zufälligen Füllungen, die keine leeren Bereiche oder überlappende Marker aufweisen. (Im Gegensatz zur zufälligen Markerfüllung, bei der die Punkte immer völlig willkürlich platziert werden… was manchmal zu visuellen Punkthäufungen oder unerwünschten leeren Bereichen führt).

- **Zufällige Markierungsfüllung**: füllt das Polygon mit einem :ref:`Markierungssymbol <vector_marker_symbols> `, das an zufälligen Stellen innerhalb der Polygongrenze platziert wird. Sie können einstellen:
  - *Zählmethode*: ob die Anzahl der darzustellenden Markierungssymbole als absolute Anzahl oder dichtebasiert betrachtet wird
  - *Punktanzahl*: die Anzahl der darzustellenden Markierungssymbole,
  - eine optionale Zufallszahl *Saat*, um eine einheitliche Platzierung zu gewährleisten
  - *Flächendichte*: stellt im Falle der dichtebasierten Zählmethode sicher, dass die Fülldichte der Marker auf verschiedenen Skalierungs-/Zoomstufen der Marker gleich bleibt, wenn die Karten aktualisiert werden (ermöglicht auch eine zufällige Platzierung, um mit dem QGIS-Server und dem kachelbasierten Rendering zusammenzuarbeiten)
  - *Markierungen durch Polygon zuschneiden*: legt fest, ob Markierungen, die in der Nähe der Kanten von Polygonen gerendert werden, an der Polygongrenze abgeschnitten werden sollen oder nicht
- **Rasterbildfüllung**: Füllt das Polygon mit Kacheln aus einem Rasterbild (`PNG` `JPG`, `BMP` …). Das Bild kann eine Datei auf der Festplatte, eine Remote-URL oder eine als Zeichenfolge codierte eingebettete Datei sein (*weitere Details*). Zu den Optionen gehören Deckkraft, Größe, Koordinatenmodus (Objekt oder Ansichtsfenster), Drehung und Versatz. Unter der Option „Größe“ können Sie die Breite und Höhe des Füllmusters unabhängig voneinander anpassen, wodurch gestreckte Rasterfüllungen in horizontaler oder vertikaler Richtung möglich sind. Die Bildbreite und -höhe kann mit einer der *gängigen Einheiten* oder als Prozentsatz der Originalgröße festgelegt werden
- **SVG-Füllung**: füllt das Polygon unter Verwendung einer *SVG-Markierung* in einer bestimmbaren Größe (*Texturbreite*).
- **Shapeburst-Füllung**: puffert eine Gradientenfüllung, bei der ein Gradient vom Rand eines Polygons zur Mitte des Polygons hin gezeichnet wird. Zu den konfigurierbaren Parametern gehören der Abstand vom Rand zur Schattierung, die Verwendung von Farbrampen oder einfachen, zweifarbigen Farbverläufen, die Verwischstärke der Füllung und ein Versatz.
- **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.

Bemerkung

Wenn der Geometrietyp Polygon ist, können Sie das automatische Beschneiden von Linien/Polygonen auf der Kartenansicht deaktivieren. Dies kann hilfreich sein, da in einigen Fällen dieses Beschneiden zu einer ungünstigen Symbologie führt (z. B. bei Schwerpunktfüllungen, bei denen der Schwerpunkt immer der Schwerpunkt des eigentlichen Objekts sein muss).

Parametrisierbare SVG

Sie haben die Möglichkeit, die Farben eines *SVG-Markers* zu ändern. Sie müssen die Platzhalter `param(fill)` für die Farbe der Füllung, `param(fill-opacity)` für die Deckkraft der Füllung, `param(outline)` und `param(outline-opacity)` für die Farbe bzw. Deckkraft der Kontur und `param(outline-width)` für die Breite der Kontur hinzufügen. Diese Platzhalter können optional von einem Standardwert gefolgt werden, z.B:

```
<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>
```

Allgemeiner gesagt, kann SVG mit `param(param_name)` frei parametrisiert werden. Dieser Parameter kann entweder als Attributwert oder als Knotentext verwendet werden:

```
<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>
```

Die Parameter können dann als Ausdrücke in der Tabelle *Dynamische SVG-Parameter* definiert werden.

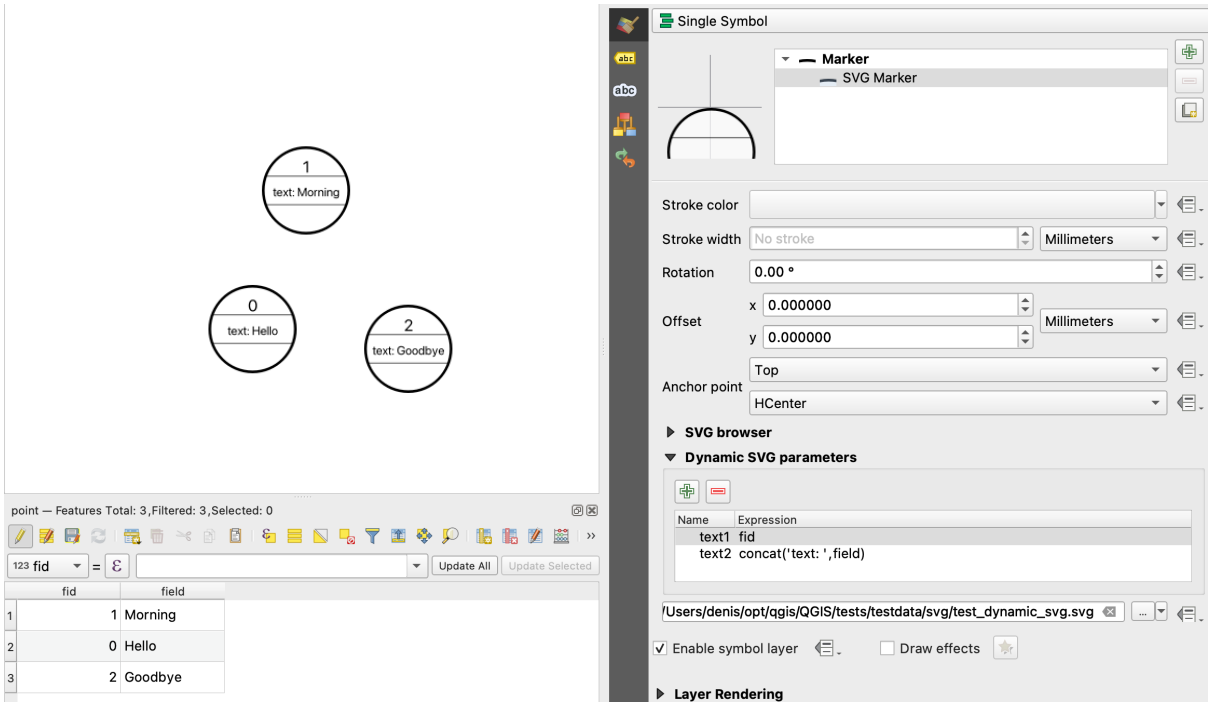


Abb. 10.22: Tabelle der dynamischen SVG-Parameter

Bemerkung

QGIS sucht nach einem vollständigen SVG-Knoten. Wenn Ihr Parameter also innerhalb eines komplexeren Knotens liegt, müssen Sie den kompletten Knoten mit dem Ausdruck injizieren. Zum Beispiel wird `transform=„rotate(param(angle)“` nicht funktionieren. Stattdessen müssen Sie `transform=„param(rotation)“` verwenden und der Parameter `rotation` wird mit dem Ausdruck `'rotate(' || coalesce(my_field, 0) || ')'` definiert.

Der Geometrie-Generator

Der Symbol Layer *Geometriegenerator* ist für alle Symboltypen verfügbar und ermöglicht es, mit *Ausdrucksyntax* während des Rendering-Prozesses eine Geometrie on the fly zu erzeugen. Die resultierende Geometrie muss nicht mit dem ursprünglichen *Geometrietyp* übereinstimmen und Sie können mehrere unterschiedlich modifizierte Symbol Layer übereinander legen.

Eine *Einheit* kann gesetzt werden: Wenn das Geometrie-Generator-Symbol nicht auf einen Layer angewendet wird (z.B. auf ein Layout-Element), erlaubt dies eine bessere Kontrolle über die generierte Ausgabe.

Einige Beispiele:

- Symbol als Schwerpunkt eines Objekts rendern

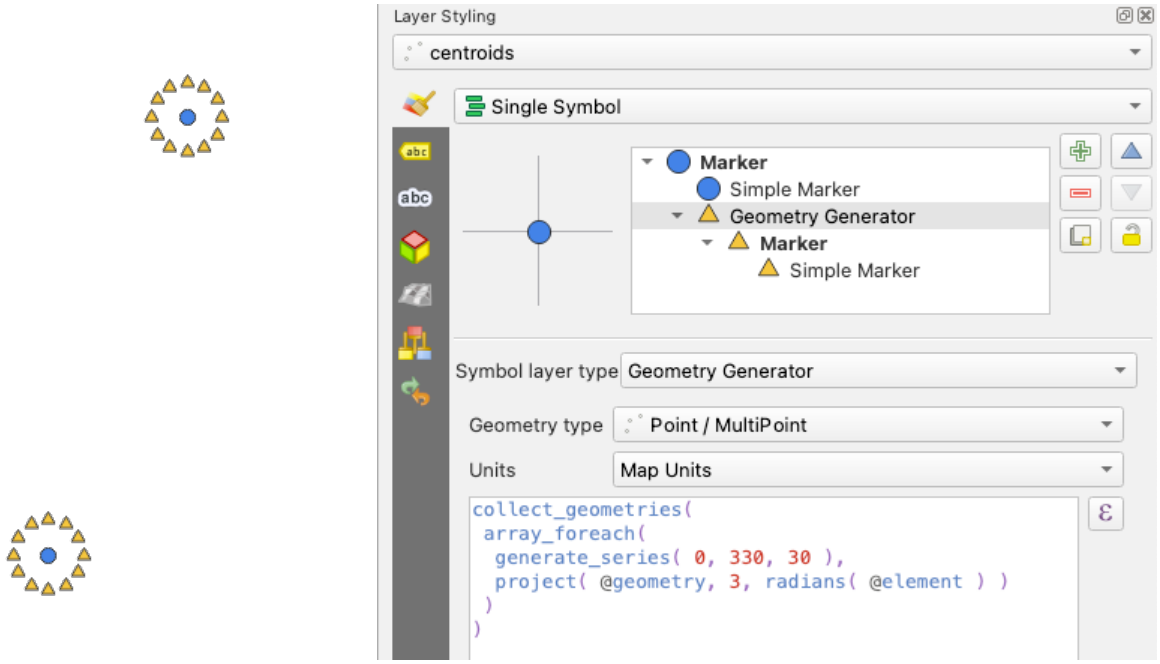
```
centroid( $geometry )
```

- Visuelle Überlappung von Objekten innerhalb eines Abstands von 100 Karteneinheiten zu einem Punkt-Objekt, d. h. Erzeugung eines Puffers von 100m um den Punkt

```
buffer( $geometry, 100 )
```

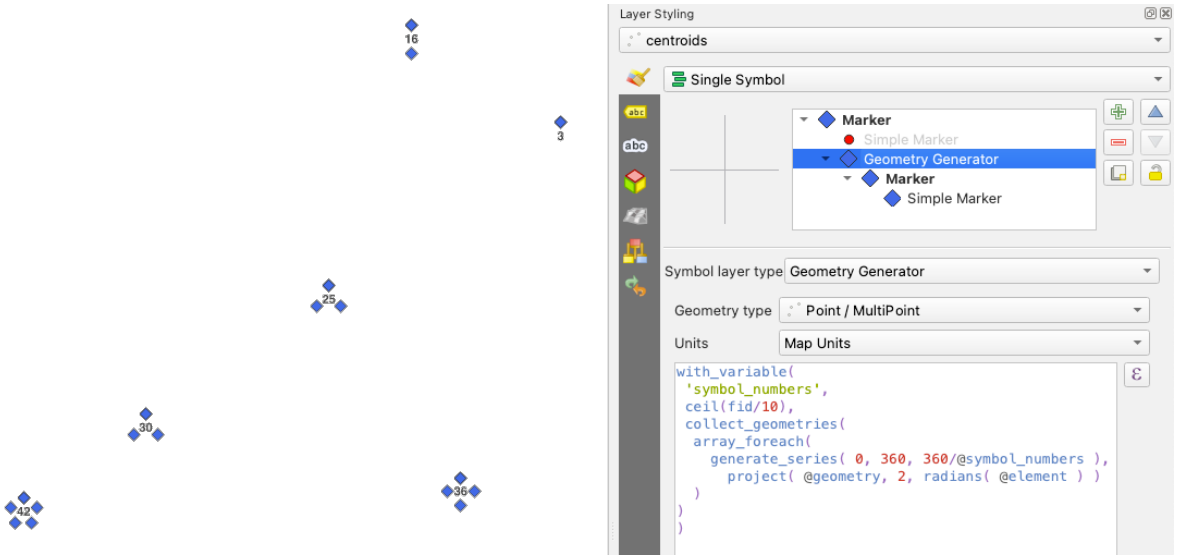
- Erzeugt einen Kreis von Punkt-Objekten, die das zentralen Punkt-Objekt umgeben, wenn es als MultiPoint-Geometriegenerator verwendet wird

```
collect_geometries(
  array_foreach(
    generate_series( 0, 330, 30 ),
    project( $geometry, 3, radians( @element ) )
  )
)
```



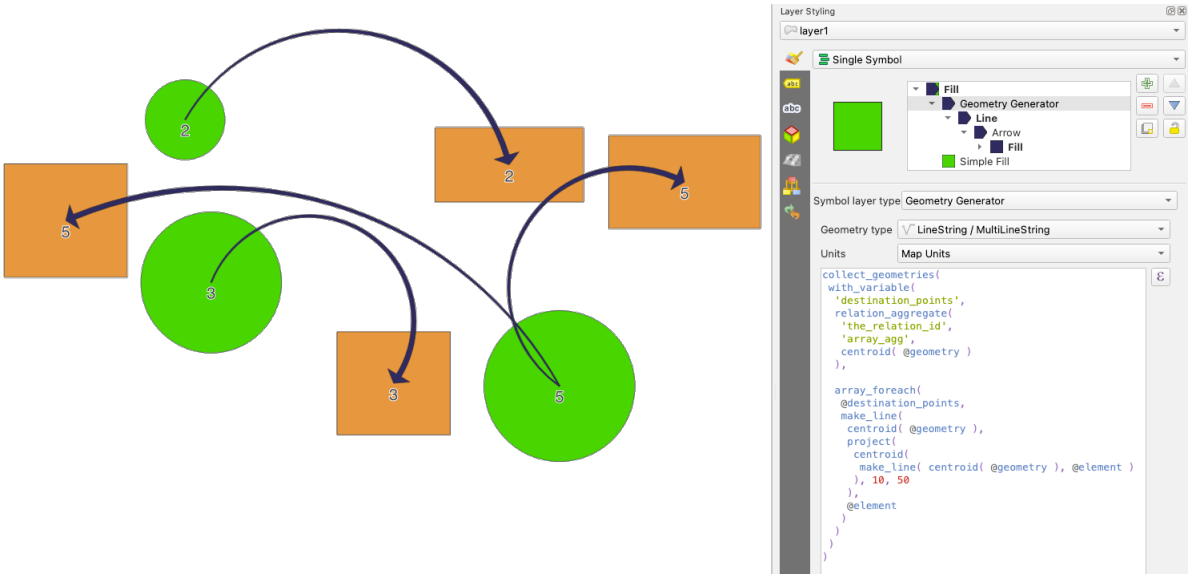
- Erzeugt einen Kreis von Punkt-Objekten, die das zentralen Punkt-Objekt umgeben. Die Anzahl der Punkte variiert je nach Feld.

```
with_variable(
  'symbol_numbers',
  ceil(fid/10),
  collect_geometries(
    array_foreach(
      generate_series( 0, 360, 360/@symbol_numbers ),
      project( $geometry, 2, radians( @element ) )
    )
  )
)
```



- Erzeugt eine gekrümmte Pfeillinie, die Objekte zweier Layer auf der Grundlage ihrer *Beziehung* verbindet

```
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
      )
    )
  )
)
```



Die Vektorfeld-Markierung

Der Vektorfeldmarker dient zur Darstellung von Vektorfelddaten wie Erdverformung, Gezeitenströmungen und dergleichen. Er zeigt die Vektoren als Linien (vorzugsweise Pfeile) an, die entsprechend den ausgewählten Attributen der Datenpunkte skaliert und ausgerichtet sind. Er kann nur zur Darstellung von Punktdaten verwendet werden; Linien- und Polygon-Layer werden von dieser Symbologie nicht gezeichnet.

Das Vektorfeld wird durch die Attribute in den Daten definiert, die das Feld darstellen können durch:

- **Kartesische** Komponenten (`x` und `y` Komponenten des Feldes)
- oder **polare** Koordinaten: In diesem Fall definieren die Attribute `Länge` und `Winkel`. Der Winkel kann entweder im Uhrzeigersinn von Norden oder gegen den Uhrzeigersinn von Osten aus gemessen werden und kann entweder in Grad oder Radiant angegeben werden.
- oder als **Nur-Höhe**-Daten, die einen vertikalen Pfeil anzeigen, der anhand eines Attributs der Daten skaliert wird. Dies ist z. B. für die Anzeige der vertikalen Komponente der Verformung geeignet.

Die Größe des Feldes kann auf eine für die Betrachtung des Feldes geeignete Größe herauf- oder herabskaliert werden.

Animierte Markierung

Mit dem Symboltyp „Animierte Markierung“ können Sie eine `.GIF`, `.WebP`, `.MNG` usw. Animationsdatei verwenden, um Punkte auf Ihrer Karte darzustellen. Sie können angeben:

- *Datei*-Pfad,
- *Bildrate*: Anzahl der Schritte, die pro Sekunde angezeigt werden, was angibt, wie schnell die Animation abgespielt wird,
- *Größe* in verschiedenen *Einheiten*,
- *Deckkraft*,
- *Drehung*,
- *Versatz* in *x* und *y* Richtung von der Markerposition aus,
- *Ankerpunkt*

Es gibt zwei Möglichkeiten, mit animierten Symbolen zu arbeiten:

- **Wenn Ihre Karte nicht als Animation** konfiguriert ist (d.h. es handelt sich um ein Standard-QGIS-Projekt ohne Animationen), wird der Rahmen für die animierten Marker ausschließlich durch den aktuellen Zeitstempel bestimmt.
- **Wenn Ihre Karte** :ref:`als Animation <maptimecontrol>`konfiguriert ist, werden die animierten Markierungen mit der Zeitachse der Animation synchronisiert. Das bedeutet, dass die animierten Marker pausieren, wenn die Animation pausiert, mit der Animation fortschreiten usw. Die Karte wird außerdem entsprechend der für die zeitliche Animation festgelegten Bildrate neu gezeichnet. Dieser Modus wird auch beim Exportieren einer Animation mit dem zeitlichen Controller angewendet.

10.3 Einrichten einer Beschriftung

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:

- *Textformat*: definiert das Aussehen des Textes, einschließlich *Schriftart, Größe, Farben*, *Schatten*, *Hintergrund*, *Puffer*, …

  Sie können verwendet werden, um Texte über der Karte darzustellen (Layout/Kartentitel, Dekorationen, Maßstabsbalken, …), normalerweise durch das Schriftauswahl-Werkzeug.

  So erstellen Sie ein *Textformat*:

  1. Öffnen Sie das Dialogfeld *Stilverwaltung*.
  2. Aktivieren Sie den Reiter *Textformat*

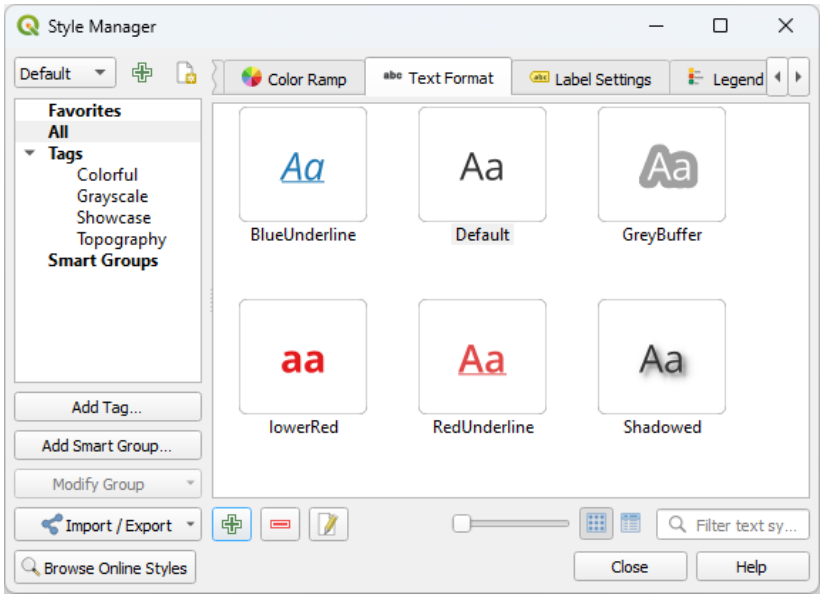


Abb. 10.23: Textformat-Reiter im Dialogfeld Stilverwaltung

  3. Drücken Sie die Schaltfläche Element hinzufügen. Der Dialog *Neues Textformat* öffnet sich und Sie können die gewünschten *Konfigurationen* vornehmen. Auch hier sind die Eigenschaften des Texformats *datendefinierbar*.

- *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* …).

  Sie werden verwendet, um die automatische Beschriftung für Vektor- und Netz-Layer über die Registerkarte *Beschriftungen* des Dialogs *Layereigenschaften* oder des Bedienfelds *Layergestaltung* oder über die Schaltfläche Layerbeschriftungseinstellungen der *Beschriftungswerkzeugleiste* zu konfigurieren.

  So erstellen Sie eine *Beschriftungseinstellung*:

  1. Öffnen Sie das Dialogfeld *Stilverwaltung*.
  2. Aktivieren Sie den Reiter *Beschriftungseinstellungen*.

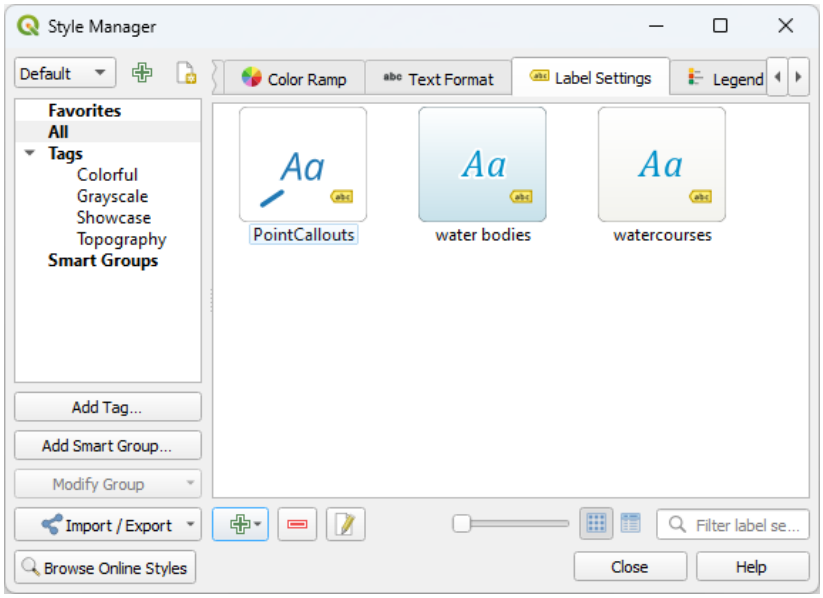


Abb. 10.24: Beschriftungseinstellungen-Reiter im Dialogfeld Stilverwaltung

3. Drücken Sie die Schaltfläche Element hinzufügen und wählen Sie den Eintrag, der dem Geometrietyp der Objekte entspricht, die Sie beschriften möchten.

Das Dialogfeld *Einstellungen für Beschriftungen* wird mit den im folgenden Kapitel gezeigten Optionen geöffnet. Wie üblich sind diese Eigenschaften *datendefinierbar*.

10.3.1 Formatierung der Text-Elemente

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:

Tab. 10.4: Label properties by data type

Reiter	Textformat	Beschriftungseinstellungen	Vector or mesh layer	raster layer
Text	✓	✓	✓	✓
Formatierung	✓	✓	✓	✓
Puffer	✓	✓	✓	✓
Maske		✓	✓	
Hintergrund	✓	✓	✓	✓
Schatten	✓	✓	✓	✓
Führungslinien		✓	✓	
Platzierung		✓	✓	✓
Darstellung		✓	✓	✓

Achtung

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“.

Text-Reiter

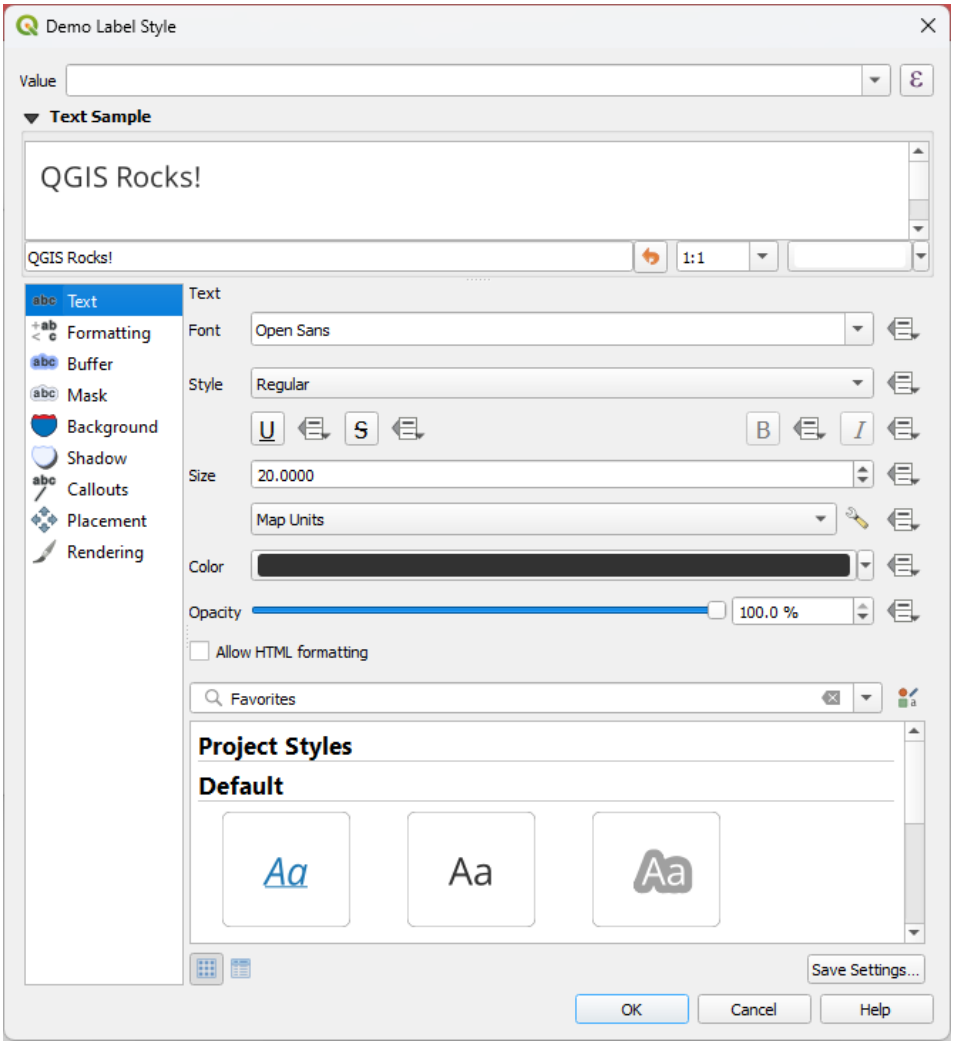


Abb. 10.25: Beschriftungseinstellungen - Text-Reiter

Im Reiter abc *Text* haben Sie Einstellungsmöglichkeiten für die folgenden Parameter der Schrift:

- :guilabel:`Schriftart `: alle auf Ihrem Rechner verfügbaren Schriftarten
- *Stil*: neben den üblichen Einstellungen für Schriftarten (fett, kursiv, …) können Sie einstellen, ob der Text unterstrichen oder durchgestrichen sein soll
- *Größe* in verschiedenen *Einheiten*
- *Farbe*
- *Deckkraft*
- *HTML-Formatierung erlauben* ermöglicht die Verwendung eines Teils von HTML-Tags und CSS-Regeln, um die Beschriftung anzupassen (s.u.)

Am unteren Rand des Reiters wird in einem Bereich eine filterbare Liste der kompatiblen Elemente an, die in Ihrer *Stilverwaltung* gespeichert sind. So können Sie auf einfache Weise das aktuelle Textformat oder die Beschriftung auf der Grundlage einer bereits vorhandenen Einstellung konfigurieren und auch ein neues Element in der Stildatenbank speichern: Drücken Sie die Schaltfläche *Format speichern…* oder *Einstellungen speichern…* und geben Sie einen Namen und ein oder mehrere Tags an.

> **Bemerkung**
>
> Wenn Sie eine *Beschriftungseinstellung* konfigurieren, sind in diesem Widget auch Textformat-Elemente verfügbar. Wählen Sie einen aus, um die aktuellen *Textuellen Eigenschaften* der Beschriftung schnell zu überschreiben. Ebenso können Sie von dort aus ein Textformat erstellen/überschreiben.

HTML-Formatierung erlauben

Wenn *HTML-Formatierung erlauben* aktiviert ist, müssen Sie den HTML-Code im Feld *Wert* eintragen. Verwenden Sie für Einrückungen Leerzeichen anstelle von Tabulatoren. Der Ausdruck wird geparst und alle unterstützten HTML-Tags überschreiben die entsprechenden Einstellungen in den Beschriftungs-Eigenschaften. Da die Liste der von QGIS unterstützten HTML-Tags und CSS-Eigenschaften zu umfangreich ist, um sie hier aufzuführen, finden Sie unten nur eine kleine Auswahl. Für eine vollständige Auflistung schauen Sie bitte in der Dokumentation der zugrunde liegenden Qt-Bibliothek nach.

Beispiele für unterstützte HTML-Tags:

- Textformatierungen wie Kursivschrift oder Fettdruck, z. B.:

```
<i>QGIS</i> <b>rocks!</b>
```

- Hoch- und Tiefstellung, wobei der Text vertikal $^{\text{super}}$ oder $_{\text{sub}}$ ausgerichtet und automatisch auf 2/3 der Schriftgröße des übergeordneten Elements skaliert wird. Sie können auch eine feste Schriftgröße für die Hoch- und Tiefstellung festlegen, indem Sie CSS-Regeln einfügen, z. B.:

```
<sup style="font-size:33pt">my superscript text</sup>
```

- Horizontale Textausrichtung, entweder mit dem HTML-Attribut „align=“xxx”” oder dem Tag „center”. Aus Gründen der HTML5-Kompatibilität sollte vorzugsweise die CSS-Eigenschaft „text-align” verwendet werden.

> **Achtung**
>
> Die horizontale Ausrichtung kann nicht für gekrümmte Beschriftungen verwendet werden.

- Strukturierung eines Textblocks mithilfe von Überschriften-Tags (wie `h1`, `h2`, …) oder Absätzen (mit `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>
```

- Einfügen von Bildern: Jedes von QGIS lesbare Bildformat kann im HTML-Beschriftungs-Inhalt verwendet werden. Es kann über lokale Dateipfade, HTTP-Links oder Base64-eingebettete Inhalte unter Verwendung des Attributs „src=„xxx““ bereitgestellt werden. Die Bildgröße kann über die Attribute `width="##"` und `height="##"` in der Einheit „Punkte“ angegeben werden. Wenn keine Breite oder Höhe angegeben ist, wird diese automatisch aus der Originalbildgröße berechnet. Bilder werden nur inline platziert, nicht als schwebende Bilder und nicht auf gekrümmten Textbeschriftungen.

```
<img src="qgis.png" width=40 height=60>
```

Beispiele für unterstützte CSS-Eigenschaften:

- Schriftart-Eigenschaften (`color`, `font-family`, `font-size`, `font-weight`, `font-style`, `word-spacing`). Beachten Sie, dass `word-spacing` immer Punkte als Einheit verwendet.
- Text decorations such as underline, overline and line-through (`text-decoration`)

- Textausrichtung (`vertical-align`, `text-align`). Die horizontale Ausrichtung kann nicht für gekrümmte Etiketten verwendet werden.
- 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.

> **Achtung**
>
> 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>
```

CSS-Eigenschaften können mit dem Attribut `style` für HTML-Tags festgelegt werden. Der HTML-Tag `span` wendet selbst keine Formatierung auf Text an und ist ideal, wenn Sie nur CSS-Stile anwenden möchten. Ein CSS-Eigenschaftsname und sein Wert werden durch einen Doppelpunkt (`:`) getrennt. Mehrere CSS-Eigenschaften werden durch ein Semikolon (`;`) getrennt, z. B.:

```
<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>
```

Nachfolgend ein Beispiel für einen HTML-basierten Ausdruck und die Darstellung (unterschiedliche Farben und Unterstreichungen für dieselbe Beschriftung):

```
format(
  '<span style="color:blue">%1</span> ( <span style="color:red"><u>%2 ft</u></span>
↪ )',
  title( lower( "Name" ) ),
  round($length)
)
```

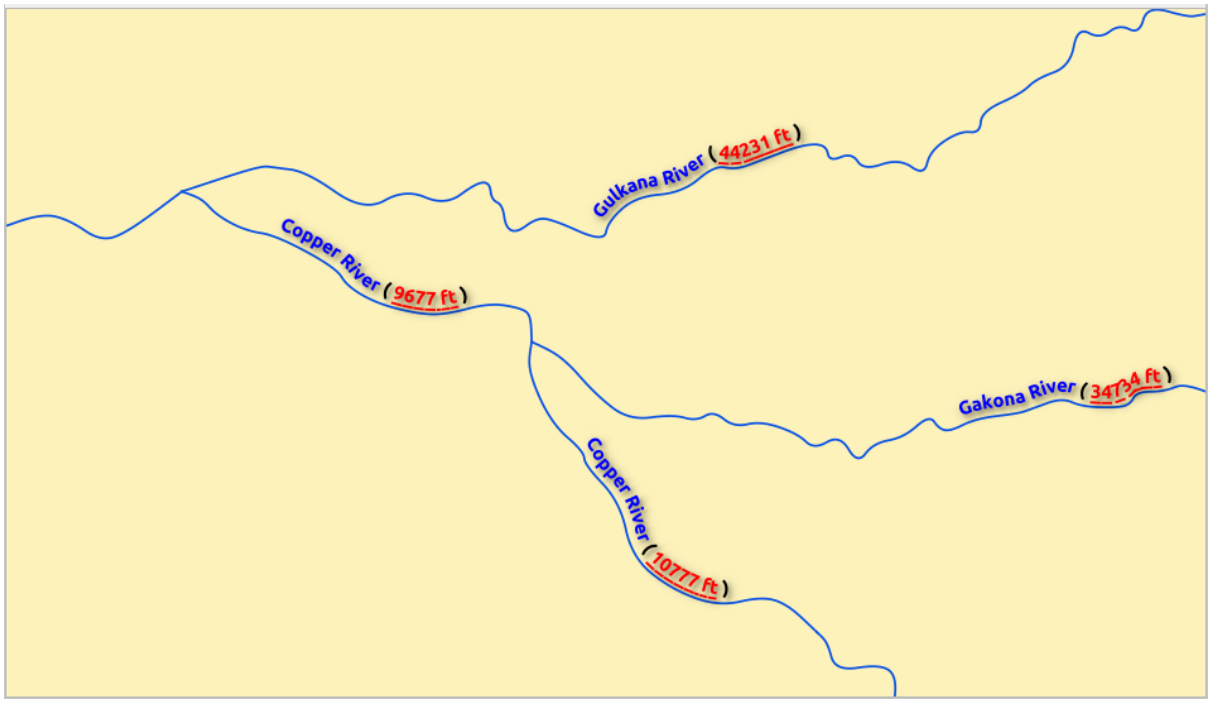


Abb. 10.26: Beschriftung mit aktivierter HTML-Formatierung

Formatierungs-Reiter

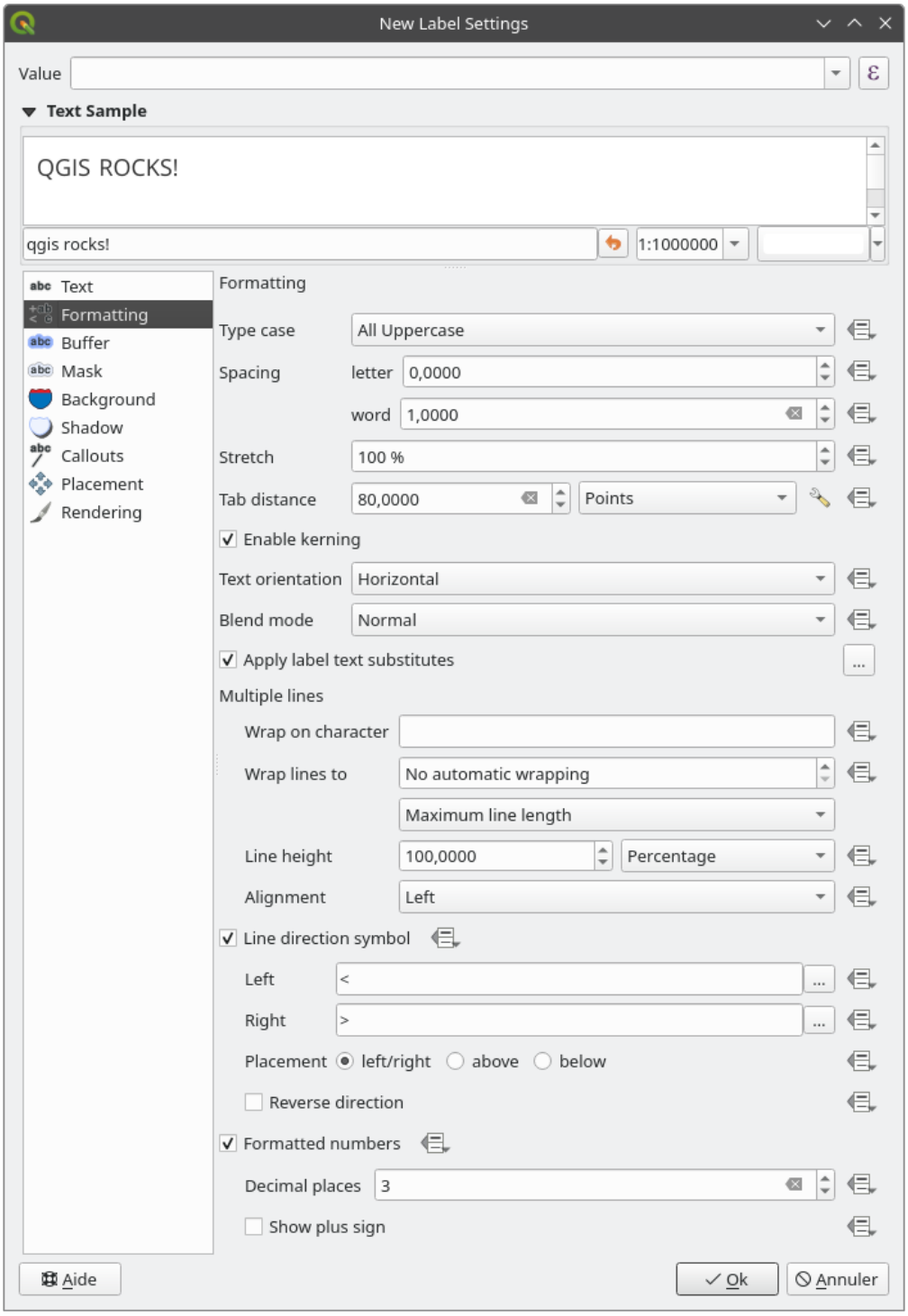


Abb. 10.27: Beschriftungseinstellungen - Formatierungs-Reiter

Im Reiter *Formatierung* können Sie:

- Verwenden Sie die Option *Groß-/Kleinschreibung*, um den Stil der Großschreibung des Textes zu ändern. Sie haben die Möglichkeit, den Text wie folgt darzustellen:
  - *Keine Änderung*

  - *Nur Großbuchstaben*
  - *Nur Kleinbuchstaben*
  - *Small Caps*: renders lowercase characters as small caps.
  - *All Small Caps*: renders all characters as small caps regardless of their original case.
  - *Titelform*: ändert den ersten Buchstaben eines jeden Wortes in Großbuchstaben und wandelt die anderen Buchstaben in Kleinbuchstaben um, wenn der Originaltext eine einzige Groß-/Kleinschreibung verwendet. Bei gemischter Groß- und Kleinschreibung im Text bleiben die anderen Buchstaben unangetastet.
  - *Großschreibung für ersten Buchstaben erzwingen*: ändert den ersten Buchstaben eines jeden Wortes in Großbuchstaben und lässt die anderen Buchstaben im Text unberührt.
- Ändern Sie unter *Zwischenräume* den Abstand zwischen den Wörtern und zwischen einzelnen Buchstaben.
- Mit *Tababstand* können Sie den Platzhalterabstand für `Tab`-Zeichen in der Beschriftung in jeder unterstützten Einheit anpassen. Dies ermöglicht beispielsweise die korrekte Trennung oder vertikale Ausrichtung von Elementen eines Layout-Legendentextes.

  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.
- *Strecken* : erlaubt es, den Text horizontal um einen Faktor zu strecken oder zu stauchen. Praktisch für die Anpassung der Breite von Schriftarten, um mehr Text in Beschriftungen unterzubringen.
- für die Schriftart *Unterschneidung einschalten*
- Setzen Sie die *Textrichtung*, die *Horizontal* oder *Vertikal* sein kann. Es kann auch *Drehungssbasiert* sein, wenn eine Beschriftung gesetzt wird (z.B. um Linienobjekte im Platzierungsmodus *parallel* richtig zu beschriften).
- Verwenden Sie die Option *Mischmodi*, um festzulegen, wie sich Ihre Beschriftungen mit den darunter liegenden Objekten mischen sollen (weitere Einzelheiten unter blend-modus).
- Mit der Option *Beschriftungstextersatz anwenden* können Sie eine Liste von Texten angeben, die anstelle von Texten in Beschriftungen von Objekten verwendet werden sollen (z. B. Abkürzungen von Straßentypen). Die Ersatztexte werden bei der Anzeige der Beschriftungen auf der Karte verwendet. Benutzer können auch Listen mit Ersatztexten exportieren und importieren, um die Wiederverwendung und gemeinsame Nutzung zu erleichtern.
- Konfigurieren Sie *Mehrzeilig*:
  - Setzen Sie ein Zeichen, das einen Zeilenumbruch im Text erzwingt, mit der Option *Bei Zeichen umbrechen*.
  - Legen Sie mit der Option *Zeilen umbrechen zu* eine ideale Zeilengröße für den automatischen Zeilenumbruch fest. Die Größe kann entweder die *Maximale Zeilenlänge* oder die *Minimale Zeilenlänge* darstellen.
  - Legen Sie die *Linienhöhe* fest: Die Werte können in *Millimetern*, *Punkten*, *Pixeln*, *Prozent* oder *Zoll* angegeben werden. Wenn die Zeilenhöhe in Prozent angegeben ist, entspricht sie dem prozentualen Anteil des Standardzeilenabstands der jeweiligen Schriftfamilie. In der Regel das 1,2- bis 1,5-fache der Textgröße.
  - Formatieren Sie die *Ausrichtung*: typische Werte sind *Links*, *Rechts*, *Blocksatz* und *Zentriert*.

    Bei der Einstellung der Eigenschaften von Punktbeschriftungen kann die Textausrichtung auch *Beschriftungsplatzierung folgen* sein. In diesem Fall hängt die Ausrichtung von der endgültigen Platzierung der Beschriftung relativ zum Punkt ab. Wird die Beschriftung z. B. links vom Punkt platziert, so wird sie rechts ausgerichtet, wird sie rechts platziert, so wird sie links ausgerichtet.

> **Bemerkung**
>
> Die Formatierung *Mehrzeilig* wird von der kurvenbasierten *Platzierung* noch nicht unterstützt. Die Optionen werden dann deaktiviert.

- Für Linienbeschriftungen können Sie *Linienrichtungssymbol* einfügen, um die Linienrichtung zu bestimmen, mit Symbolen, die *Links* oder *Rechts* anzeigen. Sie funktionieren besonders gut, wenn sie mit den Platzierungsoptionen *Kurve* oder *Parallel* aus dem Reiter *Platzierung* verwendet werden. Es gibt Optionen, um die Position des Symbols zu setzen, und um *Rückwärts* abzuschalten.
- Verwenden Sie die Option *Zahlenformatierung', um numerische Texte zu formatieren. Sie können die Anzahl der :guilabel:`Dezimalstellen* festlegen. Standardmäßig werden 3 Nachkommastellen verwendet. Verwenden Sie das Kontrollkästchen *Plus-Vorzeichen anzeigen*, wenn Sie das Plus-Zeichen für positive Zahlen anzeigen möchten.

Puffer-Reiter

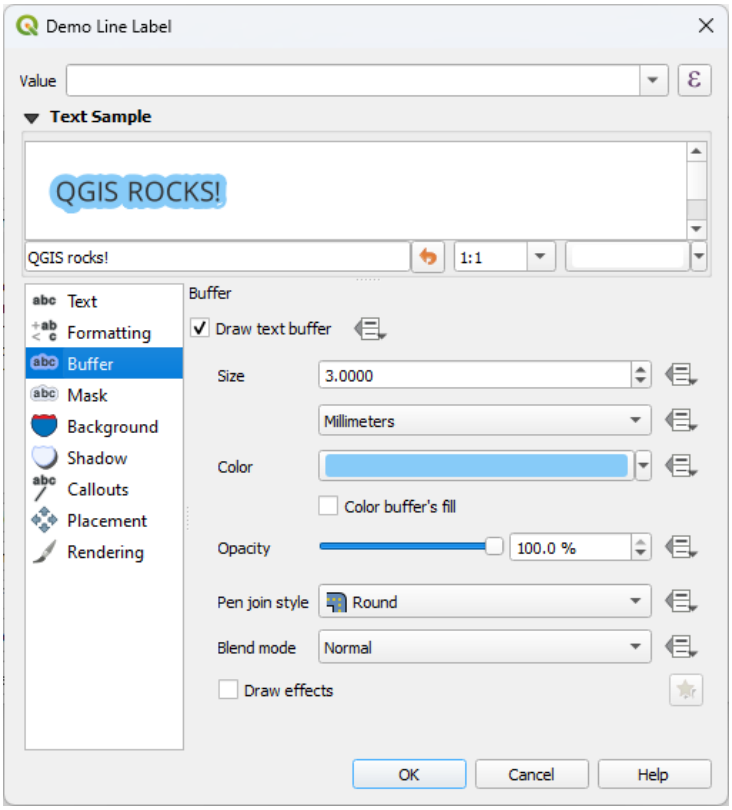


Abb. 10.28: Beschriftungseinstellungen - Puffer-Reiter

Um einen Puffer um die Beschriftung zu erstellen, aktivieren Sie das Kontrollkästchen *Textpuffer zeichnen* im Reiter *Puffer*. Sie können dann:

- Setzt die *Größe* des Puffers in einer bestimmten *Einheit*
- Wählen Sie die *Farbe* für den Puffer
- *Pufferfüllung einfärben*: Der Puffer dehnt sich vom Umriss der Beschriftung aus, so dass, wenn die Option aktiviert ist, das Innere der Beschriftung gefüllt wird. Dies kann bei der Verwendung von teilweise transparenten Beschriftungen oder bei nicht-normalen Überblendungsmodi relevant sein, die es ermöglichen, hinter den Text der Beschriftung zu sehen. Wenn Sie die Option deaktivieren (und dabei völlig transparente Beschriftungen verwenden), können Sie Beschriftungen mit Umriss erstellen.

- Legen Sie die *Deckkraft* des Puffers fest
- Anwenden eines *Stiftübergangsstils*: dieser kann *Rund*, *Eckig* oder *Abgefast* sein
- Verwenden Sie die Option *Mischmodi*, um festzulegen, wie sich der Puffer Ihrer Beschriftung mit den darunter liegenden Kartenkomponenten vermischt (weitere Details unter blend-modus).
- Markieren Sie *Zeicheneffekte*, um erweiterte *Effekte* zur Verbesserung der Lesbarkeit von Text hinzuzufügen, z.B. durch äußeres Leuchten und Verwischen.

Hintergrund-Reiter

Im Reiter *Hintergrund* können Sie eine Form konfigurieren, die hinter jeder Beschriftung erscheint. Um einen Hintergrund hinzuzufügen, aktivieren Sie das Kontrollkästchen *Hintergrund zeichnen* und wählen Sie den Typ *Form*. Das kann sein:

- eine regelmäßige Form wie *Rechteck*, *Quadrat*, *Kreis* oder *Ellipse* mit allen Eigenschaften eines *Füllungssymbols*
- ein *SVG*-Symbol aus einer Datei, einer URL oder eingebettet in die Projekt- oder Stildatenbank (*mehr Details*)
- oder ein *Markierungssymbol*, das Sie erstellen oder aus der *Symbolbibliothek* auswählen können.

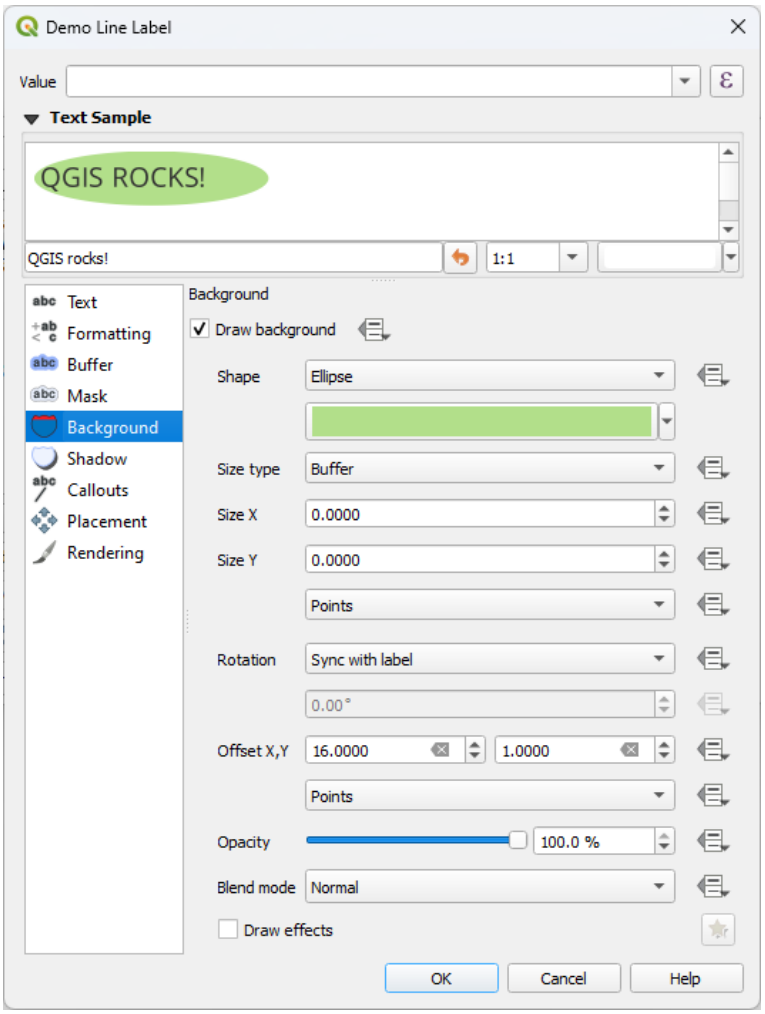


Abb. 10.29: Beschriftungseinstellungen - Hintergrund-Reiter

Je nach ausgewählter Form müssen Sie einige der folgenden Eigenschaften konfigurieren:

- Die *Größenart* des Rahmens, der sein kann:

  - *Fest*: Verwendung der gleichen Größe für alle Beschriftungen, unabhängig von der Größe des Textes
  - oder ein *Puffer* über der Bounding Box des Textes
- Die *Größe* des Rahmens in X- und Y-Richtung, unter Verwendung verschiedener :ref:` Einheiten <unit_selector>`
- Eine *Drehung* des Hintergrunds, entweder *Mit Beschriftung abgleichen*, *Beschriftungsversatz* oder *Fest*. Die letzten beiden erfordern einen Winkel in Grad.
- Ein *X,Y Versatz*, um das Hintergrundelement in die X- und/oder Y-Richtung zu verschieben
- Einen *X,Y Radius* zur Abrundung der Ecken der Hintergrundform (gilt nur für Rechteck- und Quadratformen)
- Eine *Deckkraft* des Hintergrunds
- Ein *Mischmodus*, um den Hintergrund mit den anderen Elementen im Rendering zu mischen (siehe *Mischmodi*).
- Für SVG-Symbole können Sie die Standardeigenschaften (*Symbolparameter laden*) verwenden oder eine benutzerdefinierte *Füllfarbe*, *Strichfarbe* und *Strichbreite* festlegen.
- Markieren Sie *Zeicheneffekte*, um erweiterte *Effekte* zur Verbesserung der Lesbarkeit von Text hinzuzufügen, z.B. durch äußeres Leuchten und Verwischen.

Schatten-Reiter

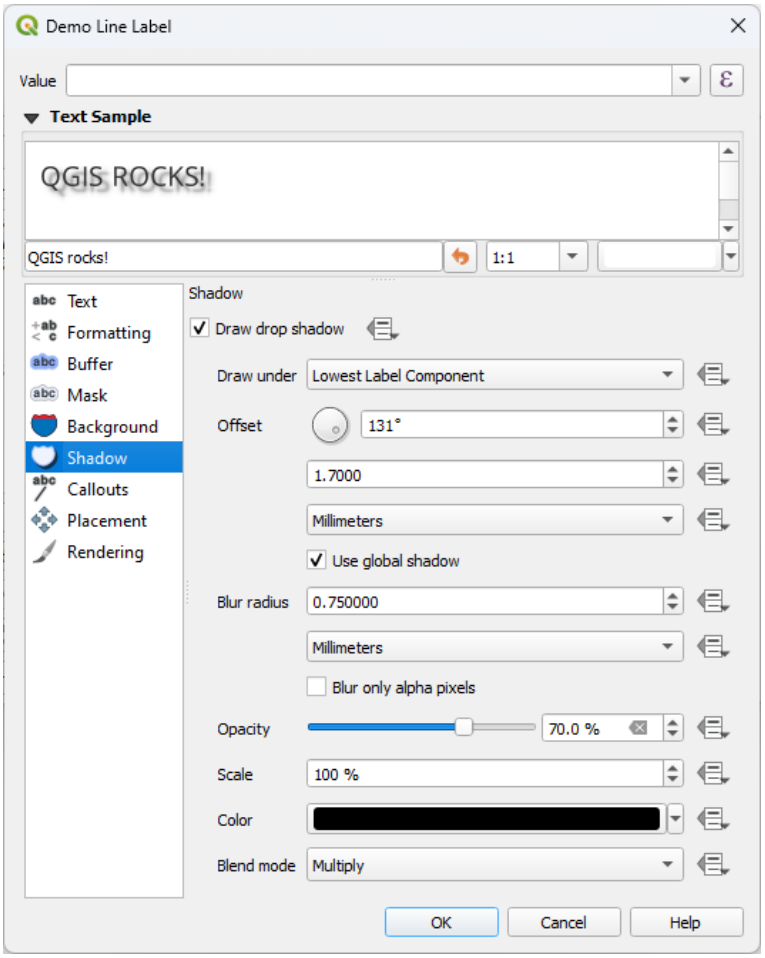


Abb. 10.30: Beschriftungseinstellungen - Schatten-Reiter

Um dem Text einen Schatten hinzuzufügen, aktivieren Sie den Reiter *Schatten* und aktivieren Sie die *Schattenwurf zeichnen*. Sie können dann:

- Geben Sie mit *Zeichnen unter* das Element an, das zur Erzeugung des Schattens verwendet wird. Es kann die *Niedrigste Beschriftungskomponente* oder eine bestimmte Komponente wie der *Text* selbst, der *Puffer* oder der *Hintergrund* sein.
- Legen Sie den *Versatz* des Schattens vom verschattenden Element fest, d.h.:
  - Winkel im Uhrzeigersinn; hängt von der Ausrichtung des zugrundeliegenden Objekts ab
  - Der Abstand des Versatzes vom schattierenden Element
  - Die Einheiten des Versatzes

  Wenn Sie das Kontrollkästchen *Globalen Schatten verwenden* aktivieren, dann ist der Winkel immer nach Norden ausgerichtet und hängt nicht von der Ausrichtung der Beschriftung ab.
- Beeinflussen Sie das Aussehen des Schattens mit dem *Verwischradius*. Je höher die Zahl, desto weicher der Schatten, in den Einheiten Ihrer Wahl.
- Legen Sie die *Deckkraft* des Schattens fest
- Die Größe des Schattens mit dem Faktor *Maßstab* neu skalieren
- Wählen Sie die *Farbe* für den Schatten
- Verwenden Sie die Option *Mischmodi*, um festzulegen, wie sich der Schatten Ihrer Beschriftung mit den darunter liegenden Kartenkomponenten vermischt (weitere Details unter blend-modus).

10.3.2 Konfigurieren der Interaktion mit Beschriftungen

Neben den oben beschriebenen Einstellungen zur Textformatierung können Sie auch festlegen, wie Beschriftungen untereinander oder mit den Objekten interagieren sollen.

Maske-Reiter

Mit dem Reiter *Maske* können Sie einen Maskenbereich um die Beschriftungen definieren. Diese Funktion ist sehr nützlich, wenn Sie überlappende Symbole und Beschriftungen mit ähnlichen Farben haben und die Beschriftungen sichtbar machen wollen. Eine Beschriftungsmaske verhindert, dass bestimmte Objekte innerhalb des für die Maske festgelegten Bereichs gezeichnet werden. Sie können beispielsweise eine Beschriftungsmaske so einstellen, dass ein bestimmter Layer nicht innerhalb von 2 mm um die Beschriftung herum gezeichnet wird, Objekte eines anderen Layers aber weiterhin angezeigt werden können. Beschriftungsmasken ähneln den Beschriftungspuffern insofern, als sie die Lesbarkeit von Beschriftungen steuern, die andere Objekte verdecken. Der Beschriftungspuffer zeichnet über alle darunter liegenden Objekte, während die Beschriftungsmaske selektiv das Zeichnen auf anderen Layern verhindert.

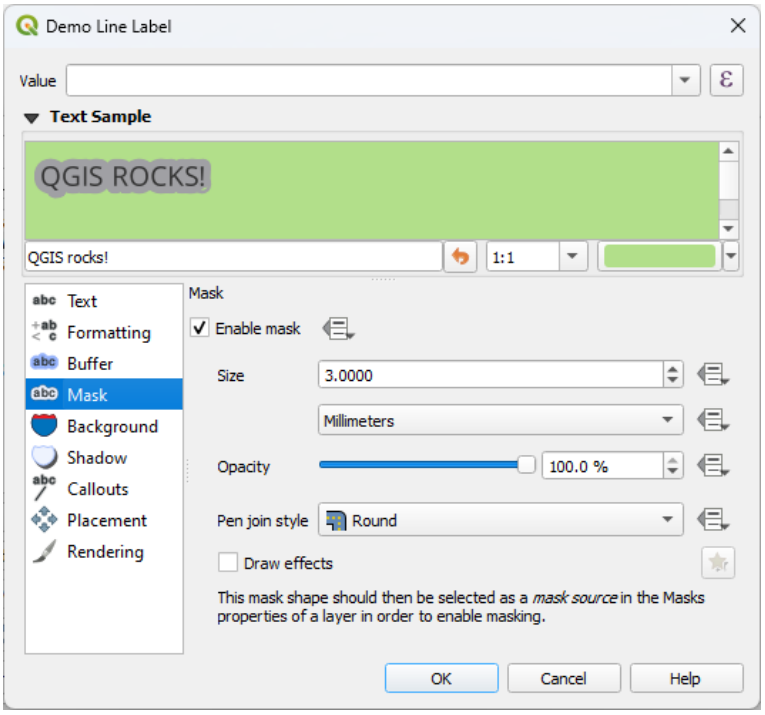


Abb. 10.31: Beschriftungseinstellungen - Masken-Reiter (wobei das Textbeispiel einen grünen Hintergrund zeigt, der einen anderen Layer darstellt, der ausgeschlossen wurde)

So erstellen Sie Maskierungseffekte auf Beschriftungen:

1. Aktivieren Sie das Kontrollkästchen *Maske aktivieren* auf dem Reiter .
2. Sie können dann einstellen:
   - die *Größe* der Maske in verschiednen *Einheiten*
   - die *Deckkraft* des Maskenbereichs um die Beschriftung
   - einen *Stiftübergangsstil*
   - besondere *Effekte* über das Kontrollkästchen *Zeicheneffekte*.
3. Wählen Sie diese Maskenform als Maskenquelle in den Eigenschaften des überlappenden Layers *Maske* Reiter (siehe *Eigenschaften Masken*).

Führungslinien-Reiter

Eine gängige Praxis bei der Platzierung von Beschriftungen auf einer überladenen Karte ist die Verwendung von **Führungslinien** - Beschriftungen, die außerhalb (oder versetzt) des zugehörigen Objekts platziert werden, werden durch eine dynamische Linie gekennzeichnet, die die Beschriftung und das Objekt verbindet. Wenn eines der beiden Enden (entweder die Beschriftung oder das Objekt) verschoben wird, wird die Form der Verbindung neu berechnet.

Abb. 10.32: Beschriftungen mit verschiedenen Führungslinien-Einstellungen

Um einer Beschriftung eine Führungslinie hinzuzufügen, gehen Sie zum Reiter *Führungslinie* und aktivieren das Kontrollkästchen *Führungslinie zeichnen*. Dann können Sie:

1. Wählen Sie den *Stil* des Verbinders, zur Auswahl stehen:
   - *Einfache Linien*: eine gerade Linie, der kürzeste Weg
   - *Manhattan Linien*: eine 90° gestrichelte Linie
   - *Kurven*: eine gebogene Linie
   - *Ballone*: eine Sprechblase, die die Beschriftung umgibt und auf das Objekt verweist. Sie kann abgerundete Ecken haben.
2. Legen Sie die Eigenschaften der Führungslinie fest. Die folgende Tabelle zeigt die verschiedenen Eigenschaften mit Beschreibung und kompatiblem Verbinderstil.

Tab. 10.5: Führungslinien-Eigenschaften

Eigenschaft	Stil der Führungslinie	Beschreibung
Füllstil	Ballone	Ein *Füllsymbol* mit vollständigen Darstellungsmöglichkeiten, einschließlich Layer-Effekten, datendefinierten Einstellungen usw., zum Zeichnen der Ballonform
Eckradius		Eckenradius der Sprechblase
Keilbreite		Legt fest, wie groß die Sprechblasen-Verbindung mit Führungslinie sein soll
Ränder		Ränder um den Text in der Beschriftung, in verschiedenen Einheiten

Fortsetzung auf der nächsten Seite

Tab. 10.5 – Fortsetzung der vorherigen Seite

Eigenschaft	Stil der Führungslinie	Beschreibung
Linienstil	Alles außer Luftballons	Ein *Liniensymbol* mit vollständigen Darstellungsmöglichkeiten, einschließlich Layer-Effekten, datendefinierten Einstellungen usw., zum Zeichnen der Führungslinie
Krümmung	Kurven	Der Prozentsatz der Krümmung der Führungslinie
Orientierung		Orientierung, beginnend bei der Beschriftung bis zum Objekt. Die Optionen sind *Im Uhrzeigersinn*, *Gegen den Uhrzeigersinn*, or *Automatisch* (Bestimmung einer optimalen Ausrichtung für jede Beschriftung)
Minimallänge	Alles außer	Mindestlänge der Führungslinie
Linien zu allen Objektteilen zeichnen	Luftballons	Legt bei mehrteiligen Objekten fest, ob eine Verbindungslinie vom Beschriftungstext zu jedem Geometrieelement gezogen werden soll oder nicht
Beschriftungsbezugspunktposition		Steuert, wo die Führungslinie mit dem Beschriftungstext verbunden werden soll. Verfügbare Optionen: • *Nächster Punkt* • *Zentroid* • Feste Position am Rand (*Oben links*, *Oben Mitte*, *Oben rechts*, *Links Mitte*, *Rechts Mitte*, *Unten links*, *Unten Mitte* und *Unten rechts*).
Abstand vom Beschriftungsbereich		Steuert den Abstand vom Ankerpunkt der Beschriftung (wo die Führungslinie endet); dadurch wird vermieden, dass Linien direkt an den Text angelegt werden.
Abstand vom Objekt	Alle	Steuert den Abstand vom Objekt (oder dessen Ankerpunkt, wenn es sich um ein Polygon handelt), an dem die Führungslinien enden. Dadurch wird beispielsweise vermieden, dass Linien direkt an den Kanten der Objekte gezeichnet werden.
Objektankerpunkt		Wo die Verbindungslinie auf dem (Polygon-)Objekt endet. Verfügbare Optionen: • *Pol der Unzugänglichkeit* • *Punkt auf der Außenseite* • *Punkt auf der Oberfläche* • *Zentroid*
Endpunktmarkierung	Ballone	Ein *Markierungssymbol* mit vollständigen Darstellungsmöglichkeiten, einschließlich Layer-Effekten, datendefinierten Einstellungen usw., zum Zeichnen eines Markierungssymbols am Ende der Ballonbeschriftung
Mischmodus	Alle	Steuert die *Mischung* der Führungslinie

Im Bereich *Datendefinierte Platzierung* können die Koordinaten des *Herkunftspunktes* (auf der Seite der Beschriftung) und/oder des *Zielpunktes* (auf der Seite des Objektes) der Beschriftung gesteuert werden. Beschriftungen können auch manuell gesteuert werden, indem man das Werkzeug Beschriftung, Diagramm oder Beschriftung verschieben in der *Beschriftungswerkzeugleiste* verwendet. Die Start- und Endpunkte jeder Beschriftung können auf diese Weise verschoben werden. Die Knoten sollten hervorgehoben werden, wenn sich der Mauszeiger in der Nähe befindet. Bei Bedarf kann die `Umschalttaste` während der Bewegung gehalten werden. Dadurch wird der Punkt so eingerastet, dass der Winkel zwischen den beiden Beschriftungspunkten um 15 Grad zunimmt.

Platzierung-Reiter

Wählen Sie die Registerkarte *Platzierung*, um die Platzierung und die Prioritäten für die Beschriftung zu konfigurieren. Beachten Sie, dass die Platzierungsoptionen je nach Typ der Vektor- oder Mesh-Layer (Punkt, Linie oder Polygon) variieren und von der globalen *PAL-Einstellung* beeinflusst werden.

Platzierung für Punkt-Layer

Folgende Modi für die Platzierung von Punktbeschriftungen sind verfügbar:

- *Kartografisch*: Beschriftungen werden unter Berücksichtigung der Sichtbarkeit vom Objekt erzeugt und folgen kartografischen Platzierungsregeln. Beschriftungen können platziert werden:
  - mit einer festgelegten *Entfernung* in verschiedenen *Einheiten*, entweder von dem Punktobjekt selbst oder von den Grenzen des Symbols, das zur Darstellung des Objekts verwendet wird (festgelegt in *Abstand aus*). Die letztere Option ist besonders nützlich, wenn die Symbolgröße nicht fest ist, z.B. wenn sie durch eine datendefinierte Größe festgelegt ist oder wenn verschiedene Symbole in einem *kategorisierten* Renderer verwendet werden.
  - innerhalb eines *Maximalabstands* vom Objekt. Diese optionale Einstellung ermöglicht es Ihnen zu steuern, wie weit ein Label vom Objekt, das es beschriftet, entfernt platziert werden kann. In Verbindung mit der Einstellung *Entfernung* lässt sich so ein Bereich für die Platzierung von Labels festlegen. Dies erhöht die Flexibilität bei der effektiven Positionierung von Labels, insbesondere auf überladenen Karten, und stellt sicher, dass sie sich sauber um die entsprechenden Objekte herum anordnen lassen.
  - unter Verwendung der Option *Platzierung priorisieren*, die festlegt, was beim Platzieren von Beschriftungen wichtiger ist. Es gibt zwei Optionen:
    * *Nähere Beschriftungen vorziehen*: Standardmäßig werden Beschriftungen nahe am Objekt platziert.
    * :guilabel:`Positionsreihung vorziehen: die Beschriftung versucht, an einer bestimmten Position zu bleiben (z. B. oben links oder oben rechts), auch wenn es etwas weiter vom Objekt entfernt ist. Die Beschriftung wird nur dann an eine andere Position verschoben, wenn innerhalb der maximalen Entfernung an Ihrer bevorzugten Position kein Platz vorhanden ist.
  - nach einer *Positionspriorität*, die die Platzierungskandidaten für die Verankerung von Beschriftungen um und (zentriert) über dem Punktmerkmal sowie die Reihenfolge festlegt, in der die Positionen getestet werden. Die Standardreihenfolge, basierend auf den Richtlinien von Krygier und Wood (2011) und anderen kartografischen Lehrbüchern, lautet wie folgt:
    1. oben rechts
    2. oben links
    3. unten rechts
    4. unten links
    5. Mitte rechts
    6. Mitte links
    7. oben, leicht rechts
    8. unten, leicht links.

    Mit der Schaltfläche Datendefinierte Übersteuerung können Sie eine durch Kommata getrennte Liste von Platzierungen in der Reihenfolge ihrer Priorität angeben. Dadurch können auch nur bestimmte Platzierungen für bestimmte Objekte verwendet werden, sodass Sie beispielsweise bei Küstenobjekten verhindern können, dass Beschriftungen über dem Land platziert werden.
- *Um Punkt*: Beschriftungen werden in einem Kreis um das Objekt herum platziert, wobei der Radius mit *Abstand* festgelegt wird. Zusätzlich können Sie den *Maximalabstand* vom Objekt festlegen, um zu steuern, wie weit eine

Beschriftung vom Objekt entfernt platziert werden kann. Die Platzierungspriorität ist im Uhrzeigersinn von „oben rechts“. Die Position kann mit der datendefinierten Option *Quadrant* eingeschränkt werden.

- *Abstand vom Punkt*: Beschriftungen werden mit einem *X,Y Versatz* vom Punkt-Feature platziert oder vorzugsweise auf dem Objekt. Es kann eine datendefinierte Überschreibung des *Quadrant* verwenden werden, um die Platzierung einzuschränken. Sie können der Beschriftung auch eine *Drehung* zuweisen.

Platzierung für Linien-Layer

Zu den Beschriftungsmodi für Linien-Layer gehören:

- *Parallel*: zeichnet die Beschriftung parallel zu einer verallgemeinerten Linie, die das Objekt darstellt, mit Vorzug für die Platzierung über geradere Abschnitte der Linie. Sie können definieren:
  - *Erlaubte Positionen*: *Über Linie*, *Auf Linie*, *Unter Linie* und *Position abhängig von der Linienrichtung* (Platzierung der Beschriftung links oder rechts der Linie). Es ist möglich, mehrere Optionen auf einmal zu wählen. In diesem Fall sucht QGIS nach der optimalen Position für die Beschriftung.
  - *Entfernung* zwischen der Beschriftung und der Linie
- *Gebogen*: zeichnet die Beschriftung entsprechend der Krümmung des Objekts. Zusätzlich zu den Parametern, die im Modus *Parallel* verfügbar sind, können Sie den *Größter Winkel zwischen Zeichen auf Kurven* einstellen, entweder innerhalb oder außerhalb.
- *Horizontal*: zeichnet Beschriftungen horizontal entlang der Länge des Linienobjekts.

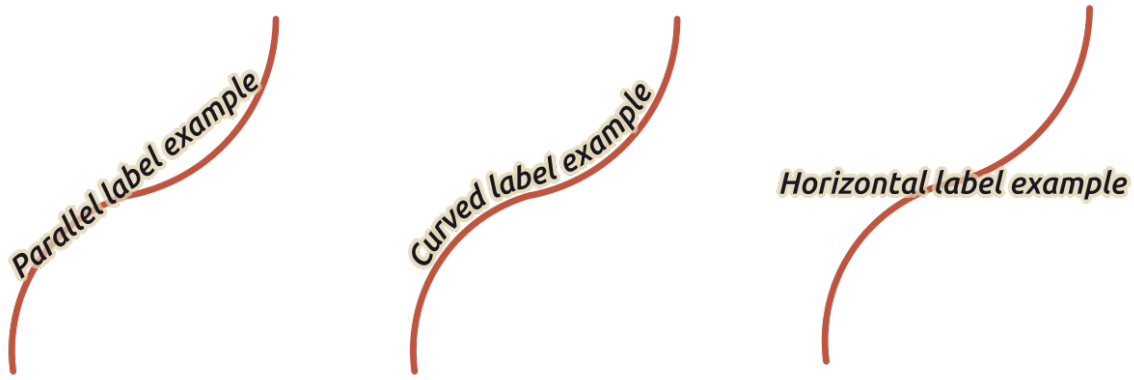


Abb. 10.33: Beispiele für die Platzierung von Beschriftungen für Linien

Außer den Platzierungsmodi können Sie einstellen:

- *Beschriftungswiederholung*: *Entfernung*, um die Beschriftung mehrfach über die Länge des Objekts anzuzeigen.
- A *Beschriftungsüberlauf Überschreibungsabstand* (nicht verfügbar für den horizontalen Modus): gibt den maximal zulässigen Abstand an, den eine Beschriftung über das Ende (oder den Anfang) von Linienobjekten hinauslaufen darf. Durch Erhöhen dieses Wertes können Beschriftungen für kürzere Objekte angezeigt werden.
- *Beschriftungsanker*: steuert die Platzierung der Beschriftungen entlang des Linienobjekts, auf das sie sich beziehen. Klicken Sie auf *Einstellungen* …, um zu wählen:
  - die Position entlang der Linie (als Verhältnis), in deren Nähe die Beschriftungen platziert werden sollen. Der Wert kann durch Daten definiert werden, mögliche Werte sind:
    - *Linienmitte*
    - *Linienanfang*
    - *Linienende*

    - oder *Benutzerdefiniert…*.
  - *Ausschneiden*: Bestimmt, wie die Platzierung der Beschriftung auf einer Linie berechnet wird. Standardmäßig wird nur die sichtbare Ausdehnung der Linie verwendet, aber die gesamte Ausdehnung kann verwendet werden, um konsistentere Ergebnisse zu erzielen.
  - *Textanker*: steuert, welcher Teil des Textes (Anfang, Mitte oder Ende) mit dem Ankerpunkt übereinstimmen soll. Die Verwendung von *Automatisch* bedeutet, dass:
    - Bei Beschriftungen, die in der Nähe des Zeilenanfangs verankert sind (0-25%), wird der Anker am **Anfang** des Beschriftungstextes gesetzt
    - Bei Beschriftungen, die in der Nähe des Zeilenendes (75-100%) verankert sind, wird der Anker am **Ende** des Beschriftungstextes platziert
    - Bei Beschriftungen, die in der Nähe der Zeilenmitte verankert sind (25-75%), wird die Verankerung in der **Mitte** des Beschriftungstextes platziert
  - *Platzierungsverhalten*: Verwenden Sie *Bevorzugter Platzierungshinweis*, um den Beschriftungsanker nur als Hinweis für die Platzierung der Beschriftung zu behandeln. Wenn Sie *Streng* wählen, werden die Beschriftungen genau auf dem Beschriftungsanker platziert.

Platzierung für Polygon Layer

Sie können einen der folgenden Modi für die Beschriftung von Polygonen wählen:

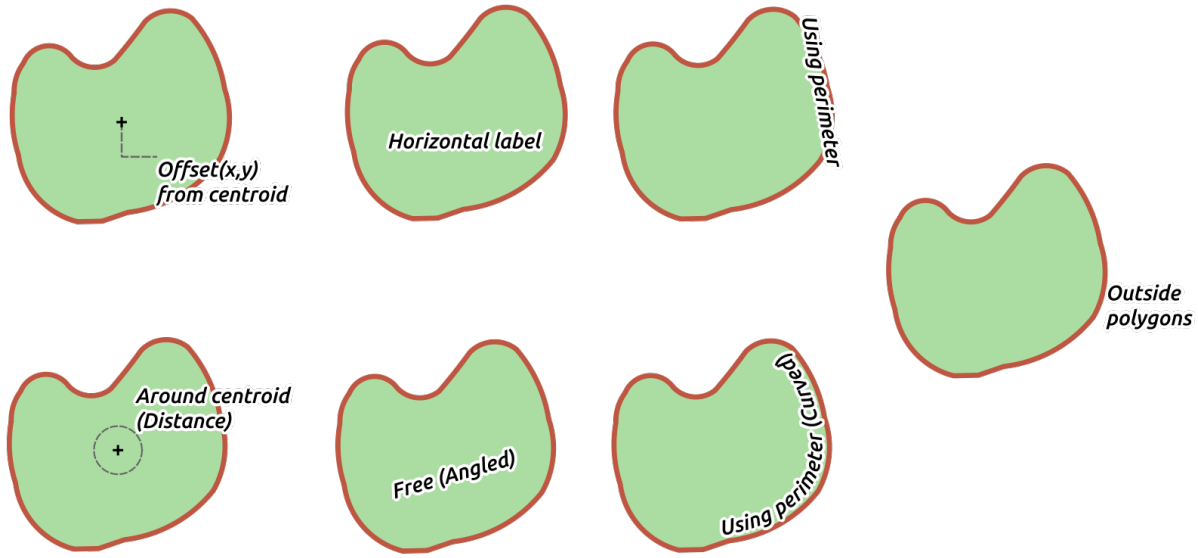


Abb. 10.34: Beispiele für die Platzierung von Beschriftungen für Polygone

- *Abstand vom Zentrum*: Beschriftungen werden über dem Objektschwerpunkt oder in einem festen *X,Y Versatz* vom Schwerpunkt platziert. Der Referenzschwerpunkt kann auf der Grundlage des in der Kartenansicht gerenderten Teils des Polygons (*sichtbarem Polygon*) oder des *ganzen Polygons* bestimmt werden, unabhängig davon, ob man es sehen kann. Sie können auch:
  - Punkt innerhalb des Polygons erzwingen
  - die Beschriftung in einem bestimmten Quadranten platzieren
  - eine Drehung zuweisen
  - *Beschriftungen außerhalb von Polygonen erlauben*, wenn es nicht möglich ist, sie innerhalb des Polygons zu platzieren. Mithilfe von datendefinierten Eigenschaften ist es möglich, Beschriftungen außerhalb des Polygons zuzulassen, zu verhindern oder zu erzwingen, dass sie außerhalb des Polygons platziert werden, und zwar für jedes einzelne Objekt.

- *Um Zentroid*: platziert die Beschriftung in einem voreingestellten Abstand um den Zentroid, mit einer Präferenz für die Platzierung direkt über dem Zentroid. Auch hier können Sie festlegen, ob der Schwerpunkt der des *sichtbaren Polygon* oder der des *ganzen Polygon* ist, und ob der Schwerpunkt innerhalb des Polygons platziert werden soll.
- *Horizontal*: platziert eine horizontale Beschriftung an der besten Position innerhalb des Polygons. Die bevorzugte Platzierung ist weiter von den Kanten des Polygons entfernt. Es ist möglich, *Beschriftungen außerhalb von Polygonen zu erlauben*.
- *Frei (Winkel)*: platziert eine gedrehte Beschriftung an der besten Position innerhalb des Polygons. Die Drehung respektiert die Orientierung des Polygons und die bevorzugte Platzierung ist weiter von den Kanten des Polygons entfernt. Es ist möglich, *Beschriftungen außerhalb von Polygonen zu erlauben*.
- *Nach Umfang*: zeichnet die Beschriftung parallel zu einer verallgemeinerten Linie, die die Polygongrenze darstellt, mit Vorzug für geradere Teile des Umfangs. Sie können definieren:
  - *Erlaubte Positionen*: *Über Linie*, *Auf Linie*, *Unter Linie* und *Position abhängig von Linienrichtung* (Platzierung der Beschriftung links oder rechts der Polygongrenze). Es ist möglich, mehrere Optionen auf einmal zu wählen. In diesem Fall sucht QGIS nach der optimalen Position für die Beschriftung.
  - *Entfernung* zwischen der Beschriftung und dem Umriss des Polygons
  - das *Beschriftungswiederholung Entfernung*, um die Beschriftung über die Länge der Umrandung mehrfach anzuzeigen.
- *Umfang (kurvig) verwenden*: zeichnet die Beschriftung entsprechend der Krümmung der Polygonbegrenzung. Zusätzlich zu den Parametern, die mit dem Modus *Umfang verwenden* verfügbar sind, können Sie den *Größter Winkel zwischen Zeichen auf Kurven*, entweder innerhalb oder außerhalb, einstellen.
- *Außerhalb der Polygone*: platziert Beschriftungen immer außerhalb der Polygone, in einer bestimmten *Entfernung*

Allgemeine Platzierungseinstellungen

Einige Einstellungen für die Platzierung von Beschriftungen sind für alle Layer-Geometrietypen verfügbar:

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*.

Achtung

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.

Achtung

This option is only available on QGIS installed with GEOS >= 3.10 (see *Help* ► *About* menu).

Geometrie-Generator

Der Bereich *Geometrie-Generator* ermöglicht es dem Benutzer, die der Beschriftung zugrunde liegende Geometrie zu verändern, indem er *Ausdrücke* verwendet. Dies kann nützlich sein, um die Geometrie dynamisch zu verschieben oder sie in eine andere Geometrie (Typ) umzuwandeln.

Um den Geometriegenerator zu verwenden:

1. Aktivieren Sie die Option *Geometriegenerator*.
2. Geben Sie den Ausdruck ein, der die Geometrie erzeugt, auf die Sie sich beziehen wollen
3. Wählen Sie gegebenenfalls den Geometrietyp der Ausdrucksausgabe: Die auf der Beschriftungsgeometrie basierenden Einstellungen wie Platzierung oder Rendering werden aktualisiert, um den neuen Geometrietyp-Fähigkeiten zu entsprechen.

Zu den Anwendungsfällen zählen:

- Verwendung einer Geometrie, die in einem anderen Feld „label_position“ gespeichert ist
- Verwenden Sie die *generierte Geometrie* aus der Symbologie auch zur Beschriftung
- Verwenden Sie die Variable @map_scale, um Entfernungen / Größen unabhängig von der Zoomstufe zu berechnen.
- In Kombination mit dem Modus für die gekrümmte Platzierung wird eine kreisförmige Beschriftung um ein punktförmiges Objekt erstellt:

```
exterior_ring(make_circle($geometry, 20))
```

- Hinzufügen einer Beschriftung am Anfang und am Ende eines Linien-Objekts:

```
collect_geometries( start_point($geometry), end_point($geometry) )
```

- Wählen Sie eine geglättete Flusslinie, um mehr Platz für die Platzierung der Beschriftung zu erhalten:

```
smooth( $geometry, iterations:=30, offset:=0.25, min_length:=10 )
```

Datendefiniert

Die *Datendefinierte Übersteuerung* bietet eine direkte Kontrolle über die Platzierung von Beschriftungen der einzelnen Objekte. Sie kann über die Attribute der Objekte oder über einen Ausdruck gesteuert werde. Festgelegt werden können:

- die *X*- und *Y*-Koordinaten
- die Textausrichtung über die oben eingestellte benutzerdefinierte Position:
  - *Horizontal*; die Optionen sind: **Links**, **Mitte** oder **Rechts**
  - *Vertikal*; die Optionen sind: **Basis**, **Hälfte**, **Kappe** oder **Oben**
- Die *Drehung* des Textes wird als Winkel im Uhrzeigersinn definiert, wobei 0° für horizontal ausgerichteten Text in Richtung Osten und für vertikal ausgerichteten Text in Richtung Norden zeigt. Für die Rotation der Beschriftung können verschiedene Einheiten definiert werden (z. B. „Grad“, „Bogenminuten“, „Drehungen“). QGIS priorisiert standardmäßig die Drehwinkel der Bildschirmanzeige und dreht den Text immer so, dass er auf dem Bildschirm am besten lesbar ist, sodass „umgekehrte” Drehwinkel beim Rendern überschrieben werden. Um dieses Verhalten zu ändern und QGIS zu zwingen, die exakten datendefinierten Drehwinkel zu rendern, gibt es eine Option auf dem Reiter **|Rendern|** *Darstellung* (s.u.) in den Beschriftungseinstellungen, unter *Kopfstehende Beschriftungen anzeigen*. Aktivieren Sie die Option *Datenrotationswerte erhalten*, wenn Sie den Rotationswert im zugehörigen Feld beibehalten und auf die Beschriftung anwenden möchten, unabhängig davon, ob die Beschriftung fixiert ist oder nicht. Wenn diese Option deaktiviert ist, wird die Beschriftungsrotation zurückgesetzt und ihr Wert aus der Attributtabelle gelöscht.

Bemerkung

Die datendefinierte Rotation mit Polygonobjekten wird derzeit nur mit dem Platzierungsmodus *Um Zentroid* unterstützt.

Bemerkung

Ausdrücke können nicht in Kombination mit den Werkzeugen für die Zuordnung von Beschriftungen (d.h. den Werkzeugen *Etikett drehen* und *Etikett verschieben*) verwendet werden, um die Platzierung von *datendefinierter* Beschriftungen zu verwenden. Das Widget wird auf das entsprechende *Hilfsspeicherfeld* zurückgesetzt.

Priorität

Im Bereich *Priorität* können Sie den Rang der einzelnen Beschriftungen bei der Platzierung festlegen, d.h. wenn es verschiedene Diagramme oder Beschriftungen gibt, die für denselben Ort in Frage kommen, wird das Element mit der höheren Priorität angezeigt und die anderen können ausgelassen werden.

Der Prioritätsrang wird auch verwendet, um zu bewerten, ob eine Beschriftung aufgrund eines höher gewichteten *Hindernisobjekts* weggelassen werden kann.

Hindernisse

In manchen Kontexten (z. B. hohe Dichte der Beschriftungen, überlappende Objekte…) kann die Platzierung der Beschriftungen dazu führen, dass Beschriftungen über nicht zusammenhängenden Objekten platziert werden.

Ein Hindernis ist ein Objekt, über dem QGIS keine Beschriftungen oder Diagramme für andere Objekte platziert. Dies kann über den Abschnitt *Hindernisse* gesteuert werden:

1. Aktivieren Sie das Kontrollkästchen *Objekte wirken als Hindernisse*, um festzulegen, dass Objekte des Layers als Hindernisse für Beschriftungen und Diagramme wirken sollen (auch für Elemente anderer Objekte desselben Layers).

   Anstelle des gesamten Layers können Sie eine Untergruppe von Objekten auswählen, die als Hindernisse verwendet werden sollen, indem Sie das Steuerelement Datendefinierte Überschreibung neben der Option verwenden.

2. Verwenden Sie die Schaltfläche *Einstellungen*, um die Gewichtung des Hindernisses zu ändern.

   - Für jedes potenzielle Objekt können Sie eine *Hindernisgewichtung* vergeben: jede *Beschriftung* oder jedes *Diagramm*, deren Platzierungspriorität größer als dieser Wert ist, kann darüber platziert werden. Beschriftungen oder Diagramme mit niedrigerem Rang werden ausgelassen, wenn keine andere Platzierung möglich ist.

     Diese Gewichtung kann auch datenabhängig sein, so dass innerhalb eines Layers bestimmte Objekte mit größerer Wahrscheinlichkeit erfasst werden als andere.

   - Bei Polygon-Layern können Sie auswählen, um welche Art von Hindernis es sich bei dem Objekt handelt:

     – **über dem Objektinneren**: vermeidet die Platzierung von Beschriftungen über dem Inneren des Polygons (bevorzugt die Platzierung von Beschriftungen ganz außerhalb oder nur leicht innerhalb des Polygons)

     – oder **über der Objektgrenze**: vermeidet die Platzierung von Beschriftungen über der Grenze des Polygons (bevorzugt die Platzierung von Beschriftungen außerhalb oder vollständig innerhalb des Polygons). Dies kann bei Layern nützlich sein, bei denen die Objekte das gesamte Gebiet abdecken (Verwaltungseinheiten, kategorische Abdeckungen, …). In diesem Fall lässt es sich nicht vermeiden, Beschriftungen innerhalb dieser Objekte zu platzieren, und es sieht viel besser aus, wenn die Beschriftung nicht über den Grenzen zwischen den Objekten platziert wird.

Darstellungs-Reiter

Im Reiter *Darstellung* können Sie einstellen, wann die Beschriftungen dargestellt werden können und wie sie mit anderen Beschriftungen und Objekten interagieren.

Beschriftungsoptionen

Unter *Beschriftungsoptionen* finden Sie folgende Einstellungsmöglichkeiten:

- Festlegen einer *Maßstabsabhängigen Sichtbarkeit* und einer *Sichtbarkeit nach Pixelgröße*.
- *Z-index der Beschriftung*: bestimmt die Reihenfolge, in der Beschriftungen gerendert werden, sowohl in Bezug auf andere Objektbeschriftungen im Layer (unter Verwendung von datendefinierten Überschreibungsausdrücken), als auch auf Beschriftungen aus anderen Layern. Beschriftungen mit einem höheren Z-index werden über Beschriftungen (aus einem beliebigen Layer) mit einem niedrigeren Z-index gerendert.

  Die Logik des Z-Index wurde dahingehend geändert, dass zwei Beschriftungen mit übereinstimmenden Z-Indizes zusammengehören, wobei folgende Regeln Anwendung finden:

  – wenn sie aus demselben Layer stammen, wird die kleinere Beschriftung über der größeren Beschriftung gezeichnet

  – wenn sie aus verschiedenen Layern stammen, werden die Beschriftungen in der gleichen Reihenfolge wie die Layer selbst gezeichnet (d.h. unter Berücksichtigung der in der Kartenlegende festgelegten Reihenfolge)

  > **Bemerkung**
  >
  > Diese Einstellung bewirkt nicht, dass Beschriftungen unter den Objekten anderer Layer gezeichnet werden, sondern steuert nur die Reihenfolge, in der Beschriftungen über allen Objekten der Layer gezeichnet werden

- *Alle minderwertigen Ersatzplatzierungen erlauben*: Standardmäßig versucht QGIS, Beschriftungen an der bestmöglichen Stelle zu platzieren, entsprechend Ihren Einstellungen. Aktivieren Sie diesen Modus, um Objekten zu erlauben, auf schlechtere Platzierungsoptionen zurückzugreifen, wenn es keine andere Wahl gibt, z.B. wenn eine Linie zu kurz ist, um einen gekrümmten Beschriftungstext unterzubringen, dann kann die Beschriftung horizontal genau über dem Mittelpunkt des Objekts platziert werden.
- Mit der datendefinierten Übersteuerung in *Beschriftung anzeigen* und *Immer anzeigen* können Sie feinabstimmen, welche Beschriftungen angezeigt werden sollen.
- *Kopfstehende Beschriftungen anzeigen*, die Optionen sind: **nie**, **bei definiertem Winkel** oder **immer**

  – **nie** – Standardeinstellung, die Lesbarkeit des Bildschirms hat Vorrang

  – **bei definiertem Winkel** – Der Winkel der Beschriftung kann auf dem Reiter *Platzierungen* innerhalb der Gruppe *datendefinierte Übersteuerung* festgelegt werden

  – **immer** – Beschriftungen dürfen auf dem Kopf stehen

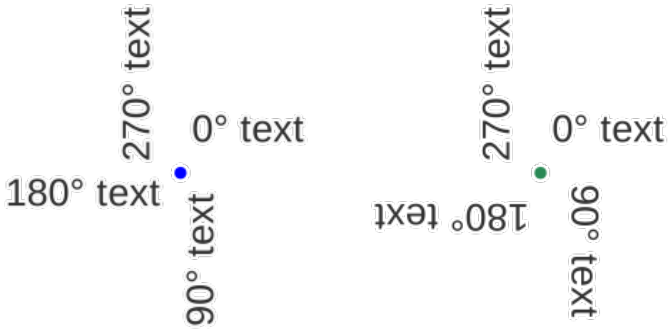


Abb. 10.35: Datendefinierte Beschriftungsdrehung mit der Option „Kopfstehende Beschriftungen anzeigen“ eingestellt auf: „nie“ (links) und „bei definiertem Winkel“ (rechts)

- Im Bereich *Überlappende Beschriftungen* können Sie steuern, ob überlappende Beschriftungen für Objekte im Layer zulässig sind und wie sie jeweils behandelt werden sollen:
  - *Nie überlappen*: Niemals überlappende Beschriftungen für den Layer platzieren, auch wenn das bedeutet, dass einige Beschriftungen fehlen
  - *Überlappungen bei Bedarf erlauben*: wenn die Beschriftung nicht anders platziert werden kann, wird eine überlappende Beschriftung gezeichnet. In diesem Modus wird die Beschriftung, wenn möglich, an eine weniger ideale Stelle verschoben, z.B. weiter vom Mittelpunkt einer Linie oder eines Polygons entfernt, WENN dadurch überlappende Beschriftungen vermieden werden. Wenn aber keine andere Position möglich ist, dann wird die Beschriftung überlappend gezeichnet.
  - *Überlappungen ohne Strafe zulassen*: Es macht überhaupt nichts, wenn die Beschriftung andere Beschriftungen oder Hindernisse überlappt, das ist in Ordnung und die beste Platzierung (z.B. die zentralste Platzierung) sollte immer verwendet werden, auch wenn eine alternative weitere Platzierung möglich ist, die Überschneidungen völlig vermeidet.

  Wenn Sie sowohl überlappende Beschriftungen als auch Ersatzplatzierungen zulassen, wird sichergestellt, dass alle Objekte im Layer beschriftet werden… nicht unbedingt in ihrer besten Darstellung!

Objektoptionen

Unter *Objektoptionen*:

- Können Sie wählen zwischen *Alle Teile mehrteiliger Objekte beschriften* und eine *Obergrenze der zu beschriftenden Objekte* festlegen.
- Sowohl Linien- als auch Polygon-Layer bieten die Möglichkeit, mit *Objekte nicht beschriften, wenn kürzer/kleiner als* eine Mindestgröße für die zu beschriftenden Objekte festzulegen.
- Bei Polygon-Features können Sie die anzuzeigenden Beschriftungen auch danach filtern, ob sie vollständig in ihr Objekt passen oder nicht.
- Bei Linienobjekten können Sie die Option *Anschließende Linien zusammenführen, um doppelte Beschriftungen zu vermeiden* wählen.

10.4 3D-Symbole erstellen

Die *Stilverwaltung* hilft Ihnen bei der Erstellung und Speicherung von 3D-Symbolen für jeden Geometrietyp zur Darstellung in der *3D-Kartenansicht*.

Aktivieren Sie wie bei den anderen Elementen den Reiter *3D Symbols* und erweitern Sie das Menü der Schaltfläche , um sie zu erstellen:

- *3D Punkt-Symbole*
- *3D Linien-Symbole*
- *3D Polygon-Symbole*

10.4.1 3D Punkt-Symbole

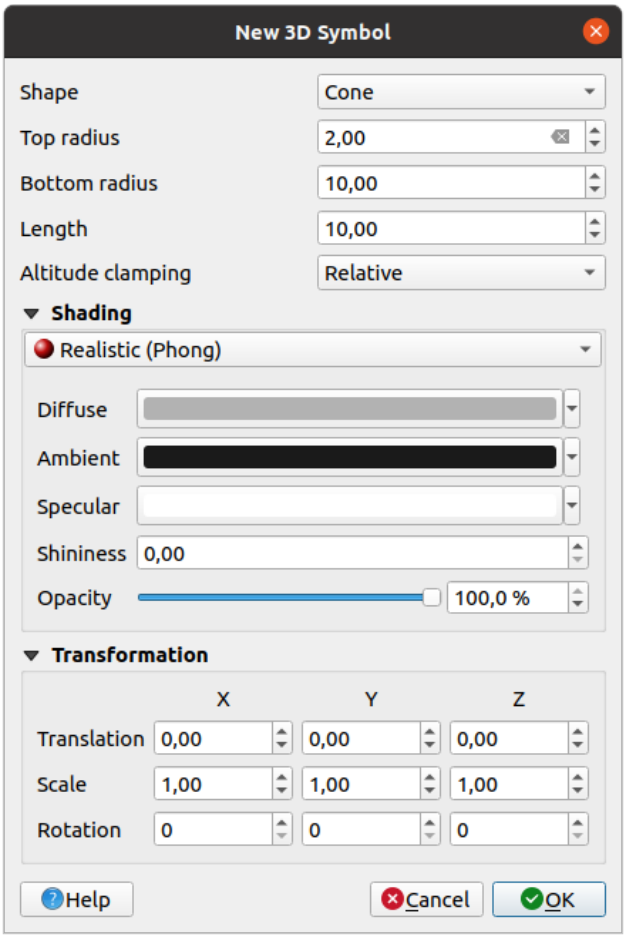


Abb. 10.36: Eigenschaften eines 3D-Punktsymbols

- Sie können verschiedene Arten von 3D *Form* für die Verwendung von Punktsymbolen definieren. Sie werden hauptsächlich durch ihre Abmessungen definiert, deren Einheit sich auf das KBS des Projekts bezieht. Verfügbare Typen sind:
  - *Sphäre* definiert durch einen *Radius*
  - *Zylinder* definiert durch *Radius* and *Länge*
  - *Würfel* definiert durch eine *Größe*

  - *Kegel* definiert durch *Radius oben*, *Radius unten* und eine *Länge*
  - *Ebene* definiert durch eine *Größe*
  - *Ring* definiert durch einen *Radius* und einen *Minimalen Radius*
  - *3D-Modell*, unter Verwendung einer 3D-Modelldatei: Unterstützte Formate sind Wavefront `.obj`, `.glTF` und `.fbx`. Modelle können eine Datei auf der Festplatte, eine Remote-URL oder *im Projekt eingebettet* sein. Von der Community erstellte Modelle werden auf dem QGIS Hub geteilt.
  - *Billboard*, definiert durch die *Billboard-Höhe* und das *Billboard-Symbol* (normalerweise basierend auf einem *Markersymbol*). Das Symbol wird eine stabile Größe haben. Praktisch für die Visualisierung von 3D-Punktwolken Shapes.
- Die *Höhenwinkelbegrenzung* kann auf *Absolut*, *Relativ* oder *Gelände* eingestellt werden. Die Einstellung *Absolut* kann verwendet werden, wenn die Höhenwerte der 3D-Vektoren als absolute Maße von 0 aus angegeben werden. *Relativ* und *Terrain* addieren die angegebenen Höhenwerte zur darunter liegenden Geländehöhe.
- Die *Schatten*-Eigenschaften können definiert werden.
- Im Rahmen *Transformationen* können Sie affine Transformationen auf das Symbol anwenden:
  - *Verschiebung*, um Objekte in der x-, y- und z-Achse zu verschieben.
  - *Maßstab* zur Größenänderung der 3D-Formen
  - *Rotation* um die x-, y- und z-Achse.

10.4.2 3D Linien-Symbole

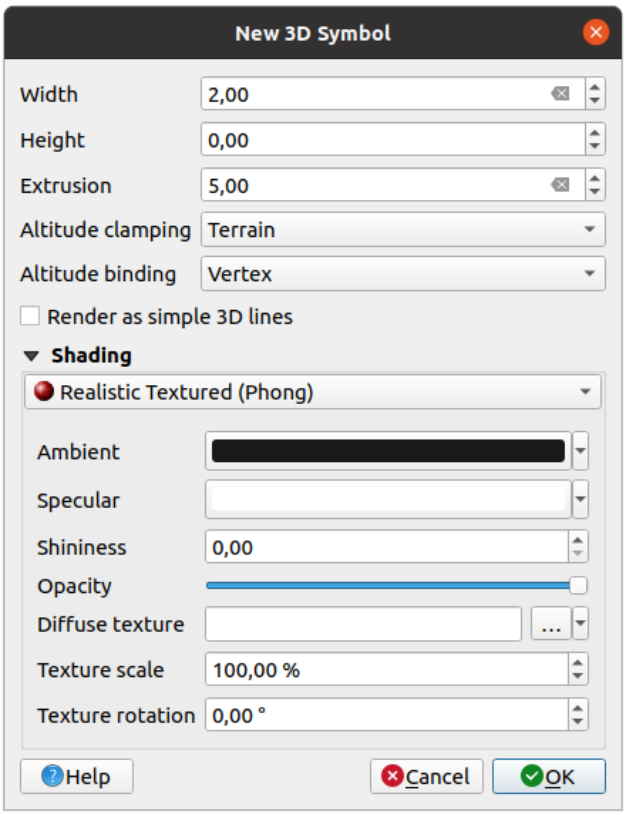


Abb. 10.37: Eigenschaften eines 3D-Liniensymbols

- Neben den Einstellungen *Breite* und *Höhe* können Sie die *Extrusion* der Vektorlinien definieren. Wenn die Linien keine z-Werte haben, können Sie mit dieser Einstellung das 3D-Volumen definieren.

- Mit der *Höhenwinkelbegrenzung* definieren Sie die Position der 3D-Linien relativ zur darunter liegenden Geländeoberfläche, wenn Sie Rasterhöhenangaben oder andere 3D-Vektoren eingebunden haben.
- Die *Höhenwinkelbindung* legt fest, wie das Objekt an das Terrain angepasst wird. Entweder wird jeder *Stützpunkt* des Objekts an das Terrain angepasst oder dies geschieht über den *Zentroid*.
- Die Option *Als einfache 3D-Linien anzeigen* kann aktiviert werden.
- Die *Schatten*-Eigenschaften können definiert werden.

10.4.3 3D Polygon-Symbole

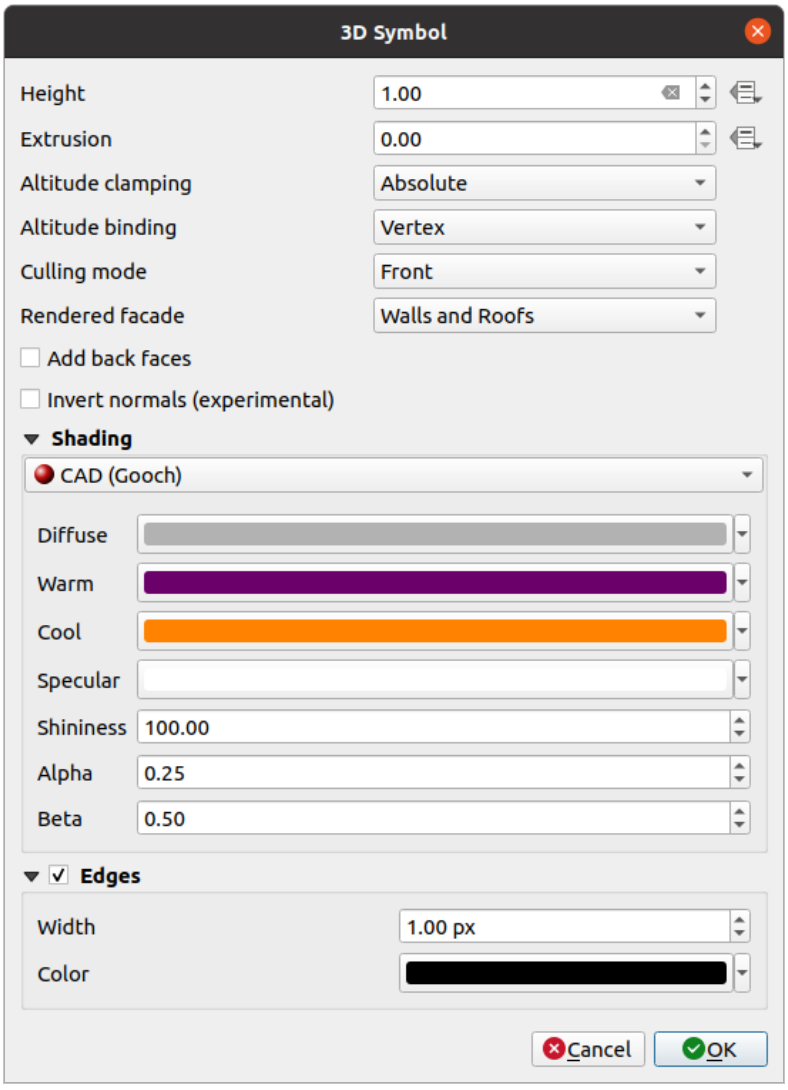


Abb. 10.38: Eigenschaften eines 3D-Polygonsymbols

- Wie bei den anderen Symbolen kann auch hier die *Höhe* in KBS-Einheiten definiert werden. Sie können auch die Schaltfläche verwenden, um den Wert mit einem benutzerdefinierten Ausdruck, einer Variablen oder einem Eintrag in der Attributtabelle zu überschreiben.
- Auch hier ist *Extrusion* für fehlende z-Werte möglich. Auch für die Extrusion können Sie die Schaltfläche verwenden, um die Werte des Vector Layers zu verwenden und für jedes Polygon unterschiedliche Ergebnisse zu erhalten:

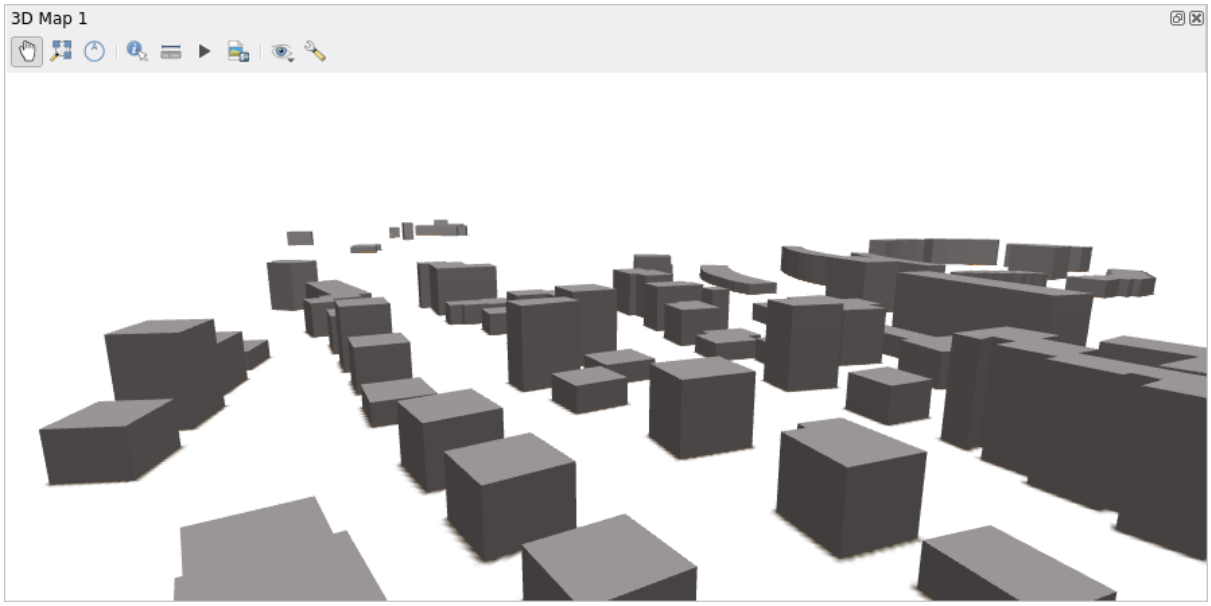


Abb. 10.39: Datendefinierte Extrusion

- Die *Höhenwinkelbegrenzung* und die *Höhenbindung* können wie oben erläutert definiert werden.
- Der *Culling-Modus* kann auf das Symbol angewendet werden; die Optionen sind:
  - *Kein Culling*: dies kann helfen, scheinbar fehlende Flächen zu vermeiden, wenn die PolygonZ/Multipatch-Daten keine konsistente Anordnung der Scheitelpunkte aufweisen (z.B. alle im oder gegen den Uhrzeigersinn)
  - *Vorderseite*
  - *Rückseite*
- Die Option *Fassade darstellen* bestimmt die anzuzeigenden Flächen. Mögliche Werte sind *Keine Fassaden*, *Wände*, *Dächer*, oder *Wände und Dächer*
- ☑ *Rückseiten hinzufügen*: erzeugt für jedes Dreieck sowohl Vorder- als auch Rückseiten mit korrekten Normalen - auf Kosten einer erhöhten Anzahl von Scheitelpunktdaten. Diese Option kann verwendet werden, um Schattierungsprobleme zu beheben (z.B. aufgrund von Daten mit inkonsistenter Reihenfolge der Scheitelpunkte).
- ☑ *Normalen invertieren (experimentell)*: kann nützlich sein, um die Reihenfolge der Eckpunkte von Flächen im oder gegen den Uhrzeigersinn zu korrigieren
- Die *Schatten*-Eigenschaften können definiert werden.
- Die Anzeige der ☑ *Kanten* der Symbole kann aktiviert und mit einer *Breite* und *Farbe* versehen werden.

Hinweis

Kombination für beste Wiedergabe von 3D-Daten

Culling-Modus, *Rückseiten hinzufügen* und *Normale invertieren* sind dazu gedacht, das Aussehen von 3D-Daten zu korrigieren, wenn sie nicht richtig dargestellt werden. Normalerweise ist es beim Laden von Daten am besten, zuerst `Culling-Modus = Rückseite` und `Rückseiten hinzufügen = deaktiviert` zu probieren - das ist am effizientesten. Wenn das Rendering nicht korrekt aussieht, versuchen Sie `Rückseiten hinzufügen = aktiviert` und behalten Sie `Culling-Modus = kein Culling` bei. Andere Kombinationen sind fortgeschrittener und nur in bestimmten Szenarien nützlich, je nachdem, wie fehlerhaft der Eingabedatensatz ist.

10.4.4 Schattierung der Textur

Die Schattierung hilft Ihnen, 3D-Details von Objekten sichtbar zu machen, die sonst durch die Beleuchtung der Szene verdeckt werden könnten. Letztendlich ist es ein einfacheres Hilfsmittel, mit dem Sie arbeiten können, da Sie sich nicht um die richtige Beleuchtung der Szene kümmern müssen, um Objekte zu visualisieren.

In QGIS werden verschiedene Schattierungstechniken verwendet, deren Verfügbarkeit vom Geometrietyp des Symbols abhängt:

- *Realistisch (Phong)*: Beschreibt die Art und Weise, wie eine Oberfläche Licht reflektiert, als Kombination aus der *diffusen* Reflexion rauer Oberflächen und der *spiegelnden* Reflexion glänzender Oberflächen (*Glanz*). Es enthält auch eine *Umgebungslicht*-Option, um die geringe Lichtmenge zu berücksichtigen, die über die gesamte Szene gestreut wird. Verwenden Sie den Schieberegler *Transparenz*, um halbtransparente Objekte in 3D zu rendern. Weitere Informationen finden Sie unter Beschreibung der Phong-Reflexion.
- *Realistisch texturiert (Phong)*: wie *Realistisch (Phong)* mit dem Unterschied, dass ein Bild als *Diffuse Textur* verwendet wird. Das Bild kann eine Datei auf der Festplatte, eine Remote-URL oder *im Projekt eingebettet sein*. Die Angaben *Texturmaßstab* und *Texturdrehung* sind erforderlich. Verwenden Sie den Schieberegler *Deckkraft*, um halbtransparente Objekte in 3D darzustellen.
- *CAD (Gooch)*: Diese Technik ermöglicht es, Schattierungen nur in Mitteltönen zu erzeugen, sodass Kantenlinien und Glanzlichter optisch hervorgehoben bleiben. Zusammen mit den Optionen *Diffus*, *Spiegelung* und *Glanz* müssen Sie eine *Warme* Farbe (für die dem Licht zugewandte Oberfläche) und eine *Kalte* Farbe (für die vom Licht abgewandte Oberfläche) angeben. Außerdem werden die relativen Beiträge der diffusen Farbe zu den kühlen und warmen Farben durch die Eigenschaften *Alpha* bzw. *Beta* gesteuert. Siehe auch Gooch-Schattierung.
- *Metallrauhigkeit*: Ein physikalisch basiertes Rendering-Material, das eine genaue Darstellung der Wechselwirkung von Licht mit Oberflächen liefert. Es stehen Optionen zur Einstellung des Materials *Grundfarbe*, *Metalität* und *Rauheit* zur Verfügung.
- *Eingebettete Texturen* mit 3D-Form-Modellen

10.4.5 Anwendungsbeispiel

Die oben erläuterten Einstellungen können Sie unter https://app.merginmaps.com/projects/saber/luxembourg/tree einsehen.

KAPITEL 11

Verwaltung von Datenquellen

11.1 Öffnen von Daten

Als Teil eines Open-Source-Software-Ökosystems baut QGIS auf verschiedenen Bibliotheken auf, die in Kombination mit den eigenen Funktionen die Fähigkeiten bieten, viele Formate zu lesen und oft auch zu schreiben:

- Zu den Vektordatenformaten gehören GeoPackage, GML, GeoJSON, GPX, KML, kommagetrennte Werte (CSV), ESRI-Formate (Shapefile, Geodatabase…), MapInfo- und MicroStation-Dateiformate, AutoCAD DWG/DXF, GRASS und viele mehr… Die vollständige Liste der unterstützten Vektorformate finden Sie unter **`https://gdal.org/en/latest/drivers/vector/index.html`__**.
- Zu den Rasterdatenformaten gehören GeoTIFF, JPEG, ASCII Gridded XYZ, MBTiles, R- oder Idrisi-Raster, GDAL Virtual, SRTM, Sentinel-Daten, ERDAS IMAGINE, ArcInfo Binary Grid, ArcInfo ASCII Grid und viele mehr… Die vollständige Liste der unterstützten Rasterformate finden Sie unter **`https://gdal.org/en/latest/drivers/raster/index.html`__**.
- Database formats include PostgreSQL, SQLite/SpatiaLite, Oracle, MS SQL Server, SAP HANA, MySQL…
- Webkarten- und Datendienste (WM(T)S, WFS, WCS, CSW, XYZ- Tiles, ArcGIS-Services, …) werden ebenfalls von QGIS-Providern verarbeitet. Siehe *Arbeiten mit OGC / ISO-Protokollen* für weitere Informationen über einige dieser Dienste.
- Sie können unterstützte Dateien aus archivierten Ordnern (ZIP, …) lesen und QGIS-eigene Formate wie QML-Dateien (*QML - Das QGIS-Stil-Dateiformat*) und virtuelle und Speicher-Layer verwenden.

Mehr als 80 Vektor- und 140 Rasterformate werden von GDAL und den nativen Funktionen von QGIS unterstützt.

Bemerkung

Aus verschiedenen Gründen könnten unter Umständen nicht alle der aufgelisteten Formate in QGIS funktionieren. Zum Beispiel benötigen einige externe, proprietäre Bibliotheken, oder die GDAL/OGR Installation Ihres Betriebssystems unterstützt nicht das Format, das Sie verwenden möchten. Um die Liste der verfügbaren Formate zu sehen, führen Sie in der Kommandozeile `ogrinfo --formats` (für Vektor) und `gdalinfo --formats` (für Raster) aus, oder überprüfen Sie das Menü *Settings* ► *Options* ► *GDAL* in QGIS.

In QGIS gibt es je nach Datenformat verschiedene Werkzeuge zum Öffnen eines Datensatzes, die hauptsächlich im Menü *Layer* ► *Layer hinzufügen* ► oder in der *Layerverwaltungswerkzeugleiste* (zu aktivieren über das Menü *Ansicht* ► *Werkzeugleisten*) verfügbar sind. Alle diese Werkzeuge verweisen auf einen Dialog, die *Datenquellenverwaltung*, den Sie mit der Schaltfläche Datenquellenverwaltung öffnen in der *Datenquellenverwaltungsleiste* oder durch Drücken von `Strg+L` öffnen können. Der Dialog *Datenquellenverwaltung* (Abb. 11.1) bietet eine einheitliche Schnittstelle zum Öffnen von dateibasierten Daten sowie von Datenbanken oder Webservices, die von QGIS unterstützt werden.

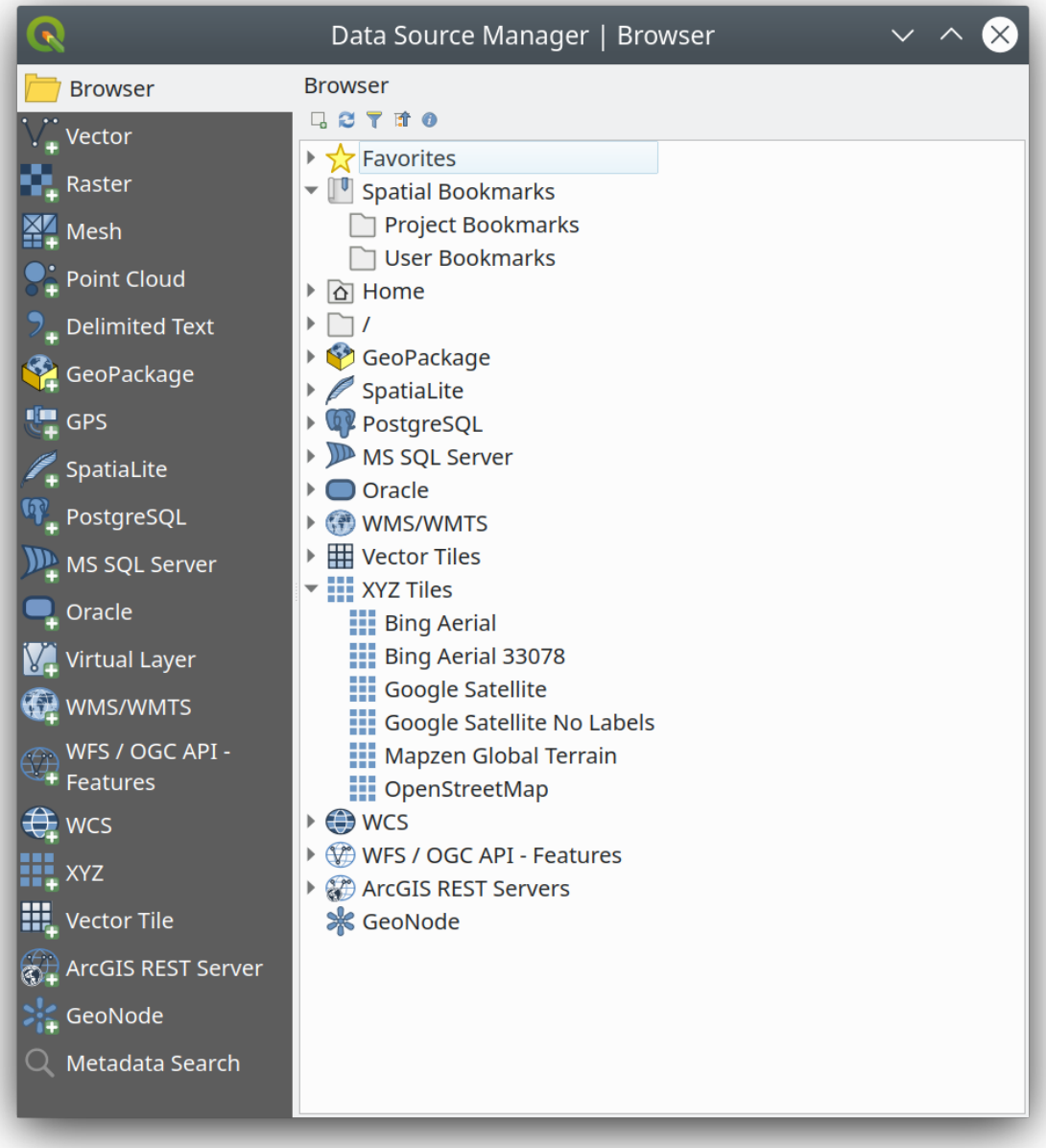


Abb. 11.1: Der QGIS Datenverwaltungs-Dialog

Neben diesem Haupteinstiegspunkt gibt es auch das Plugin *DB Manager*, das erweiterte Möglichkeiten zur Analyse und Manipulation von verbundenen Datenbanken bietet. Weitere Informationen zu den DB-Manager-Funktionen finden Sie in *DB Verwaltung*.

Es gibt viele andere Tools, native oder Drittanbieter-Plugins, mit denen Sie verschiedene Datenformate öffnen können.

In diesem Kapitel werden nur die Werkzeuge beschrieben, die standardmäßig in QGIS zum Laden von Daten zur

Verfügung stehen. Es wird sich hauptsächlich auf den *Datenquellenverwaltung*-Dialog konzentrieren, aber über die Beschreibung der einzelnen Reiter hinaus werden auch die Werkzeuge auf der Grundlage des Datenanbieters oder der Formatbesonderheiten untersucht.

11.1.1 Das Browser-Bedienfeld

Der *Browser* ist eine der wichtigsten Möglichkeiten, Ihre Daten schnell und einfach zu Projekten hinzuzufügen. Er ist verfügbar als:

- einen Reiter *Datenquellenverwaltung*, der durch Drücken der Schaltfläche Datenquellenverwaltung öffnen (`Strg+L`) aktiviert wird;
- als QGIS Bedienfeld können Sie über das Menü *Ansicht* ► *Bedienfelder* (oder *Einstellungen* ► *Bedienfelder*) oder durch Drücken von `Strg+2` öffnen.

In beiden Fällen hilft Ihnen der *Browser* bei der Navigation in Ihrem Dateisystem und der Verwaltung von Geodaten, unabhängig vom Typ des Layers (Raster, Vektor, Tabelle) oder dem Format der Datenquelle (einfache oder komprimierte Dateien, Datenbanken, Webdienste).

Erkundung der Schnittstelle

Oben im Bedienfeld des Browsers finden Sie einige Schaltflächen, die Ihnen dabei helfen:

- Ausgewählte Layer hinzufügen: Sie können auch Daten zur Kartenansicht hinzufügen, indem Sie **Gewählte(n) Layer dem Projekt hinzufügen** aus dem Kontextmenü des Layers wählen;
- Aktualisieren des Browser-Baums;
- Browser filtern, um nach bestimmten Daten zu suchen. Geben Sie ein Suchwort oder einen Platzhalter ein und der Browser filtert den Baum so, dass nur Pfade zu passenden DB-Tabellen, Dateinamen oder Ordnern angezeigt werden. Es kann festgelegt werden, ob der Vergleich zwischen Groß- und Kleinschreibung unterscheiden soll oder nicht. Darüber hinaus zwischen den folgenden Such-Optionen gewählt werden:
  - *Normal*: Einträge anzeigen, die den Suchtext enthalten
  - *Platzhalter*: Verfeinern Sie die Suche, indem Sie die Zeichen `?` und/oder `*` verwenden, um die Position des Suchtextes zu bestimmen
  - *Reguläre Ausdrücke*
- Alle einklappen reduziert alle Einträge des Browser-Baums auf die oberste Ebene;
- Eigenschaftenelement ein-/ausschalten: wenn eingeschaltet, wird ein neues Widget am unteren Rand des Bedienfelds hinzugefügt, das, falls zutreffend, Metadaten für das ausgewählte Element anzeigt.

Die Einträge im Bedienfeld *Browser* sind hierarchisch geordnet, und es gibt mehrere Einträge der obersten Ebene:

1. *Favoriten*, wo Sie Verknüpfungen zu häufig genutzten Orten ablegen können
2. *Spatial Bookmarks*, wo Sie häufig verwendete Kartenausschnitte speichern können (siehe *Räumliche Lesezeichen*)
3. *Projektverzeichnis*: für einen schnellen Zugriff auf den Ordner, in dem (die meisten) Daten zu Ihrem Projekt gespeichert sind. Der Standardwert ist das Verzeichnis, in dem sich Ihre Projektdatei befindet.
4. *Home* Verzeichnis im Dateisystem und das Wurzelverzeichnis des Dateisystems.
5. Angeschlossene lokale oder Netz-Laufwerke
6. Hinzu kommt eine Reihe von Container-/Datenbanktypen und Dienstprotokollen, abhängig von Ihrer Plattform und den zugrunde liegenden Bibliotheken:

- *GeoPackage*
- *SpatiaLite*
- *PostgreSQL*
- *SAP HANA*
- *MS SQL Server*
- *Oracle*
- *WMS/WMTS*
- *Vector Tiles*
- *XYZ Tiles*
- *WCS*
- *WFS/OGC API-Features*
- *ArcGIS REST Server*

Interaktion mit den Browser-Elementen

Der Browser unterstützt Drag & Drop innerhalb des Browsers, vom Browser in die Kartenansicht und in das *Layer* Bedienfeld und auch vom *Layer* Bedienfeld zu Layer-Containern im Browser (z.B. GeoPackage).

QGIS-Projektdateien (*.qgs, *.qgz) können innerhalb des Browsers aufgeklappt werden und zeigen dann alle im Projekt enthaltenen Layer (einschließlich Gruppen) an. Diese Projektinhalte werden wie jedes andere Element im Browser behandelt, d. h. sie können innerhalb des Browsers gezogen und abgelegt werden (z. B. um ein Layer-Element in eine Geopaketdatei zu kopieren) oder dem aktuellen Projekt durch Ziehen und Ablegen oder Doppelklicken hinzugefügt werden.

Ein Kontextmenü kann für jedes Element im *Browser* Bedienfeld durch einen Rechtsklick auf das Element geöffnet.

Für Dateisystem-Verzeichniseinträge bietet das Kontextmenü folgende Möglichkeiten:

- *Neu* ►, um im ausgewählten Eintrag ein neues Element zu erstellen:
  - *Ordner…*
  - *GeoPackage…*
  - *Shapefile…*
- *Als Favorit hinzufügen*: Favoritenordner können jederzeit umbenannt (*Favorit umbenennen…*) oder entfernt (*Favorit entfernen*) werden.
- *Im Browser verstecken*: Der Eintrag wird zukünftig nicht mehr angezeigt; versteckte Einträge können über die Einstellung *Verborgene Elemente* wieder sichtbar gemacht werden
- *Verzeichnis schnell durchsuchen*
- *Verzeichnis öffnen*
- *In Terminal öffnen*
- *Eigenschaften…*
- *Verzeichniseigenschaften…*

Für Einträge im Verzeichnisbaum, die als Layer im Projekt fungieren können, enthält das Kontextmenü unterstützende Einträge. Zum Beispiel für nicht datenbank- und nicht dienstbasierte Vektor-, Raster- und Netzdatenquellen:

- *Layer exportieren ► in Datei…*
- *Layer zu Projekt hinzufügen*
- *Layereigenschaften*
- *Mit Datenquellenverwaltung öffnen…*
- *Verwalte ► „<name of file>“…umbenennen* oder *Lösche „<name of file>“…*
- *In Dateien anzeigen*
- *Dateieigenschaften*

Im Eintrag *Layereigenschaften* finden Sie (ähnlich wie im Dialog Layer-Eigenschaften *Vektor* und *Raster*, sobald die Layer dem Projekt hinzugefügt wurden):

- *Metadaten* für den Layer. Metadaten-Gruppen: *Information vom Anbieter* (wenn möglich, wird in *Pfad* ein Hyperlink zur Quelle erstellt), *Identifikation*, *Ausdehnung*, *Zugriff*, *Felder* (für Vektor-Layer), *Bänder* (für Raster-Layer), *Kontakte*, *Links* (für Vektor-Layer), *Referenzen* (für Raster-Layer), *Protokoll*.
- Ein *Vorschau* Feld
- Die Attributtabelle für Vektorquellen (im Reiter *Attribute*).

Verwenden Sie *Mit Datenquellenverwaltung öffnen…*, um die Datenquelle direkt in der *Datenquellenverwaltung* mit der URI der Datenquelle zu öffnen und zu konfigurieren. Dies vereinfacht das Hinzufügen einer Ebene aus dem *Browser*, da Sie spezifische Öffnungsoptionen für die Datenquelle festlegen können. Diese Funktion ist derzeit für Vektor- (einschließlich des dedizierten GeoPackage-Eintrags), Raster- und SpatiaLite-Datenquellen verfügbar.

Hinzufügen eines Layers zum Projekt unter Verwendung des *Browser*:

1. Aktivieren Sie das *Browser* wie oben beschrieben. Es wird ein Browser-Baum mit Ihrem Dateisystem, Datenbanken und Webdiensten angezeigt. Möglicherweise müssen Sie Datenbanken und Webdienste verbinden, bevor sie angezeigt werden (siehe entsprechende Abschnitte).
2. Suchen Sie den Layer in der Liste.
3. Verwenden Sie das Kontextmenü, doppelklicken Sie auf den Namen des Layers oder ziehen Sie ihn in die *Kartenansicht*. Ihr Layer wird nun zum :ref:`Bedienfeld <label_legend> ` hinzugefügt und kann in der Kartenansicht angezeigt werden.

> **Tipp**
>
> **Ein QGIS-Projekt direkt vom Browser öffnen**
>
> Sie können ein QGIS-Projekt auch direkt aus dem Bedienfeld des Browsers öffnen, indem Sie auf seinen Namen doppelklicken oder es per Drag-and-Drop in die Kartenansicht ziehen.

Sobald eine Datei geladen ist, können Sie mit den Navigationswerkzeugen in der Karte zoomen. Um den Stil eines Layers zu ändern, öffnen Sie den Dialog *Eigenschaften des Layers*, indem Sie auf den Namen des Layers doppelklicken oder indem Sie mit der rechten Maustaste auf den Namen in der Legende klicken und *Eigenschaften* aus dem Kontextmenü wählen. Weitere Informationen zum Einstellen der Symbologie für Vektorebenen finden Sie im Abschnitt *Symbolisierung*.

Wenn Sie mit der rechten Maustaste auf ein Element in der Browserstruktur klicken, können Sie dies tun:

- nach einer Datei oder einer Tabelle suchen, ihre Metadaten anzeigen oder sie in Ihrem Projekt öffnen. Tabellen können sogar umbenannt, gelöscht oder gekürzt werden.
- für einen Ordner, setzen Sie ein Lesezeichen für ihn in Ihren Favoriten oder blenden Sie ihn aus der Browserstruktur aus. Versteckte Ordner können über den Reiter *Einstellungen ► Optionen ► Datenquellen* verwaltet werden.
- Verwalten Sie Ihre räumlichen Lesezeichen <sec_bookmarks> `: Lesezeichen können als ``XML`-Dateien erstellt, exportiert und importiert werden.
- eine Verbindung zu einer Datenbank oder einem Webdienst herstellen.

- ein Schema aktualisieren, umbenennen oder löschen.

Sie können auch Dateien in Datenbanken importieren oder Tabellen von einem Schema/einer Datenbank in eine andere kopieren, indem Sie sie einfach per Drag-and-Drop ziehen. Es gibt ein zweites Bedienfeld im Browser, um langes Scrollen beim Ziehen zu vermeiden. Wählen Sie einfach die Datei aus und ziehen Sie sie per Drag-and-Drop von einem Bedienfeld in das andere.

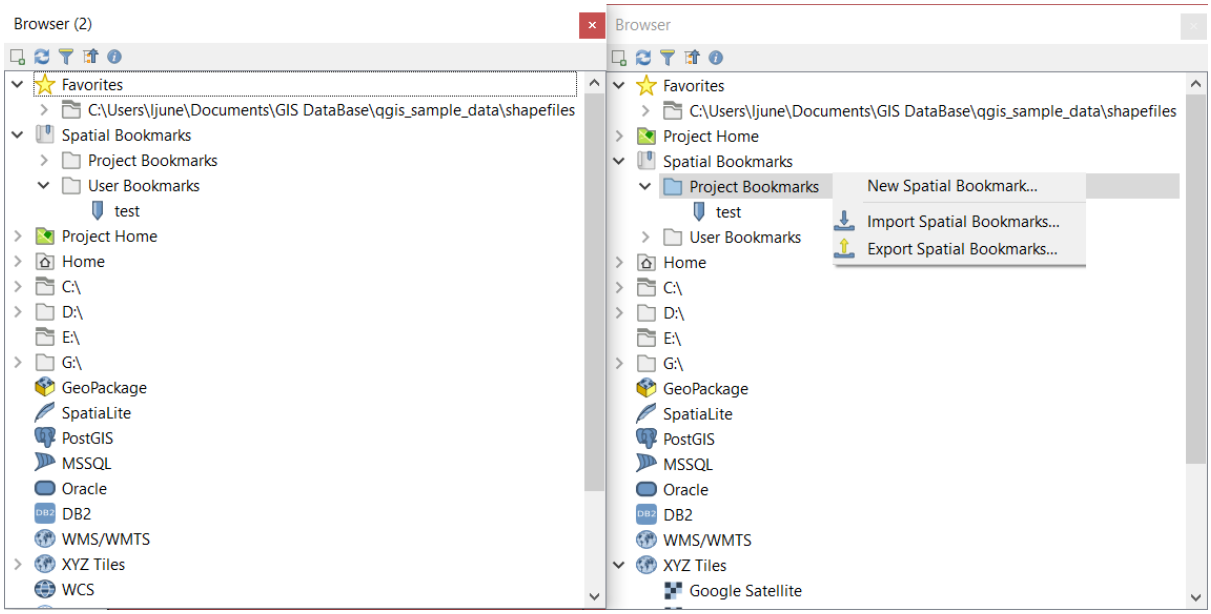


Abb. 11.2: QGIS Browser Bedienfelder nebeneinander

Tipp

Hinzufügen von Layern zu QGIS durch einfaches Ziehen und Ablegen aus Ihrem OS-Dateibrowser

Sie können dem Projekt auch Dateien hinzufügen, indem Sie sie per Drag-and-Drop aus dem Dateibrowser Ihres Betriebssystems auf das Bedienfeld *Layers* oder die Kartenansicht ziehen.

11.1.2 Die DB-Verwaltung

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.

Die DB-Verwaltung bietet verschiedene Funktionen:

- eine Verbindung zu Datenbanken herstellen und deren Struktur und Inhalt anzeigen
- Vorschau von Datenbanktabellen
- Layer zur Kartenansicht hinzufügen, entweder durch Doppelklick oder durch Ziehen und Ablegen.
- Hinzufügen von Layern zu einer Datenbank aus dem QGIS Browser oder aus einer anderen Datenbank
- SQL-Abfragen erstellen und ihre Ausgabe in die Kartenansicht einfügen
- *virtuelle Layer* erstellen

Weitere Informationen zu den Fähigkeiten des DB-Managers finden Sie unter *DB Verwaltung*.

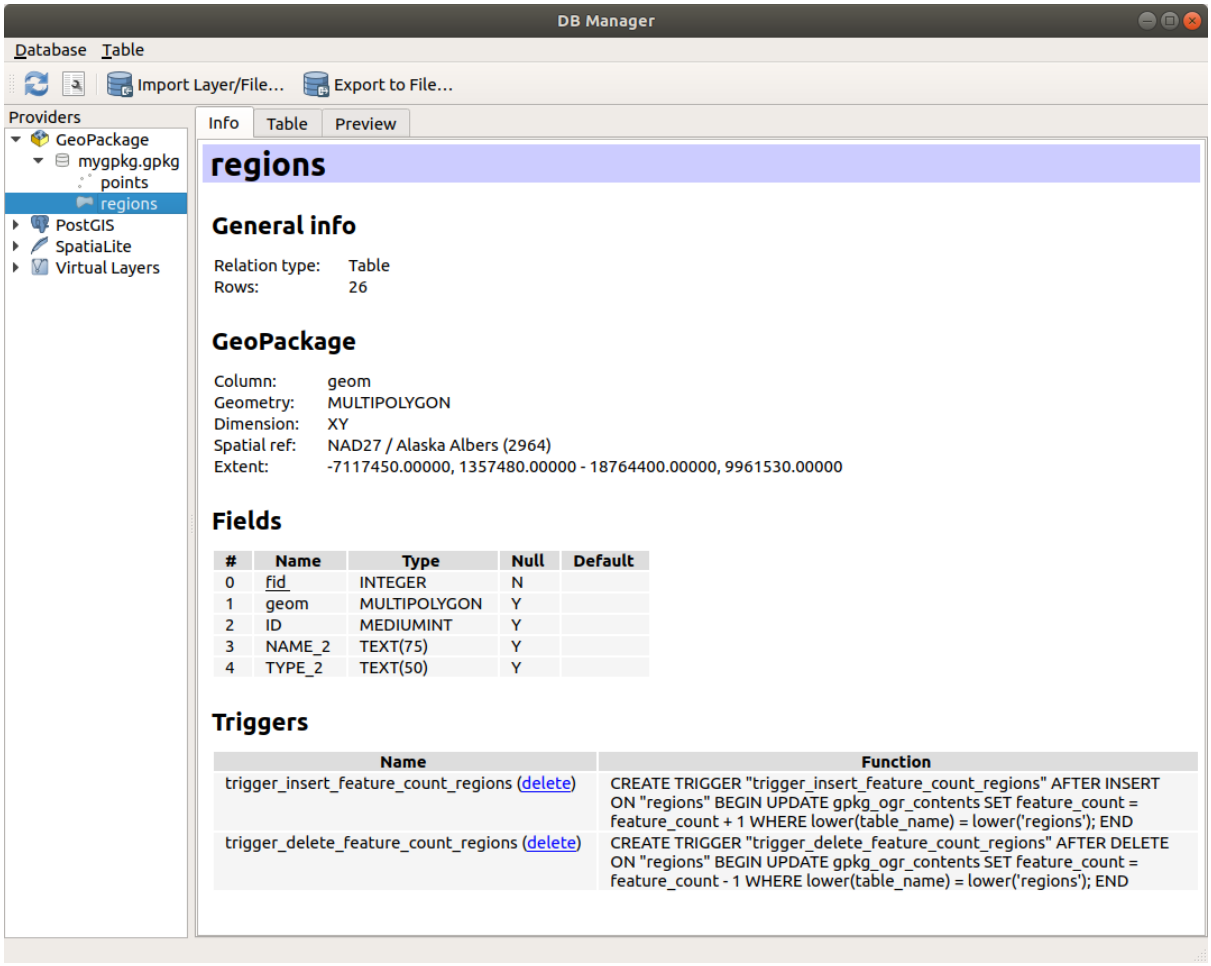


Abb. 11.3: Die DB-Verwaltung

11.1.3 Anbieterbasierte Ladetools

Neben dem Browser-Bedienfeld und der DB-Verwaltung, den beiden wichtigsten Werkzeugen, die QGIS zum Hinzufügen von Layern bereitstellt, finden Sie auch Werkzeuge, die speziell auf einzelne Datenanbieter zugeschnitten sind.

> **Bemerkung**
>
> Einige *externe Plugins* bieten auch Werkzeuge zum Öffnen bestimmter Dateiformate in QGIS.

Laden eines Layers aus einer Datei

So laden Sie einen Layer aus einer Datei:

1. Öffnen Sie den Reiter für den Layer-Typ im Dialogfeld *Datenquellenverwaltung*, d. h. klicken Sie auf die Schaltfläche Datenquellenverwaltung öffnen (oder drücken Sie `Strg+L`) und aktivieren Sie den Reiter Ziel oder:
   - für Vektordaten (wie GML-, ESRI Shapefile-, Mapinfo- und DXF-Layer): Drücken Sie `Strg+Umschalt+V`, wählen Sie die Menüoption *Layer* ► *Layer hinzufügen* ► *Vektorlayer hinzufügen* oder klicken Sie auf die Schaltfläche Vektorlayer hinzufügen der Werkzeugleiste.

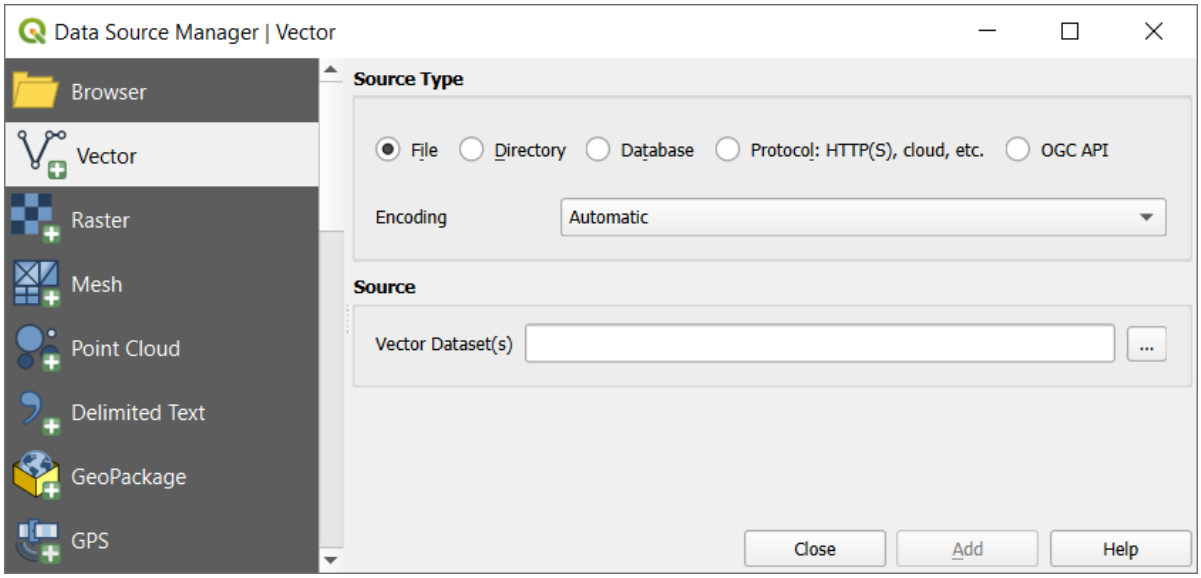


Abb. 11.4: Dialogfeld Vektor-Layer hinzufügen

- für Rasterdaten (wie GeoTiff-, MBTiles-, GRIdded Binary- und DWG-Layer): drücken Sie `Strg+Umschalt+R`, wählen Sie die Menüoption *Layer* ► *Layer hinzufügen* ► *Raster-Layer hinzufügen* oder klicken Sie auf die Schaltfläche Raster-Layer hinzufügen der Werkzeugleiste.

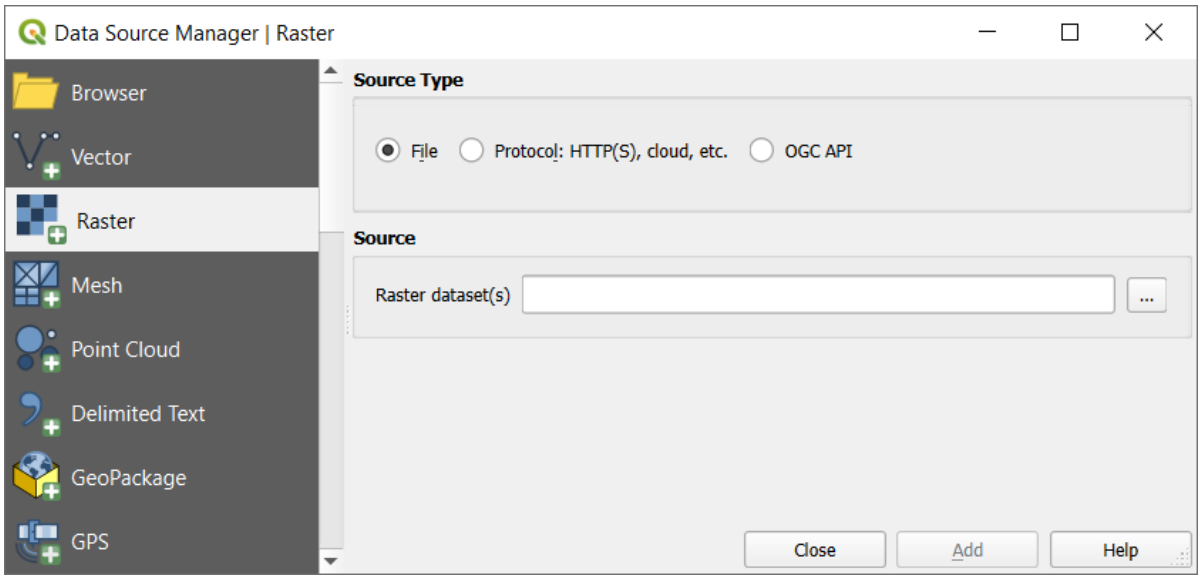


Abb. 11.5: Dialogfeld Raster-Layer hinzufügen

2. *Quelltyp* auswählen
3. Klicken Sie auf die Schaltfläche … Durchsuchen
4. Navigieren Sie im Dateisystem und laden Sie eine unterstützte Datenquelle. Mehrere Layer können gleichzeitig geladen werden, indem Sie die Taste `Strg` gedrückt halten und auf mehrere Elemente im Dialogfeld klicken oder die Taste `Umschalt` gedrückt halten, um einen Bereich von Elementen auszuwählen, indem Sie auf das erste und das letzte Element des Bereichs klicken. Nur Formate, die gut getestet wurden, erscheinen im Formatfilter. Andere Formate können durch Auswahl von `Alle Dateien` (der oberste Punkt im Pull-Down-Menü) geladen werden.

5. Drücken Sie *Öffnen*, um die ausgewählte Datei in das Dialogfeld *Datenquellenverwaltung* zu laden.

   Je nach ausgewähltem Layertyp stehen zusätzliche *Optionen* (Kodierung, Geometrietyp, Tabellenfilterung, Datei-Locking, Datenformatierung …) zur Konfiguration zur Verfügung. Diese Optionen werden ausführlich in der spezifischen GDAL-Treiberdokumentation Vektor oder Raster beschrieben. Oben in den Optionen führt ein Text mit Hyperlink direkt zur Dokumentation des entsprechenden Treibers für das ausgewählte Dateiformat.

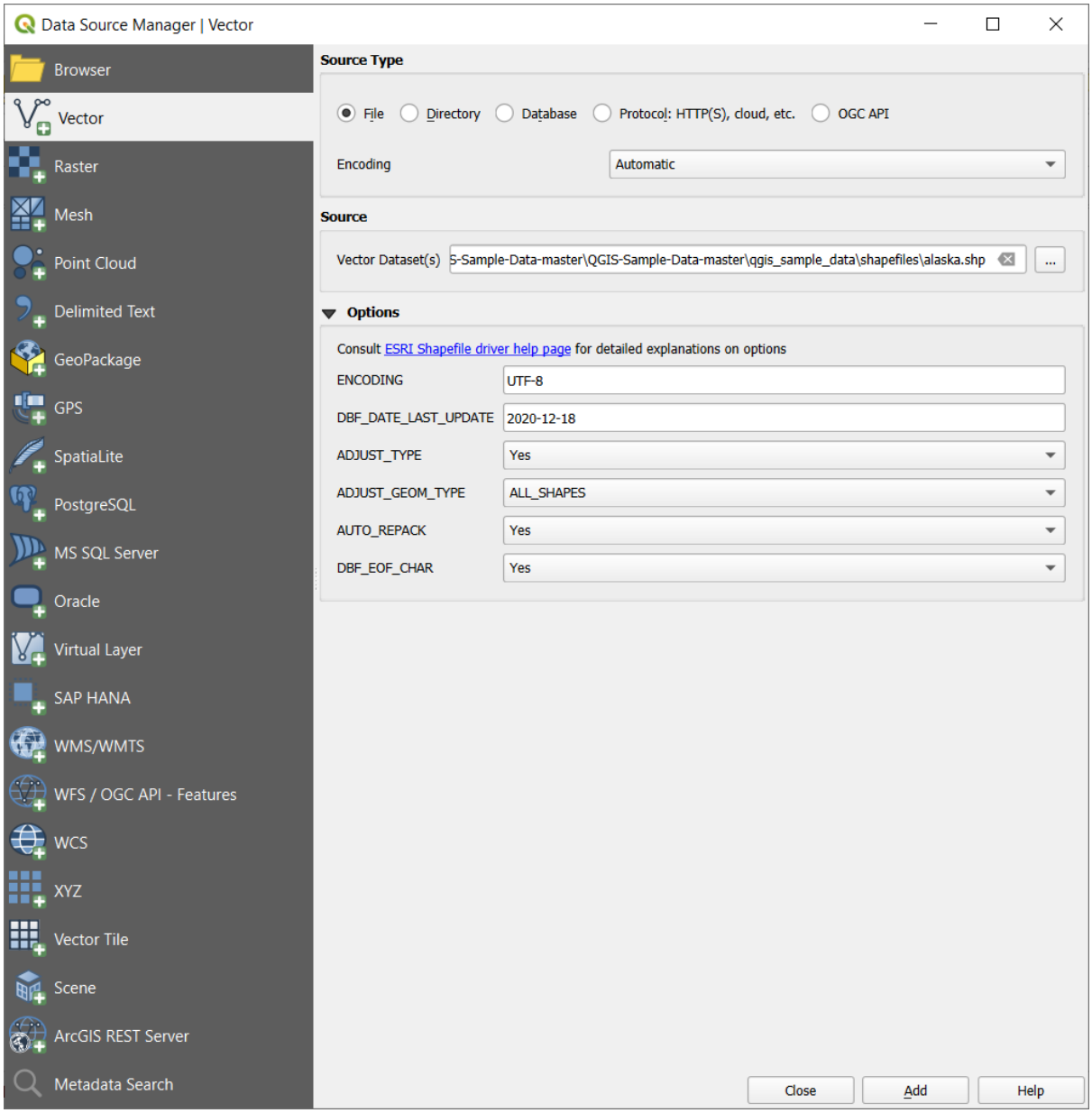


Abb. 11.6: Laden eines Shapefiles mit Optionen zum Öffnen

6. Drücken Sie *Hinzufügen*, um die Datei in QGIS zu laden und sie in der Kartenansicht anzuzeigen. Wenn Sie Vektordatensätze mit mehreren Layern hinzufügen, erscheint der Dialog *Elemente zum Hinzufügen auswählen*. In diesem Dialog können Sie die einzelnen Layer Ihres Datensatzes auswählen, die Sie hinzufügen möchten. Außerdem können Sie unter *Optionen* Folgendes auswählen:

   - ☑ *Layer einer Gruppe hinzufügen*
   - ☑ *Anzeigen von System- und internen Tabellen*

- *Leere Vektor-Layer anzeigen.*

Abb. 11.7 zeigt QGIS nach dem Laden der Datei `alaska.shp`.

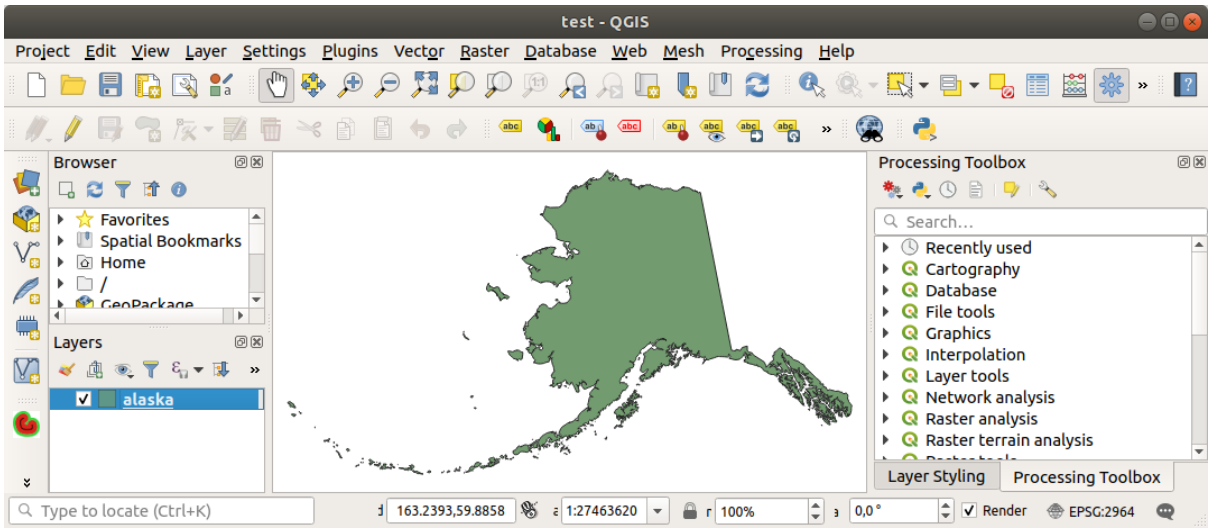


Abb. 11.7: QGIS mit geladenem Shapefile von Alaska

Bemerkung

Da einige Formate wie MapInfo (z.B. `.tab`) oder Autocad (`.dxf`) es erlauben, verschiedene Geometrietypen in einer einzigen Datei zu mischen, öffnet sich beim Laden solcher Datensätze ein Dialog zur Auswahl der zu verwendenden Geometrien, um eine Geometrie pro Layer zu haben.

Die Reiter Vektor-Layer hinzufügen und Raster-Layer hinzufügen ermöglichen das Laden von Layern aus anderen Quellentypen als *Datei*:

- Sie können spezifische Vektorformate laden, wie `ArcInfo Binary Coverage`, `UK. National Transfer Format`, sowie das rohe TIGER-Format des `US Census Bureau` oder `OpenfileGDB`.

  Dazu wählen Sie *Verzeichnis* als *Quellentyp*. In diesem Fall kann ein Verzeichnis im Dialog nach Drücken von … Durchsuchen ausgewählt werden.

- Mit dem Quellentyp `RadioButtonOn` *Datenbank*` können Sie eine bestehende Datenbankverbindung auswählen oder eine Verbindung zu dem gewählten Datenbanktyp erstellen. Einige mögliche Datenbanktypen sind `ODBC`, `Esri Personal Geodatabase`, `MS SQL Server` sowie `PostgreSQL` oder `MySQL`.

  Pressing the *New* button opens the *Create a New OGR Database Connection* dialog whose parameters are among the ones you can find in *Erstellen einer gespeicherten Verbindung*. Pressing *Open* lets you select from the available tables.

- Der Quellentyp *Protokoll: HTTP(S), Cloud, etc.* öffnet lokal oder im Netzwerk gespeicherte Daten, die entweder öffentlich zugänglich sind oder in privaten Buckets von kommerziellen Cloud-Speicherdiensten. Unterstützte Protokolltypen sind:

  – `HTTP/HTTPS/FTP`, mit einer *URI* und, falls erforderlich, einer *Authentifizierung*.

  – Cloud-Speicher wie „AWS S3“, „Google Cloud Storage“, „Microsoft Azure Blob“, „Microsoft Azure Data Lake Storage“, „Alibaba OSS Cloud“ und „Open Stack Swift Storage“ unterstützen die direkte Steuerung über VSI *Credential Options* beim Hinzufügen von OGR-Vektor- oder GDAL-Rasterebenen. Sie müssen zunächst den *Bucket oder Container* und den *Objektschlüssel* eingeben. Danach können Sie die erforderlichen *Anmeldeoptionen* hinzufügen.

Wenn Sie OGR-Vektor- oder GDAL-Rasterebenen aus den cloudbasierten Protokollen hinzufügen, können Sie auch zusätzliche *Anmeldeoptionen* für diesen bestimmten Treiber und Bucket festlegen. Wenn Anmeldeoptionen in der URI des Layers gefunden werden, werden sie ebenfalls automatisch festgelegt. Auf diese Weise können verschiedene Layer unterschiedliche Anmeldedaten verwenden.

- Dienst, der OGC `WFS 3` (noch experimentell) unterstützt, das `GeoJSON` oder `GEOJSON - Newline Delimited` Format verwendet oder auf der `CouchDB` Datenbank basiert. Eine *URI* ist erforderlich, mit optionaler *Authentifizierung*.
- Für alle Vektorquellentypen ist es möglich, das *Kodierung* zu definieren oder die Einstellung *Automatisch* ► zu verwenden.

- Der Quellentyp *OGC API* ermöglicht Ihnen den Zugriff auf Vektor- und Raster- Daten von Servern, die die OGC API-Standards implementieren. So verwenden Sie diese Option:
  1. Wählen Sie im Dialogfeld „Datenquellenverwaltung“ die Option „OGC API“ aus.
  2. Geben Sie den Endpunkt des OGC-API-Dienstes ein, mit dem Sie sich verbinden möchten. Beachten Sie, dass Sie dem Endpunkt nicht „OGCAPI:“ voranstellen müssen.
  3. Klicken Sie auf *Verbinden*, um eine Verbindung zum Server herzustellen.

Einen Netz-Layer laden

Ein Netz-Layer ist ein unstrukturiertes Gitter, das in der Regel zeitliche und andere Komponenten enthält. Die räumliche Komponente enthält eine Sammlung von Eckpunkten, Kanten und Flächen im 2D- oder 3D-Raum. Weitere Informationen zu Mesh Layern finden Sie unter *Working with Mesh Data*.

Um einen Mesh Layer zu QGIS hinzuzufügen:

1. Öffnen Sie den Dialog *Datenquellenverwaltung*, indem Sie ihn entweder aus dem Menü *Layer* ► auswählen oder auf die Schaltfläche Datenquellenverwaltung öffnen klicken.
2. Aktivieren Sie den Reiter *Netz* auf dem linken Bedienfeld
3. Drücken Sie die Taste … Durchsuchen, um die Datei auszuwählen. Es werden *verschiedene Formate* unterstützt.
4. Wählen Sie die Datei und drücken Sie *Hinzufügen*. Der Layer wird unter Verwendung des nativen Netz-Renderings hinzugefügt.
5. Wenn die ausgewählte Datei mehrere Netz-Layer enthält, werden Sie in einem Dialog aufgefordert, die zu ladenden Sublayer auszuwählen. Treffen Sie Ihre Auswahl und drücken Sie *OK* und die Layer werden mit dem nativen Netz-Rendering geladen. Es ist auch möglich, sie innerhalb einer Gruppe zu laden.

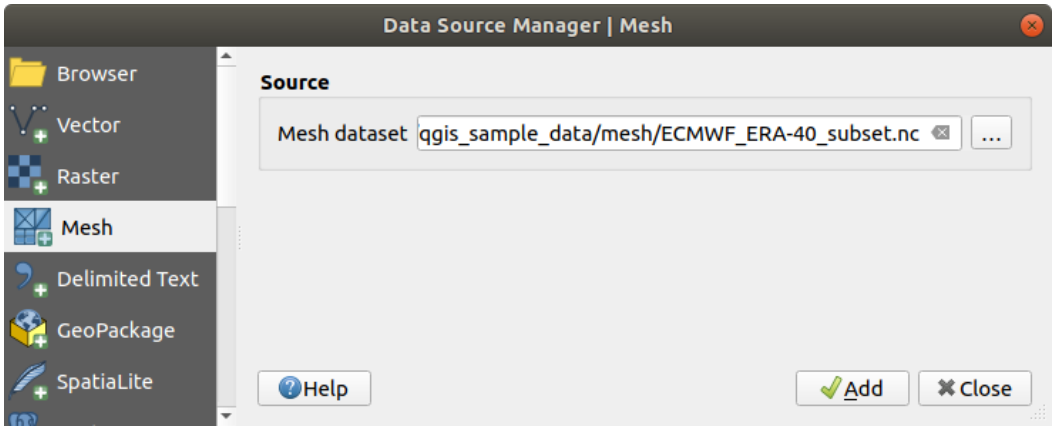


Abb. 11.8: Der Netz-Reiter in der Datenquellenverwaltung

Importieren einer Textdatei (mit Trennzeichen)

Durch Trennzeichen getrennte Textdateien (z.B. `.txt`, `.csv`, `.dat`, `.wkt`) können mit den oben beschriebenen Werkzeugen geladen werden. Auf diese Weise werden sie als einfache Tabellen angezeigt. Manchmal können begrenzte Textdateien Koordinaten / Geometrien enthalten, die Sie visualisieren möchten. Hierfür können Sie den Reiter *Getrennte Texte* verwenden.

1. Klicken Sie auf das Symbol Datenquellenverwaltung öffnen, um den Dialog *Datenquellenverwaltung* zu öffnen
2. Aktivieren Sie den Reiter *Getrennte Text*
3. Wählen Sie die zu importierende Textdatei mit Trennzeichen (z. B. `qgis_sample_data/csv/elevp.csv`) durch Klicken auf die Schaltfläche … Durchsuchen.
4. Konfigurieren Sie die Einstellungen entsprechend Ihrem Datensatz und Ihren Anforderungen, wie unten beschrieben.

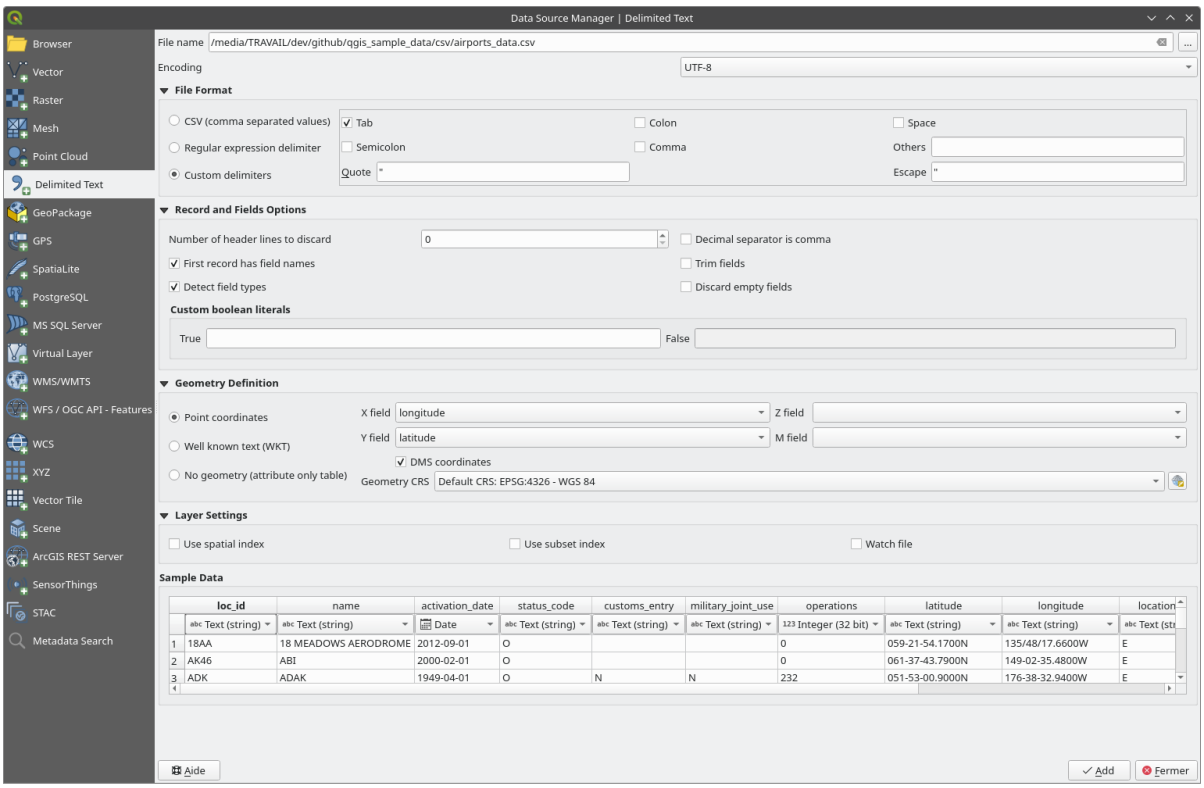


Abb. 11.9: Dialog zum Hinzufügen von getrennten Texten

Dateiformat

Sobald die Datei ausgewählt ist, versucht QGIS, die Datei mit dem zuletzt verwendeten Begrenzungszeichen zu analysieren und Felder und Zeilen zu identifizieren. Damit QGIS die Datei korrekt analysieren kann, ist es wichtig, das richtige Trennzeichen zu wählen. Sie können ein Trennzeichen angeben, indem Sie wählen zwischen:

- *CSV (Komma getrennte Werte)*, um das Kommazeichen zu verwenden.
- *Reguläre Ausdrücke*: tragen Sie in das Feld *Ausdruck* den entsprechenden Text ein. Um zum Beispiel Tabulator als Trennzeichen zu verwenden, tragen Sie `\t` ein (dies wird in regulären Ausdrücken für das Tabulatorzeichen verwendet).

- *Benutzerdefiniert*, Auswahl zwischen einigen vordefinierten Begrenzungszeichen wie `Komma`, `Leerzeichen`, `Tabulator`, *Semikolon*`, ...

Datensatz- und Feldoptionen

Für die Datenerkennung können einige andere praktische Optionen verwendet werden:

- *Anzahl der zu überspringenden Kopfzeilen*: Praktisch, wenn Sie die ersten Zeilen der Datei beim Import vermeiden wollen, entweder weil es sich um Leerzeilen oder um Zeilen mit einer anderen Formatierung handelt.
- *Erster Datensatz hat Feldnamen*: Werte in der ersten Zeile werden als Feldnamen (Spaltenüberschriften) verwendet, ansonsten verwendet QGIS die Feldnamen `Feld_1`, `Feld_2`...
- *Feldtypen bestimmen*: versucht, automatisch den Feldtyp zu erkennen (Text, Datum, etc.). Wenn diese Option nicht markiert ist, werden alle Felder als Text-Feld behandelt.
- *Dezimalseparator ist Komma*: Sie können festlegen, dass das Dezimaltrennzeichen ein Komma ist.
- *Felder abschneiden*: ermöglicht es Ihnen, führende und nachfolgende Leerzeichen aus Feldern zu entfernen.
- *Leere Felder überspringen*.
- *Benutzer-Boolean-Werte*: Ermöglicht das Hinzufügen von benutzerdefinierten Zeichenfolgen, die als boolesche Werte erkannt werden.

Feldtyp-Erkennung

QGIS versucht, die Feldtypen automatisch zu erkennen (es sei denn, das Kontrollkästchen „Feldtypen erkennen“ ist nicht aktiviert), indem es den Inhalt einer optionalen CSVT-Sidecar-Datei (siehe GeoCSV specification) überprüft und die gesamte Datei scannt, um sicherzustellen, dass alle Werte tatsächlich fehlerfrei konvertiert werden können. Der Fallback-Feldtyp ist Text.

Der ermittelte Feldtyp wird unter dem Feldnamen in der Tabelle der Probendatenvorschau angezeigt und kann bei Bedarf manuell geändert werden.

Die folgenden Feldtypen werden unterstützt:

- `Boolean`: Groß-/Kleinschreibung nicht berücksichtigende Literalpaare, die als boolesche Werte interpretiert werden, sind `1`/`0`, `t`/*false*`, `t`/*f*`, `yes`/*no*`
- `Ganz-Zahl (Integer)`
- `Ganz-Zahl (Integer - 64 bit)`
- „Dezimalzahl“: Gleitkommazahl mit doppelter Genauigkeit
- `Datum`
- `Uhrzeit`
- `Datum und Uhrzeit`
- `Text`

Definition der Geometrie

Für die *Geometriedefinition* haben Sie die folgenden Optionen:

- *Punktkoordinaten*: neben einem *X-Feld* und einem *Y-Feld* können Sie optional auch ein *Z-Feld* (für 3-dimensionale Daten) und ein *M-Feld* (für die Messdimension) an, wenn Ihre Daten diese Felder enthalten. Wenn die Koordinaten als Grad/Minuten/Sekunden definiert sind, aktivieren Sie das Kontrollkästchen *GMS-Koordinaten*. Geben Sie das entsprechende *Geometrie KBS* unter Verwendung des KBS wählen Werkzeugs an.
- *Well known text (WKT)* Option, wenn die räumlichen Informationen als WKT vorliegen: Wählen Sie das *Geometriefeld*, das die WKT-Geometrie enthält und wählen Sie den entsprechenden *Geometrietyp* oder lassen Sie QGIS es automatisch erkennen. Geben Sie das entsprechende *Geometrie-KBS* mit dem KBS wählen Werkzeug an.
- Wenn die Datei keine räumlichen Daten enthält, aktivieren Sie *Keine Geometrie (nur Attributtabelle)* und die Daten werden als einfache Tabelle geladen.

Layer-Einstellungen

Zusätzlich können Sie aktivieren:

- *Räumlichen Index benutzen*, um die Leistung der Anzeige und räumlichen Auswahl von Objekten zu verbessern.
- *Untermengenindex benutzen*, um die Leistung von *Untermengenfilter* zu verbessern (wenn in den Layer-Eigenschaften definiert).
- *Datei überwachen*, um auf Änderungen an der Datei durch andere Anwendungen zu achten, während QGIS läuft.

Klicken Sie am Ende auf *Hinzufügen*, um den Layer zur Karte hinzuzufügen. In unserem Beispiel wird ein Punkt-Layer namens `Elevation` zum Projekt hinzugefügt und verhält sich wie jeder andere Layer in QGIS. Dieser Layer ist das Ergebnis einer Abfrage der `.csv`-Quelldatei (und daher mit dieser verknüpft) und würde :ref:`gespeichert <general_saveas>‘ werden müssen, um einen räumlichen Layer auf der Festplatte zu erhalten.

Beispieldaten

Wenn Sie die Eigenschaften einstellen, wird die Beispieldatenvorschau entsprechend den vorgenommenen Einstellungen aktualisiert.

Auch in der Beispieldatentabelle ist es möglich, die automatisch ermittelten Spaltentypen zu überschreiben.

Importieren einer DXF- oder DWG-Datei

`DXF` und `DWG` Dateien können durch einfaches Drag-and-Drop aus dem Bedienfeld zu QGIS hinzugefügt werden. Sie werden aufgefordert, die Sublayer auszuwählen, die Sie dem Projekt hinzufügen möchten. Layer werden mit zufälligen Stileigenschaften hinzugefügt.

> **Bemerkung**
>
> Bei DXF-Dateien, die mehrere Geometrietypen (Punkt, Linie und/oder Polygon) enthalten, wird der Name der Layer als *<Dateiname.dxf> Entität <Geometrietyp>* generiert.

Um die dxf/dwg-Dateistruktur und ihre Symbolik in QGIS beizubehalten, können Sie das spezielle Werkzeug *Projekt ► Import/Export ► Layer aus DWG/DXF importieren…* verwenden, das Ihnen erlaubt:

1. Elemente aus der Zeichnungs-Datei in eine GeoPackage-Datenbank importieren.
2. importierte Elemente zum Projekt hinzufügen.

Im Dialog *DWG/DXF Import*, um den Zeichnungsinhalt zu importieren:

1. Geben Sie den Speicherort der *Quellzeichnung* ein, d.h. die zu importierende DWG/DXF-Zeichnungsdatei.
2. Geben Sie das Koordinatenbezugssystem der Daten in der Zeichnungsdatei an.
3. Geben Sie den Speicherort des *Zielpakets* an, d.h. die GeoPackage-Datei, in der die Daten gespeichert werden. Wenn eine bestehende Datei angegeben wird, wird diese überschrieben.
4. Wählen Sie die Art des Imports von `Blöcken` mit der entsprechenden Auswahlliste:
   - *Blockgeometrien expandieren*: importiert die Blöcke in der Zeichnungsdatei als normale Elemente.
   - *Blockgeometrien expandieren und Einfügepunkte erzeugen*: importiert die Blöcke im Teilbild als normale Elemente und fügt den Einfügepunkt als Punkt-Layer hinzu.
   - *Nur Einfügepunkte erzeugen*: fügt den Blockeinfügepunkt als Punkt-Layer hinzu.
5. Aktivieren Sie das Kontrollkästchen *Kurven verwenden*, um die importierten Layer zu einem `gekrümmten` Geometrietyp zu machen.
6. Verwenden Sie die Schaltfläche *Importieren*, um die Zeichnung in die Ziel-GeoPackage-Datei zu importieren. Die GeoPackage-Datenbank wird automatisch mit dem Inhalt der Zeichnungsdatei befüllt. Je nach Größe der Datei kann dies einige Zeit in Anspruch nehmen.

Nachdem die `.dwg` oder `.dxf` Daten in die GeoPackage Datenbank importiert worden sind, wird der Rahmen in der unteren Hälfte des Dialogs mit der Liste der Layer aus der importierten Datei gefüllt. Dort können Sie auswählen, welche Layer dem QGIS-Projekt hinzugefügt werden sollen:

1. Legen Sie oben einen *Gruppennamen* fest, um die Zeichnungsdateien im Projekt zu gruppieren. Standardmäßig ist dies der Dateiname der Quellzeichnungsdatei.
2. Markieren Sie Layer, die angezeigt werden sollen: Jeder ausgewählte Layer wird zu einer Ad-hoc-Gruppe hinzugefügt, die Vektorlayer für die Punkt-, Linien-, Beschriftungs- und Flächenobjekte des Zeichnungslayers enthält. Der Stil der Layer ähnelt dem Aussehen, das sie ursprünglich in *CAD hatten.
3. Wählen Sie, ob der Layer beim Öffnen sichtbar sein soll.
4. Wenn Sie die Option *Layer zusammenführen* aktivieren, werden alle Layer in einer einzigen Gruppe zusammengefasst.
5. Drücken Sie *OK*, um die Layer in QGIS zu öffnen.

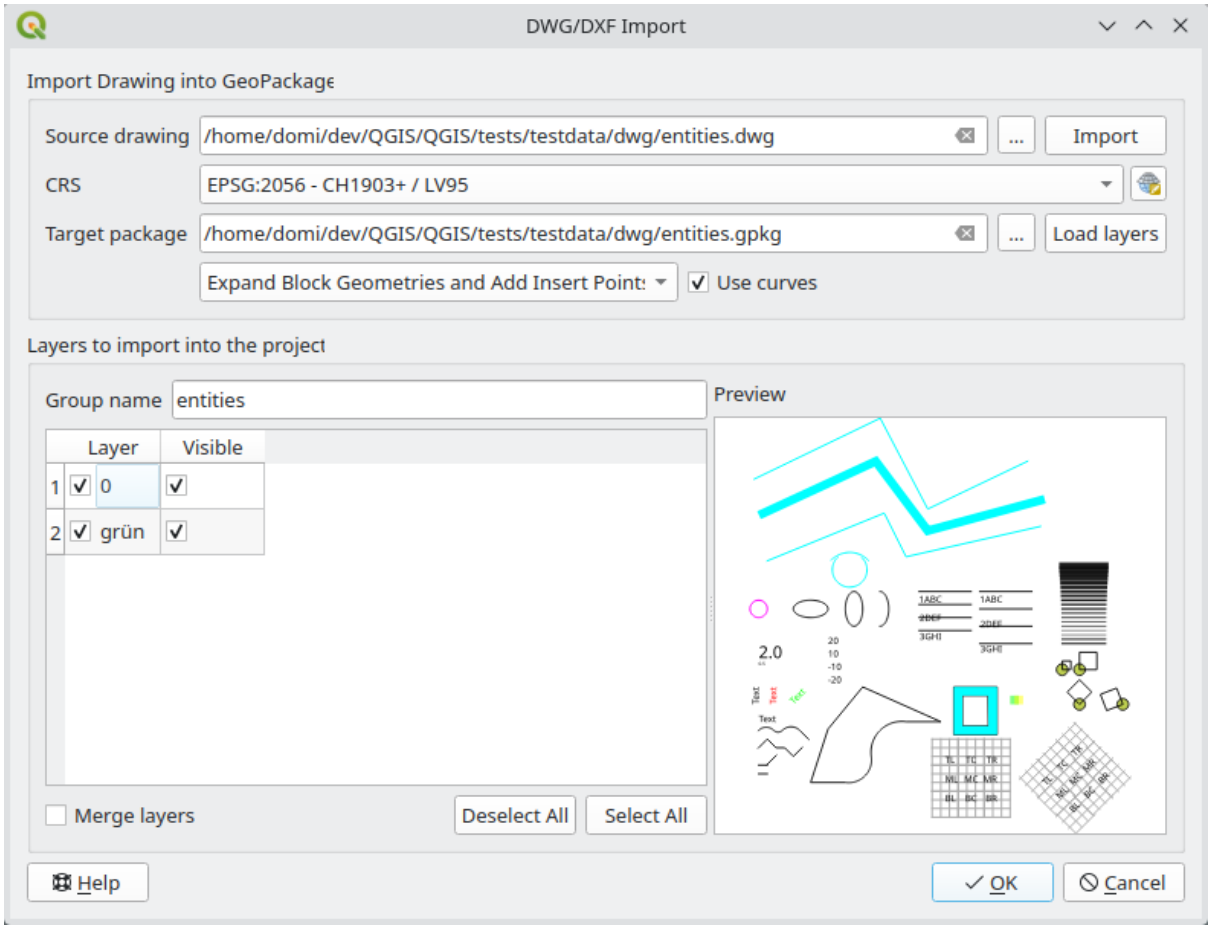


Abb. 11.10: Importdialog für DWG/DXF-Dateien

OpenStreetMap-Vektordaten importieren

Das OpenStreetMap-Projekt ist beliebt, weil in vielen Ländern keine freien Geodaten wie digitale Straßenkarten verfügbar sind. Ziel des OSM-Projekts ist es, aus GPS-Daten, Luftaufnahmen und lokalem Wissen eine frei editierbare Weltkarte zu erstellen. Um dieses Ziel zu unterstützen, bietet QGIS Unterstützung für OSM-Daten.

Mit dem *Browser Bedienfeld* können Sie eine `.osm` Datei in die Kartenansicht laden. In diesem Fall erhalten Sie einen Dialog zur Auswahl von Unterebenen basierend auf dem Geometrietyp. Die geladenen Layer enthalten alle Daten des Geometrietyps aus der `.osm`-Datei und behalten die Datenstruktur der `osm`-Datei bei.

SpatiaLite-Layer

Wenn Sie zum ersten Mal Daten aus einer SpatiaLite-Datenbank laden, beginnen Sie mit:

- Klicken auf die Schaltfläche SpatiaLite Layer hinzufügen in der Werkzeugleiste
- Auswahl der Option *SpatiaLite-Layer hinzufügen…* aus dem Menü *Layer* ► *Layer hinzufügen*
- oder durch Eingabe von `Strg+Umschalt+L`

Es öffnet sich ein Fenster, in dem Sie entweder eine Verbindung zu einer SpatiaLite-Datenbank herstellen können, die QGIS bereits bekannt ist (die Sie aus dem Dropdown-Menü auswählen), oder eine neue Verbindung zu einer neuen Datenbank definieren können. Um eine neue Verbindung zu definieren, klicken Sie auf *Neu* und verwenden Sie den Dateibrowser, um auf Ihre SpatiaLite-Datenbank zu zeigen, die eine Datei mit der Erweiterung `.sqlite` ist.

QGIS unterstützt auch editierbare Ansichten in SpatiaLite.

GPS

Es gibt viele verschiedene Datenformate zum Speichern von GPS-Daten. Das von QGIS unterstützte Format ist GPX (GPS eXchange format), ein Standardformat, welches Wegpunkte, Strecken und Tracks in einer Datei enthalten kann.

Verwenden Sie die Schaltfläche … Browse, um die GPX-Datei auszuwählen, und verwenden Sie dann die Kontrollkästchen, um die Objekte auszuwählen, die Sie aus dieser GPX-Datei laden möchten. Jedes Objekt wird in einen eigenen Layer geladen.

Mehr über GPS- Datenbearbeitung unter *Arbeiten mit GPS Daten*.

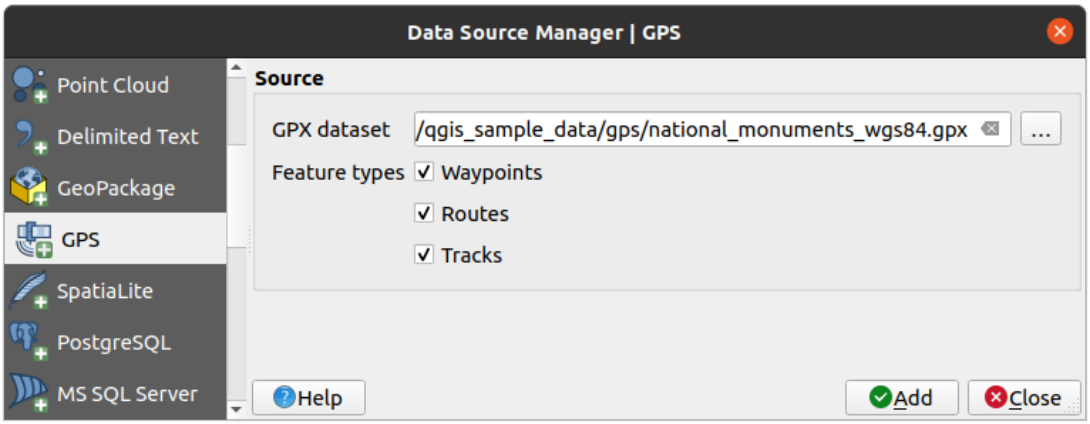


Abb. 11.11: Dialog GPS-Daten laden

GRASS

Die Arbeit mit GRASS-Vektordaten wird im Abschnitt *GRASS GIS Integration* beschrieben.

Datenbankbezogene Werkzeuge

Erstellen einer gespeicherten Verbindung

Um Tabellen aus einem von QGIS unterstützten Datenbankformat lesen und schreiben zu können, müssen Sie eine Verbindung zu dieser Datenbank herstellen. Während der *QGIS Browser* der einfachste und empfohlene Weg ist, sich mit Datenbanken zu verbinden und diese zu nutzen, bietet QGIS andere Werkzeuge, um sich mit jeder dieser Datenbanken zu verbinden und ihre Tabellen zu laden:

- *Add PostgreSQL Layer…* or by typing `Ctrl+Shift+D`
- *Oracle Spatial-Layer hinzufügen…* oder durch Eingabe von `Strg+Umschalt+O`
- *MS-SQL-Server -Layer hinzufügen*
- *SAP-HANA-Spatial-Layer hinzufügen…* oder durch Eingabe von `Strg+Shift+G`

Diese Werkzeuge sind entweder über die *Layerverwaltungswerkzeugleiste ` oder über das Menü :menuselection:`Layer –> Layer hinzufügen –>* zugänglich. Die Verbindung zur SpatiaLite-Datenbank wird unter *SpatiaLite-Layer* beschrieben.

> **Tipp**
>
> **Verbindung zur Datenbank über das Bedienfeld des QGIS-Browsers herstellen**
>
> Wenn Sie das entsprechende Datenbankformat im Browser-Baum auswählen, mit der rechten Maustaste klicken und „Verbinden“ wählen, wird der Dialog für die Datenbankverbindung angezeigt.

Die meisten Verbindungsdialoge folgen einer gemeinsamen Struktur:

- ein Abschnitt mit Anmeldeinformationen für die Verbindung zur Datenbank
- ein Abschnitt mit Optionen zur Einstellung, welche Daten in der Datenbank abgefragt werden können

Verbindung zu PostgreSQL herstellen

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.

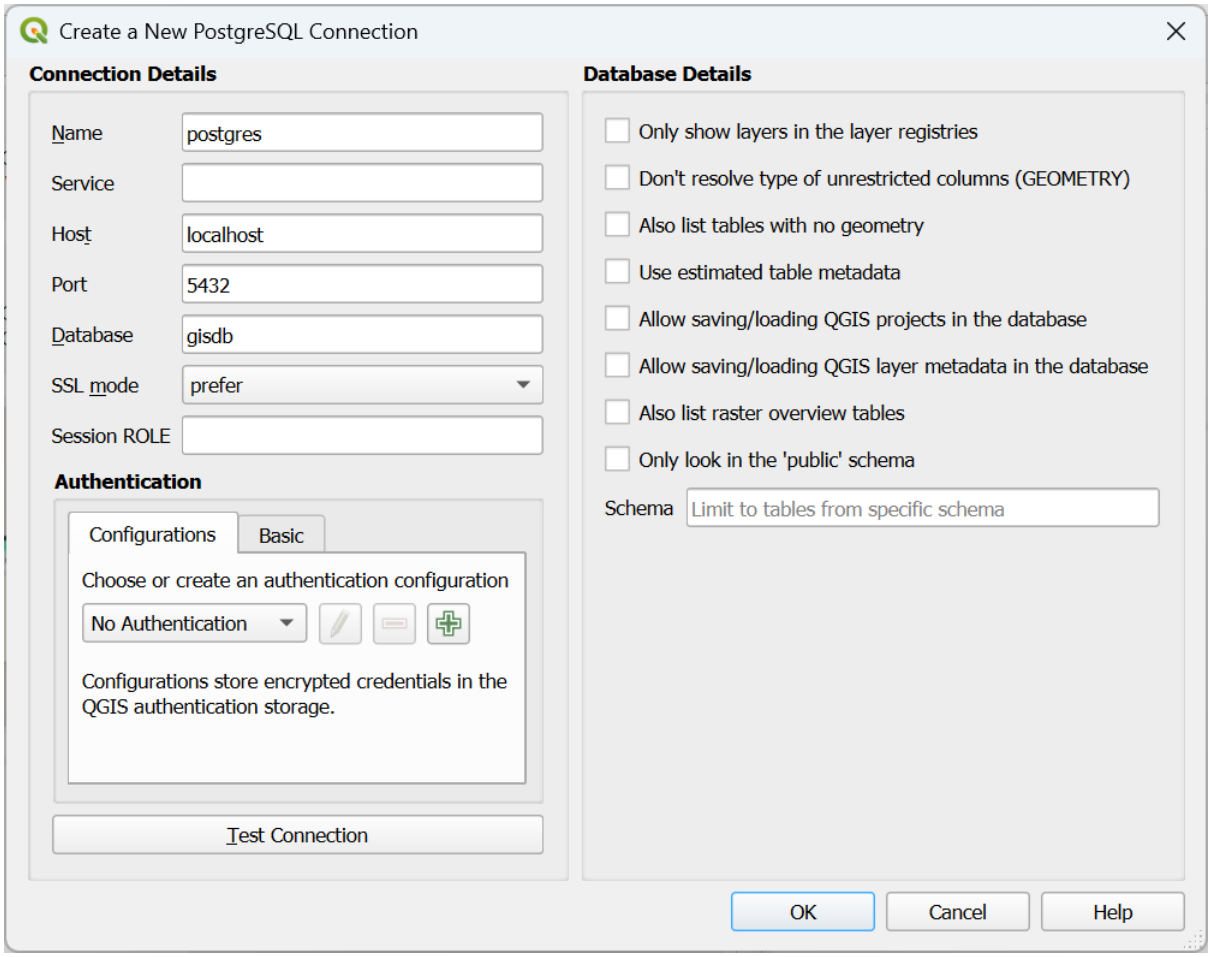


Abb. 11.12: Create a New PostgreSQL Connection Dialog

- *Name*: Ein Name für diese Verbindung. Er kann der gleiche sein wie *Datenbank*.
- *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*: Name des Datenbank-Hosts. Dies muss ein auflösbarer Hostname sein, wie er zum Öffnen einer TCP/IP-Verbindung oder zum Anpingen des Hosts verwendet werden würde. Befindet sich die Datenbank auf demselben Computer wie QGIS, geben Sie hier einfach *localhost* ein.
- *Port*: Port number the PostgreSQL database server listens on. The default port for PostgreSQL is `5432`.
- *Datenbank*: Name der Datenbank.
- *SSL-Modus*: SSL-Verschlüsselungseinstellungen. Folgende Optionen stehen zur Verfügung:
  - *Bevorzugen* (die Standardeinstellung): Die Verschlüsselung ist mir egal, aber ich möchte die Kosten für die Verschlüsselung tragen, wenn der Server sie unterstützt.
  - *Verlangen*: Ich möchte, dass meine Daten verschlüsselt werden, und ich akzeptiere den damit verbundenen Aufwand. Ich vertraue darauf, dass das Netz dafür sorgt, dass ich immer mit dem gewünschten Server verbunden werde.
  - *verify-ca*: Ich möchte, dass meine Daten verschlüsselt werden, und ich akzeptiere die damit verbundenen Kosten. Ich möchte sicher sein, dass ich mich mit einem Server verbinde, dem ich vertraue.
  - *verify-full*: Ich möchte, dass meine Daten verschlüsselt werden, und ich akzeptiere den damit verbundenen Aufwand. Ich möchte sicher sein, dass ich mich mit einem Server verbinde, dem ich vertraue, und dass es der ist, den ich angegeben habe.
  - *Erlauben*: Die Sicherheit ist mir egal, aber ich bin bereit, die Kosten für die Verschlüsselung zu tragen, wenn der Server darauf besteht.
  - *Abschalten*: Ich lege keinen Wert auf Sicherheit, und ich möchte nicht für die Kosten der Verschlüsselung aufkommen.
- *Sitzungsrolle*: Wird verwendet, um die aktuelle Benutzerkennung der aktuellen Sitzung festzulegen. Dies ist nützlich, um die Eigentumsrechte an einem neuen Objekt (Tabelle, Ansicht, Funktion) automatisch der Gruppe „session_role“ zuzuweisen und so die Eigentumsrechte und die damit verbundenen Rechte mit allen Mitgliedern der Gruppe „session_role“ zu teilen. Weitere Informationen finden Sie unter session role.
- *Authentication*: For general details about the authentication dialog behavior, see *Authentifizierung*.

Optional können Sie, je nach Datenbanktyp, die folgenden Kontrollkästchen aktivieren:

- ☑ *Nur Layer aus den Layerregistern anzeigen*
- ☑ *Geometrietyp von Spalten ohne Einschränkung nicht fesstellen (GEOMETRY)*
- ☑ *Auch geometrielose Tabellen anzeigen*: gibt an, dass auch Tabellen ohne Geometrie standardmäßig angezeigt werden sollen.
- ☑ *Geschätzte Tabellenmetadaten nutzen*: Bei der Initialisierung von Layern können verschiedene Abfragen erforderlich sein, um die Eigenschaften der in der Datenbanktabelle gespeicherten Geometrien zu ermitteln. Wenn diese Option aktiviert ist, untersuchen diese Abfragen nur eine Stichprobe der Zeilen und verwenden die Tabellenstatistiken anstelle der gesamten Tabelle. Dies kann die Verarbeitung großer Datensätze erheblich beschleunigen, jedoch zu einer falschen Charakterisierung der Layer führen (z. B. wird die Anzahl der Objekte in gefilterten Layern nicht genau bestimmt) und sogar zu unerwarteten Verhaltensweisen führen, wenn Spalten, die eigentlich eindeutig sein sollten, dies nicht sind.
- ☑ *Speichern/Laden von QGIS-Projekten in/aus der Datenbank erlauben* - mehr Details *hier*
- ☑ *Speichern/Laden von QGIS-Layermetadaten in der Datenbank erlauben* - mehr Details *hier*
- ☑ *Also list raster overview tables*
- ☑ *Nur im Schema ‚public‘ nachschauen*
- *Schema*: Ermöglicht die Angabe eines einzelnen Datenbankschemas, auf das eine Verbindung beschränkt werden soll. Wenn diese Option aktiviert ist, werden nur Tabellen aus dem entsprechenden Schema im Browserfenster und in der Datenquellenauswahl für die Verbindung angezeigt. Dies kann verwendet werden, um

den Datenbankaufwand für das Befüllen von Tabellen für eine Verbindung zu einem großen Datenbankspeicher zu begrenzen

Sobald alle Parameter und Optionen eingestellt sind, können Sie die Verbindung testen, indem Sie auf die Schaltfläche *Verbindung testen* klicken, oder sie übernehmen, indem Sie auf die Schaltfläche *OK* klicken.

PostgreSQL Service connection file

Die Dienstverbindungsdatei ermöglicht es, PostgreSQL-Verbindungsparameter mit einem einzigen Dienstnamen zu verknüpfen. Dieser Servicename kann dann von einem Client angegeben werden und die zugehörigen Einstellungen werden verwendet.

Sie heißt `.pg_service.conf` unter *nix Systemen (GNU/Linux, macOS etc.) und `pg_service.conf` unter Windows.

Die Servicedatei kann wie folgt aussehen:

```
[water_service]
host=192.168.0.45
port=5433
dbname=gisdb
user=paul
password=paulspass

[wastewater_service]
host=dbserver.com
dbname=water
user=waterpass
```

Bemerkung

In dem obigen Beispiel gibt es zwei Dienste: `water_service` und `wastewater_service`. Sie können diese benutzen, um sich von QGIS, pgAdmin, etc. aus zu verbinden, indem Sie nur den Namen des Dienstes angeben, mit dem Sie sich verbinden wollen (ohne die umschließenden Klammern). Wenn Sie den Dienst mit `psql` benutzen wollen, können Sie `psql service=water_service` angeben.

Alle PostgreSQL-Parameter finden Sie hier <https://www.postgresql.org/docs/current/libpq-connect.html#LIBPQ-PARAMKEYWORDS>`_

Bemerkung

Wenn Sie die Passwörter nicht in der Servicedatei speichern wollen, können Sie die Option .pg_pass verwenden.

Bemerkung

QGIS Server und Dienste

Wenn Sie eine Servicedatei und QGIS Server verwenden, müssen Sie den Dienst auch auf der Serverseite konfigurieren. Sie können der QGIS Server Dokumentation folgen.

Auf *nix-Betriebssystemen (GNU/Linux, macOS etc.) können Sie die Datei `.pg_service.conf` im Home-Verzeichnis des Benutzers speichern und PostgreSQL-Clients werden automatisch davon Kenntnis nehmen. Wenn zum Beispiel der angemeldete Benutzer `web` ist, sollte `.pg_service.conf` im Verzeichnis `/home/web/` gespeichert werden, um direkt zu arbeiten (ohne weitere Umgebungsvariablen anzugeben).

Sie können den Speicherort der Servicedatei angeben, indem Sie eine Umgebungsvariable `PGSERVICEFILE` erstellen (führen Sie beispielsweise den Befehl `export PGSERVICEFILE=/home/web/.pg_service.conf` unter Ihrem *nix Betriebssystem aus, um die Variable „PGSERVICEFILE“ vorübergehend festzulegen).

Sie können die Servicedatei auch systemweit (für alle Benutzer) verfügbar machen, indem Sie entweder die Datei `.pg_service.conf` in `pg_config --sysconfdir` platzieren oder die Umgebungsvariable `PGSYSCONFDIR` hinzufügen, um das Verzeichnis mit der Servicedatei anzugeben. Wenn Service-Definitionen mit dem gleichen Namen in der Benutzer- und der Systemdatei existieren, hat die Benutzerdatei Vorrang.

Warnung

Unter Windows gibt es einige Vorbehalte:

- Die Service-Datei sollte als `pg_service.conf` und nicht als `.pg_service.conf` gespeichert werden.
- Die Servicedatei sollte im Unix-Format gespeichert werden, damit sie funktioniert. Eine Möglichkeit, dies zu tun, ist, sie mit Notepad++ zu öffnen und *Edit* ► *EOL Conversion* ► *UNIX Format* ► *File save*.
- Es gibt verschiedene Möglichkeiten, Umgebungsvariablen hinzuzufügen; eine getestete, von der bekannt ist, dass sie zuverlässig funktioniert, ist *Systemsteuerung* ► *System und Sicherheit* ► *System* ► *Erweiterte Systemeinstellungen* ► *Umgebungsvariablen* und fügt `PGSERVICEFILE` mit dem Pfad hinzu - z.B. `C:\Users\John\pg_service.conf`
- Nachdem Sie eine Umgebungsvariable hinzugefügt haben, müssen Sie möglicherweise den Computer neu starten.

Verbinden mit Oracle Spatial

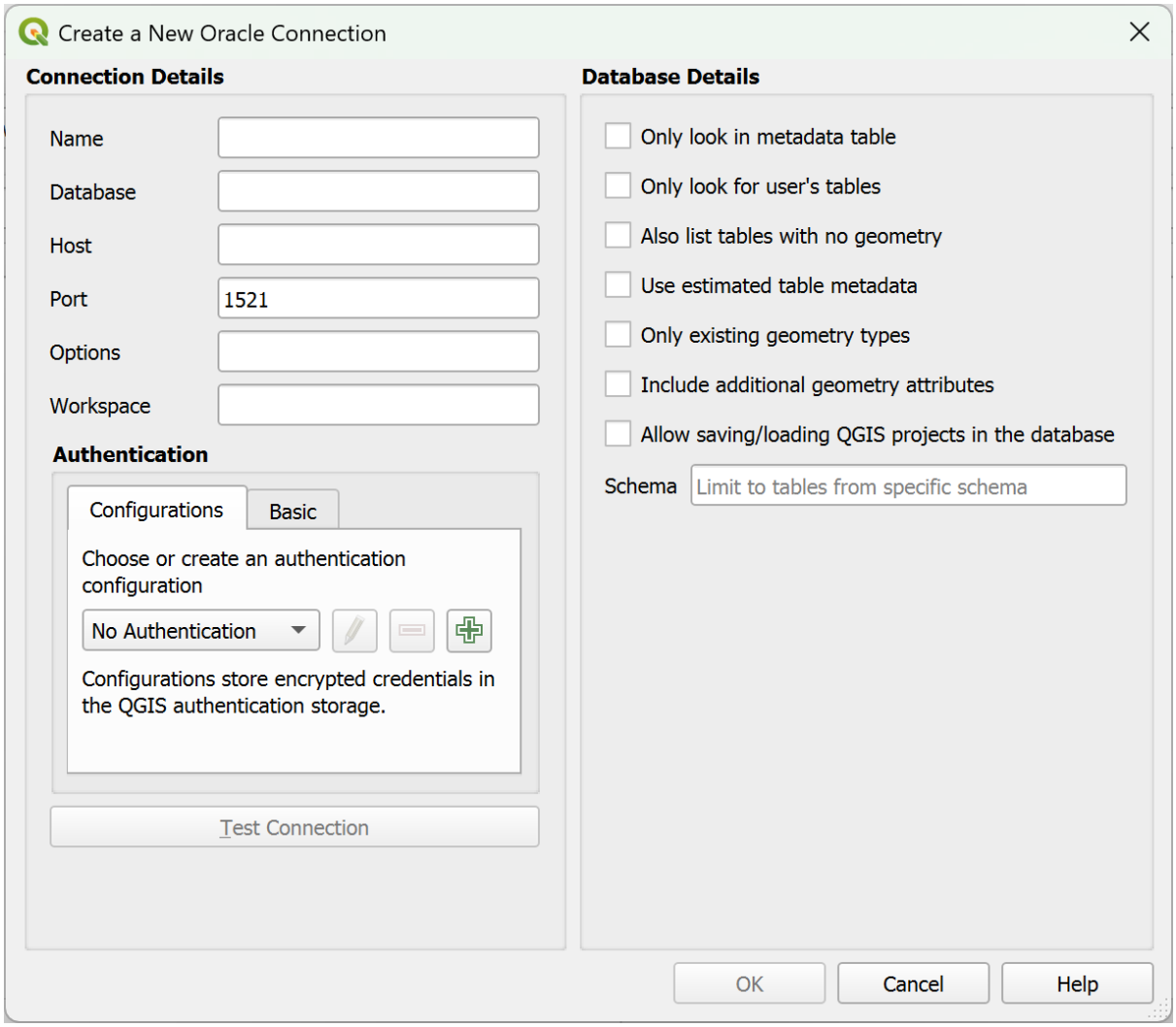


Abb. 11.13: Create a New Oracle Connection Dialog

Die räumlichen Funktionen in Oracle Spatial unterstützen Benutzer bei der Verwaltung von geografischen Daten und Standortdaten in einem nativen Typ innerhalb einer Oracle-Datenbank. Der Verbindungsdialog bietet Folgendes an:

- *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 *Authentifizierung*.

Optional können Sie die folgenden Kontrollkästchen aktivieren:

- *Nur in der Metadatentabelle suchen*: schränkt die angezeigten Tabellen auf diejenigen ein, die sich in der Ansicht `all_sdo_geom_metadata` befinden. Dies kann die anfängliche Anzeige von räumlichen Tabellen

beschleunigen.

- ☑ *Nur Tabellen des Benutzers berücksichtigen*: schränkt die Suche nach räumlichen Tabellen auf Tabellen ein, die dem Benutzer gehören.
- ☑ *Auch geometrielose Tabellen anzeigen*: gibt an, dass auch Tabellen ohne Geometrie standardmäßig angezeigt werden sollen.
- ☑ *Geschätzte Tabellenmetadaten nutzen*: Beim Einrichten des Layers werden verschiedene Metadaten für die Oracle-Tabelle benötigt. Dazu gehören Informationen wie die Anzahl der Tabellenzeilen, der Geometrietyp und die räumliche Ausdehnung der Daten in der Geometriespalte. Enthält die Tabelle eine große Anzahl von Zeilen, kann die Ermittlung dieser Metadaten zeitaufwändig sein. Wenn diese Option aktiviert ist, werden die folgenden schnellen Metadatenoperationen für die Tabelle durchgeführt: Die Zeilenzahl wird aus `all_tables.num_rows` ermittelt. Die Tabellenausdehnung wird immer mit der Funktion SDO_TUNE.EXTENTS_OF bestimmt, auch wenn ein Layer-Filter angewendet wird. Die Tabellengeometrie wird aus den ersten 100 Geometriezeilen in der Tabelle ermittelt, die nicht Null sind.
- ☑ *Nur vorhandene Geometrietypen*: listet nur die vorhandenen Geometrietypen auf und bietet nicht an, weitere hinzuzufügen.
- ☑ *Weitere Geometrieattribute einschließen*.
- ☑ *Speichern/Laden von QGIS-Projekten in/aus der Datenbank erlauben* - mehr Details *hier*
- *Schema*: Ermöglicht die Angabe eines einzelnen Datenbankschemas, auf das eine Verbindung beschränkt werden soll. Wenn diese Option aktiviert ist, werden nur Tabellen aus dem entsprechenden Schema im Browserfenster und in der Datenquellenauswahl für die Verbindung angezeigt. Dies kann verwendet werden, um den Datenbankaufwand für das Befüllen von Tabellen für eine Verbindung zu einem großen Datenbankspeicher zu begrenzen

Tipp

Oracle Spatial-Layer

Normalerweise wird ein Oracle Spatial-Layer durch einen Eintrag in der Tabelle **USER_SDO_METADATA** definiert.

Um sicherzustellen, dass die Auswahlwerkzeuge korrekt funktionieren, wird empfohlen, dass Ihre Tabellen einen **Primärschlüssel** haben.

Verbindung mit MS SQL Server

Wie in *Erstellen einer gespeicherten Verbindung* erwähnt, erlaubt QGIS das Erstellen einer MS SQL Server Verbindung durch die *Datenquellenverwaltung*.

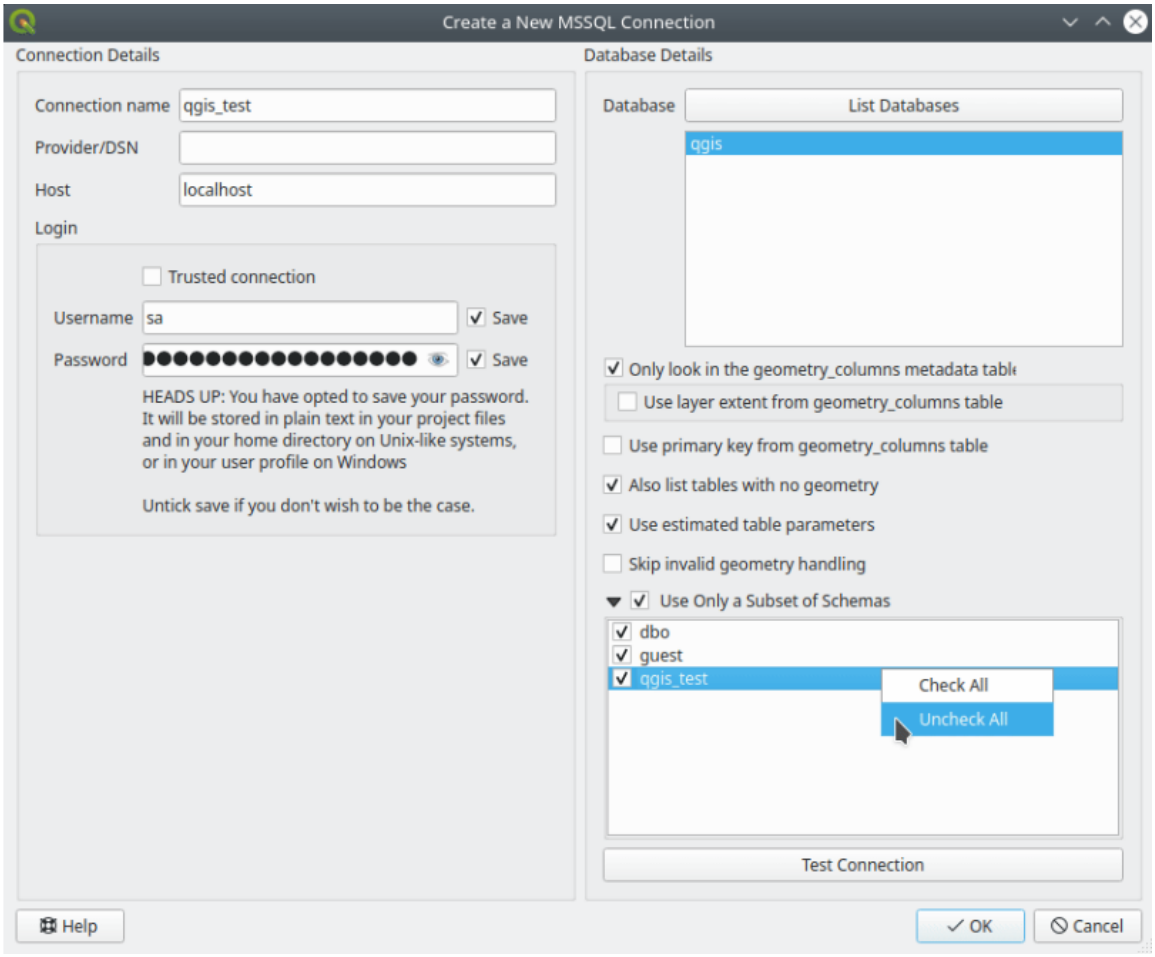


Abb. 11.14: MS SQL-Server-Verbindung

Um eine neue MS SQL Server-Verbindung zu erstellen, müssen Sie einige der folgenden Informationen in das Dialogfeld *Verbindungsdetails* eingeben:

- *Verbindungsname*
- *Datenanbieter/DSN*
- *Host*
- Information zum *Anmelden*. Sie können wählen, ob Sie die Verbindungsdaten *Speichern* wollen.

Navigieren Sie zum Abschnitt *Datenbankdetails* und klicken Sie auf die Schaltfläche *Datenbankliste*, um die verfügbaren Datensätze anzuzeigen. Wählen Sie die gewünschten Datensätze aus und drücken Sie dann auf *OK*. Optional können Sie auch *Verbindung testen* durchführen. Sobald Sie auf *OK* klicken, schließt sich der Dialog *Neue MS SQL Server-Verbindung erstellen* In der *Datenquellenverwaltung* drücken Sie dann auf *Verbinden*, wählen einen Layer aus und klicken auf *Hinzufügen*.

Wahlweise können Sie die folgenden Optionen aktivieren:

- *Nur in der Metadatentabelle geometry_columns nachsehen*: Beschränkt die verfügbaren Tabellen beim Suchen nach Tabellen auf diejenigen in der Metadatentabelle `geometry_columns`. Dies kann das Durchsuchen der Tabellen beschleunigen.
- *Layerausmaße aus Tabelle geometry_columns verwenden*: Diese Option, die von der vorherigen abhängt, ermöglicht es QGIS, die Berechnung der Ausdehnung beim Laden von Layern zu überspringen und somit die zum Laden benötigte Zeit zu verkürzen. Sie basiert auf einer manuell festgelegten Ausdehnung, die mithilfe zusätzlicher QGIS-spezifischer Spalten (`qgis_xmin`, `qgis_xmax`, `qgis_ymin`, `qgis_ymax`) in der Tabelle `geometry_columns` angegeben wird.

- *Primärschlüssel aus Tabelle geometry_columns verwenden*: Ermöglicht es QGIS, die Berechnung des Primärschlüssels für Ansichten beim Laden zu überspringen, wodurch sich die zum Laden benötigte Zeit verkürzt. Dies basiert auf Namen, die manuell in eine QGIS-spezifische Spalte `qgis_pkey` in der Tabelle `geometry_columns` eingegeben werden. Wenn mehr als eine Spalte für den Primärschlüssel verwendet wird, sollten diese als durch Kommas getrennte Werte eingegeben werden.
- *Auch geometrielose Tabellen anzeigen*: Tabellen ohne eine angehängte Geometriespalte werden ebenfalls in der Liste der verfügbaren Tabellen angezeigt.
- *Geschätzte Tabellenparameter nutzen*: Es werden nur geschätzte Tabellenmetadaten verwendet. Dies vermeidet einen langsamen Tabellenscan, kann aber zu falschen Layer-Eigenschaften wie der Layer-Ausdehnung führen.
- *Behandlung ungültiger Geometrien überspringen*: Die Behandlung von Datensätzen mit ungültiger Geometrie wird deaktiviert. Dies beschleunigt den Provider, aber wenn ungültige Geometrien in einer Tabelle vorhanden sind, ist das Ergebnis unvorhersehbar und kann fehlende Datensätze enthalten. Aktivieren Sie diese Option nur, wenn Sie sicher sind, dass alle in der Datenbank vorhandenen Geometrien gültig sind, und dass alle neu hinzugefügten Geometrien oder Tabellen ebenfalls gültig sind.
- *Nur eine Teilmenge von Schemas verwenden* ermöglicht es Ihnen, Schemas für MS SQL-Verbindungen zu filtern. Wenn diese Option aktiviert ist, werden nur markierte Schemata angezeigt. Sie können mit der rechten Maustaste auf ein beliebiges Schema in der Liste klicken, um es zu markieren oder die Markierung aufzuheben.

Verbindung zu SAP HANA

Bemerkung

Sie benötigen den SAP HANA Client, um eine Verbindung zu einer SAP HANA-Datenbank herzustellen. Sie können den SAP HANA Client für Ihre Plattform von der SAP Development Tools Website herunterladen.

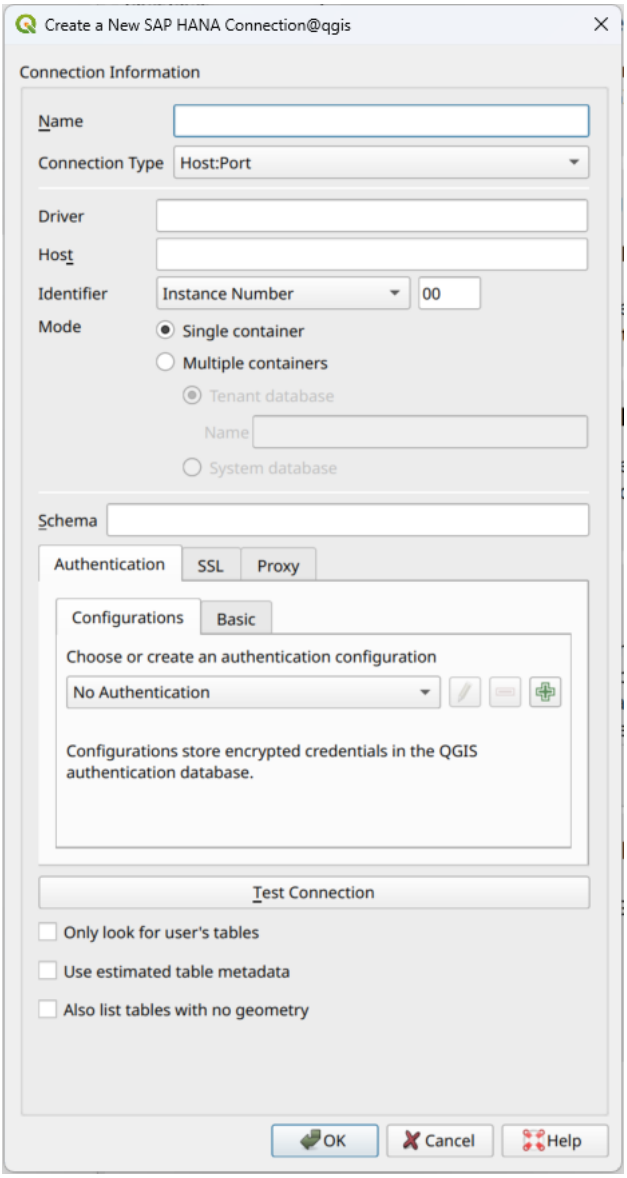


Abb. 11.15: Dialog zum Erstellen einer neuen SAP HANA-Verbindung

Die folgenden Parameter können eingegeben werden:

- *Name*: Ein Name für diese Verbindung.
- *Treiber* : Der Name des HANA ODBC-Treibers. Er lautet `HDBODBC`, wenn Sie 64-bit QGIS verwenden, `HDBODBC32`, wenn Sie 32-bit QGIS verwenden. Der entsprechende Treibername wird automatisch eingetragen.
- *Treiber* : Entweder der Name, unter dem der SAP HANA ODBC-Treiber in `/etc/odbcinst.ini` registriert wurde oder der vollständige Pfad zum SAP HANA ODBC-Treiber. Das SAP HANA Client-Installationsprogramm installiert den ODBC-Treiber standardmäßig in `/usr/sap/hdbclient/libodbcHDB.so`.
- *Host*: Der Name des Datenbank-Hosts.
- *Bezeichner*: Identifiziert die Instanz, zu der eine Verbindung auf dem Host hergestellt werden soll. Dies kann entweder *Instanznummer* oder *Portnummer* sein. Instanznummern bestehen aus zwei Ziffern, Portnummern liegen im Bereich von 1 bis 65.535.
- *Mode*: Legt den Modus fest, in dem die SAP HANA-Instanz läuft. Diese Einstellung wird nur berücksichtigt, wenn *Kennung* auf *Instanznummer* gesetzt ist. Wenn die Datenbank mehrere Container beherbergt, können

Sie sich entweder mit einem Tenant mit dem unter *Tenant Datenbank* angegebenen Namen oder mit der Systemdatenbank verbinden.

- *Schema*: Dieser Parameter ist optional. Wenn ein Schemaname angegeben wird, wird QGIS nur nach Daten in diesem Schema suchen. Wird dieses Feld leer gelassen, sucht QGIS nach Daten in allen Schemata.
- *Authentication*: For general details about the authentication dialog behavior, see *Authentifizierung*.
- *SSL Reiter*
  - *TLS/SSL-Verschlüsselung einschalten*: Aktiviert die Verschlüsselung TLS 1.1 - TLS1.2. Der Server wählt die höchste verfügbare aus.
  - *Datenanbieter*: Gibt den Anbieter der kryptographischen Bibliothek an, die für die SSL-Kommunikation verwendet wird. *sapcrypto* sollte auf allen Plattformen funktionieren, *openssl* sollte auf funktionieren, *mscrypto* sollte auf funktionieren und *commoncrypto* erfordert die Installation von CommonCryptoLib.
  - *SSL-Zertifikat validieren*: Wenn diese Option aktiviert ist, wird das SSL-Zertifikat mit Hilfe des in *Vertrauensspeicherdatei mit öffentlichem Schlüssel* angegebenen Vertrauensspeichers validiert.
  - *Hostnamen des Zertifikats übersteuern*: Gibt den Hostnamen an, der zur Überprüfung der Identität des Servers verwendet wird. Der hier angegebene Hostname prüft die Identität des Servers anstelle des Hostnamens, mit dem die Verbindung hergestellt wurde. Wenn Sie `*` als Hostname angeben, wird der Hostname des Servers nicht überprüft. Andere Wildcards sind nicht erlaubt.
  - *Keystore-Datei mit privatem Schlüssel*: Wird derzeit ignoriert. Dieser Parameter könnte in Zukunft die Authentifizierung über ein Zertifikat statt über Benutzer und Passwort ermöglichen.
  - *Vertrauensspeicherdatei mit öffentlichem Schlüssel*: Gibt den Pfad zu einer Truststore-Datei an, die die öffentlichen Zertifikate des Servers enthält, wenn OpenSSL verwendet wird. Normalerweise enthält der Trust Store das Stammzertifikat oder das Zertifikat der Zertifizierungsstelle, die die öffentlichen Zertifikate des Servers signiert hat. Wenn Sie die kryptografische Bibliothek CommonCryptoLib oder msCrypto verwenden, lassen Sie diese Eigenschaft leer.
- *Nur Tabellen des Benutzers berücksichtigen*: Wenn diese Option aktiviert ist, sucht QGIS nur nach Tabellen und Ansichten, die dem Benutzer gehören, der sich mit der Datenbank verbindet.
- *Geschätzte Tabellenparameter nutzen*: Wenn diese Option aktiviert ist, werden geschätzte Tabellenparameter verwendet, sofern verfügbar. Bei großen Tabellen vermeidet dies langsame Tabellenladevorgänge und potenziell aufwendige Berechnungen, kann jedoch zu falschen Layereigenschaften wie beispielsweise der Layerausdehnung führen. Die schnelle Ausdehnungsschätzung ist ab QRC1/2024 und SP8 in HANA Cloud bzw. HANA On-Premise verfügbar.
- *Auch geometrielose Tabellen anzeigen*: Wenn diese Option aktiviert ist, sucht QGIS auch nach Tabellen und Ansichten, die keine räumliche Spalte enthalten.

Tipp

Verbindung zur SAP HANA Cloud

Wenn Sie sich mit einer SAP HANA Cloud-Instanz verbinden möchten, müssen Sie normalerweise :guilabel:` Port Nummer` auf `443` setzen und *TLS/SSL-Verschlüsselung aktivieren*` aktivieren.

Laden eines Datenbank-Layers

Once you have one or more connections defined to a database (see section *Erstellen einer gespeicherten Verbindung*), you can load layers from it. Of course, this requires that data are available. See section *Daten in PostgreSQL importieren* for a discussion on importing data into a PostgreSQL database.

Um einen Layer aus einer Datenbank zu laden, können Sie die folgenden Schritte durchführen:

1. Öffnen Sie die entsprechende Registerkarte der Datenbank im Dialogfeld *Datenquellenverwaltung*.
2. Wählen Sie den Verbindungsnamen aus der Dropdown-Liste aus und klicken Sie auf *Verbinden*.
3. The table below will be filled with your data grouped by schema, with a number of metadata information helpful for loading.
4. An- oder abwählen *Auch geometrielose Tabellen anzeigen*.
5. Optional können Sie mit *Suchoptionen* die Liste der Tabellen auf diejenigen reduzieren, die Ihrer Suche entsprechen. Sie können diese Option auch setzen, bevor Sie auf die Schaltfläche *Verbinden* klicken, um den Abruf der Datenbank zu beschleunigen.
6. Suchen Sie den/die Layer, den/die Sie hinzufügen möchten, in der Liste der verfügbaren Layer.
7. Wählen Sie sie aus, indem Sie darauf klicken. Sie können mehrere Layer auswählen, indem Sie beim Klicken die Taste `Shift` oder `Strg` gedrückt halten.

   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.

   > **Tipp**
   >
   > **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. Falls zutreffend, verwenden Sie die Schaltfläche *Filter setzen* (oder doppelklicken Sie auf den Layer), um den Dialog *Abfragegenerator* (siehe Abschnitt *Abfrageerstellung*) zu starten und festzulegen, welche Objekte vom ausgewählten Layer geladen werden sollen. Der Filterausdruck erscheint in der Spalte `sql`. Diese Einschränkung kann im Rahmen *Layereigenschaften* ► *Allgemein* ► *Provider Feature Filter* entfernt oder bearbeitet werden.
9. Das standardmäßig aktivierte Kontrollkästchen in der Spalte `Auswählen nach ID` liefert die Objekt-IDs ohne die Attribute und beschleunigt im Allgemeinen das Laden der Daten.
10. Klicken Sie auf die Schaltfläche *Hinzufügen*

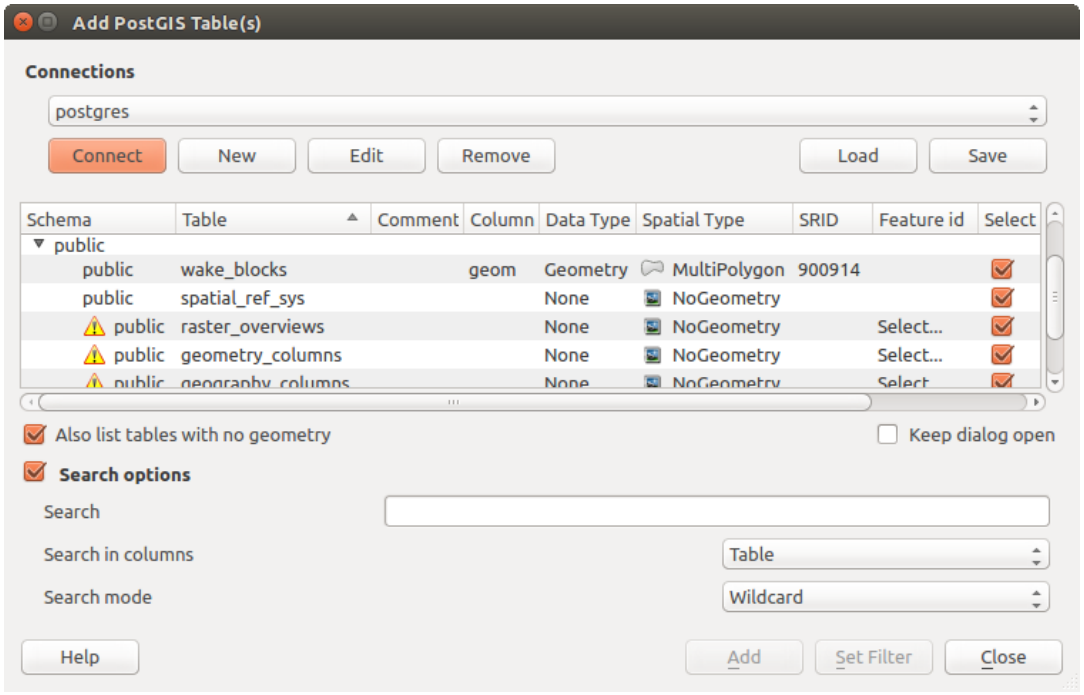


Abb. 11.16: Add PostgreSQL Table(s) Dialog

Tipp

Verwenden Sie das „Browser“ Bedienfeld, um das Laden von Datenbanktabellen zu beschleunigen

Das Hinzufügen von DB-Tabellen aus der *Datenquellenverwaltung* kann manchmal zeitaufwendig sein, da QGIS vorher Statistiken und Eigenschaften (z.B. Geometrietyp und -feld, KBS, Anzahl der Objekte) für jede Tabelle abruft. Um dies zu vermeiden, ist es besser, sobald :ref:`die Verbindung gesetzt ist <vector_create_stored_connection> `, das :ref:`Browser Panel <browser_panel> ` oder den :ref:`DB Manager <dbmanager> ` zu verwenden, um die Datenbanktabellen in die Kartenansicht zu ziehen und abzulegen.

11.1.4 Das Bedienfeld für die Suche nach Layer-Metadaten

Standardmäßig kann QGIS die Layer-Metadaten von den Verbindungen oder Datenanbietern abrufen, die die Speicherung von Metadaten erlauben (mehr Details zu *Metadaten in der Datenbank speichern*). Das Bedienfeld *Metadatensuche* erlaubt es, die Layer nach ihren Metadaten zu durchsuchen und sie dem Projekt hinzuzufügen (entweder mit einem Doppelklick oder mit der Schaltfläche *Hinzufügen*). Die Liste kann gefiltert werden:

- nach Text, unter Beobachtung einer Reihe von Metadateneigenschaften (Kennung, Titel, Zusammenfassung)
- nach räumlicher Ausdehnung, unter Verwendung der aktuellen :ref:`Projektausdehnung <project_full_extent> ` oder der Kartenansicht
- durch den Layer (Geometrie) Typ

Bemerkung

Die Quellen der Metadaten werden durch ein Schicht-Metadaten-Provider-System implementiert, das durch Plugins erweitert werden kann.

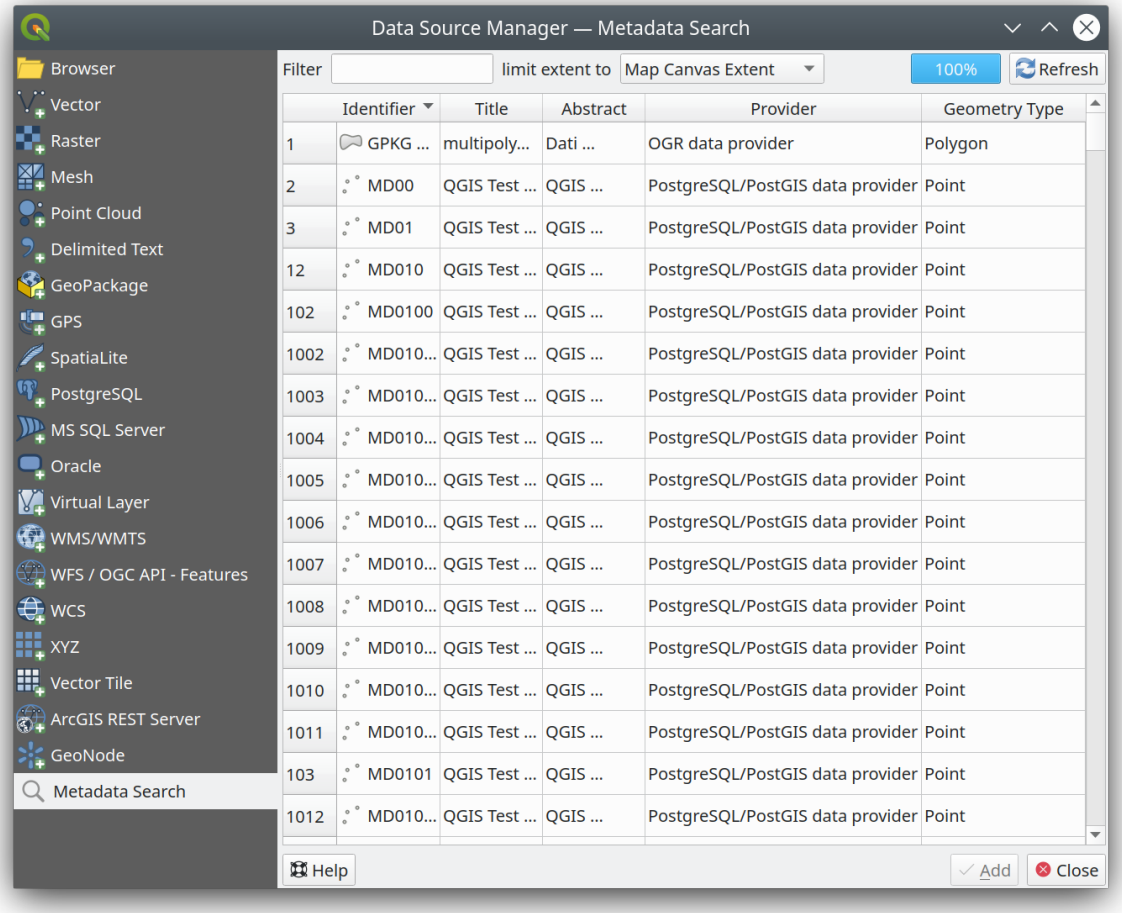


Abb. 11.17: Suchfeld für Layer-Metadaten

11.1.5 Eigene QGIS Formate

QGIS bietet zwei eigene Formate an:

- Temporärer Scratch- Layer: ein Speicher- Layer, die an das Projekt gebunden ist (siehe *Erstellen eines neuen temporären Scratch Layers* für weitere Informationen)
- Virtuelle Layer: ein Layer, der aus einer Abfrage auf anderen Layern resultiert (siehe *Erstellen von virtuellen Layern* für weitere Informationen)

11.1.6 QLR - QGIS Layer Definitionsdatei

Layer-Definitionen können als *Layer Definitionsdatei* (QLR - `.qlr`) mit :menuselection:` Export –> Als Layerdefinitionsdatei speichern…` im Layer-Kontextmenü gespeichert werden.

Das QLR-Format ermöglicht es, „vollständige“ QGIS- Layer mit anderen QGIS-Nutzern zu teilen. QLR-Dateien enthalten Links zu den Datenquellen und alle QGIS-Stilinformationen, die für die Gestaltung des Layers erforderlich sind.

QLR-Dateien werden im Browser-Bedienfeld angezeigt und können verwendet werden, um Layer (mit ihren gespeicherten Stilen) zum Layer-Bedienfeld hinzuzufügen. Sie können QLR-Dateien auch per Drag & Drop aus dem Systemdateimanager auf die Kartenansicht ziehen.

Die folgenden Aktionen sind für QLR-Dateien im Browser-Panel verfügbar:

- *Exportiere Layer ► In Datei*
- *Layer zu Projekt hinzufügen*
- *Layereigenschaften…*

11.1.7 Verbindung zu Webdiensten

Mit QGIS können Sie Zugang zu verschiedenen Arten von OGC-Webdiensten (WM(T)S, WFS(-T), WCS, CSW, …) erhalten. Dank QGIS Server können Sie solche Dienste auch veröffentlichen. QGIS-Server-manual enthält Beschreibungen dieser Möglichkeiten.

Verwendung von Vector Tiles-Diensten (Vektorkacheln)

Vektorkachel-Services können über die Registerkarte *Vektorkacheln* des *Datenquellenverwaltung*-Dialogs oder das Kontextmenü des Eintrags *Vector Tiles* im *Browser*-Bedienfeld hinzugefügt werden. Dienste können entweder eine *Neue allgemeine Verbindung…* oder eine *Neue ArcGIS-Vektorkacheldienst-Verbindung…* sein.

Sie richten einen Dienst ein, indem Sie ihn hinzufügen:

- einen *Namen*
- eine *Stil-URL*: eine URL zu einer MapBox GL JSON Stil-Konfiguration. Falls angegeben, wird dieser Stil immer dann angewendet, wenn die Layer der Verbindung zu QGIS hinzugefügt werden. Im Falle von Arcgis Vektor-Kachel-Service-Verbindungen überschreibt die URL die in der Serverkonfiguration angegebene Standard-Stilkonfiguration.

  Sie können Vektor-Kacheln direkt über eine *Stil-URL* laden. Die Datenquelle wird automatisch aus dem Stil analysiert (URLs mit mehreren Quellen werden unterstützt). Dadurch wird *Quell-URL* optional.
- die *Quell-URL*: vom Typ `http://example.com/{z}/{x}/{y}.pbf` für generische Dienste und `http://example.com/arcgis/rest/services/Layer/VectorTileServer` für ArcGIS-basierte Dienste. Der Dienst muss Kacheln im `.pbf`-Format bereitstellen.
- das Feld ☑ *Minimale Zoomstufe* und die ☑ *Maximale Zoomstufe*. Vektor-Kacheln haben eine Pyramidenstruktur. Mit diesen Optionen haben Sie die Möglichkeit, einzelne Ebenen aus der Kachelpyramide zu erzeugen. Diese Layer werden dann zum Rendern der Vektorkachel in QGIS verwendet.

  Bei der Mercator-Projektion (die von OpenStreetMap Vector Tiles verwendet wird) stellt die Zoomstufe 0 die gesamte Welt im Maßstab 1:500.000.000 dar. Zoomstufe 14 entspricht dem Maßstab 1:35.000.
- the *authentication* configuration if necessary
- einen *Referer*

Abb. 11.18 zeigt das Dialogfeld mit der Konfiguration des Vektorkachel-Dienstes an.

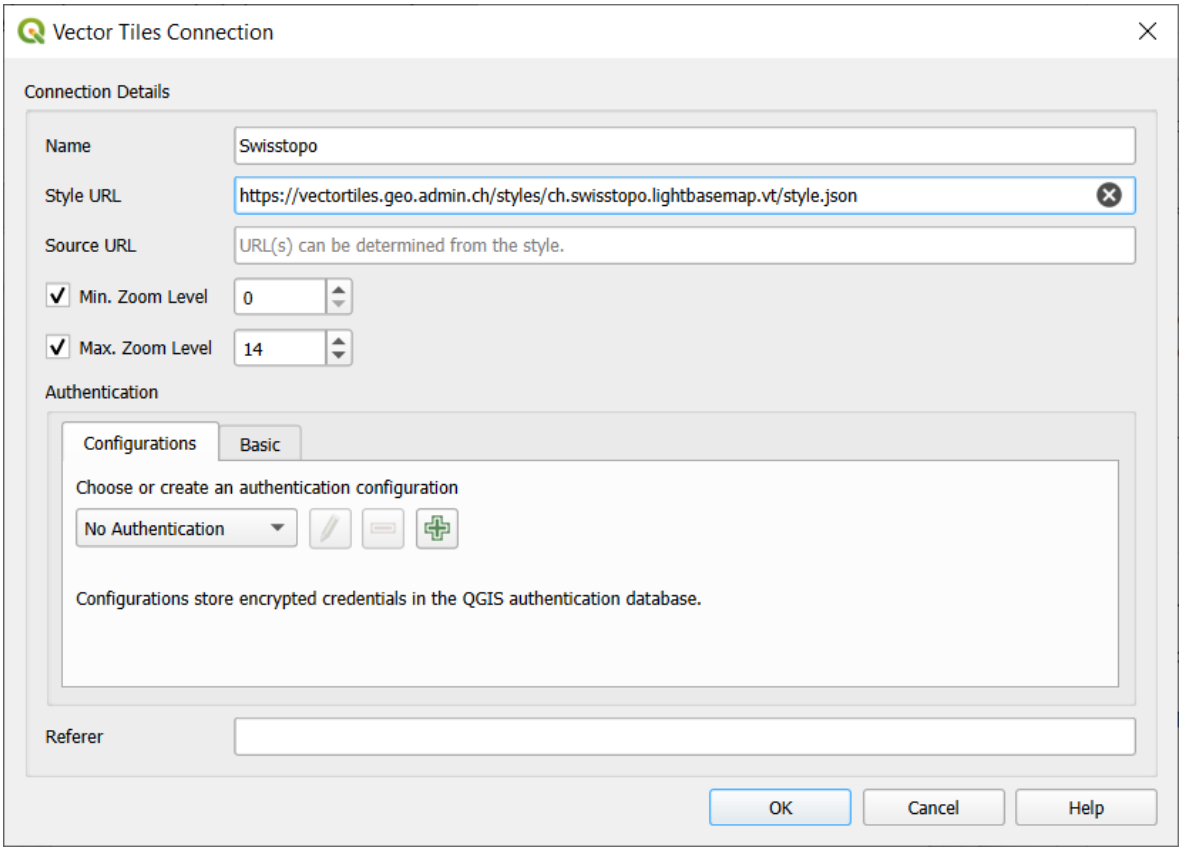


Abb. 11.18: Vektor-Kacheln – Verbindungseinstellungen

Konfigurationen können in einer `.XML`-Datei (*Speichern*) über den Eintrag *Vektorkacheln* im *Datenquellenmanager*-Dialog oder dessen Kontextmenü im *Browser*- Bedienfeld gespeichert werden. Ebenso können sie aus einer Datei (*Laden*) hinzugefügt werden.

Sobald eine Verbindung zu einem Vektorkacheldienst hergestellt ist, ist es möglich:

- *Bearbeiten* der Verbindungseinstellungen der Vektorkacheln
- *Entfernen* der Verbindung
- Klicken Sie im *Browser* Bedienfeld mit der rechten Maustaste auf den Eintrag und Sie können auch:
  - *Layer zu Projekt hinzufügen*; ein Doppelklick fügt auch den Layer hinzu
  - Sehen Sie sich die *Layereigenschaften…* an und erhalten Sie Zugriff auf Metadaten und eine Vorschau auf die vom Dienst bereitgestellten Daten. Weitere Einstellungen sind verfügbar, wenn der Layer in das Projekt geladen wurde.

Verwendung von XYZ Tile-Diensten

XYZ-Kacheldienste können über die Registerkarte *XYZ* des *Datenquellenverwaltung*-Dialogs oder über das Kontextmenü des Eintrags *XYZ Tiles* im *Browser* Bedienfeld hinzugefügt werden. Standardmäßig bietet QGIS einige Standard- und gebrauchsfertige XYZ-Kacheldienste an:

- *Mapzen Global Terrain*, die einen sofortigen Zugriff auf globale DEM-Quellen für die Projekte ermöglicht. Weitere Details und Ressourcen unter https://registry.opendata.aws/terrain-tiles/
- *OpenStreetMap*, um auf die 2D-Weltkarte zuzugreifen. Abb. 11.19 zeigt den Dialog mit der Konfiguration des OpenStreetMap XYZ Kacheldienstes.

Um einen neuen Dienst hinzuzufügen, drücken Sie *Neu* (bzw. *Neue Verbindung* auf dem Bedienfeld des Browsers) und geben Sie ein:

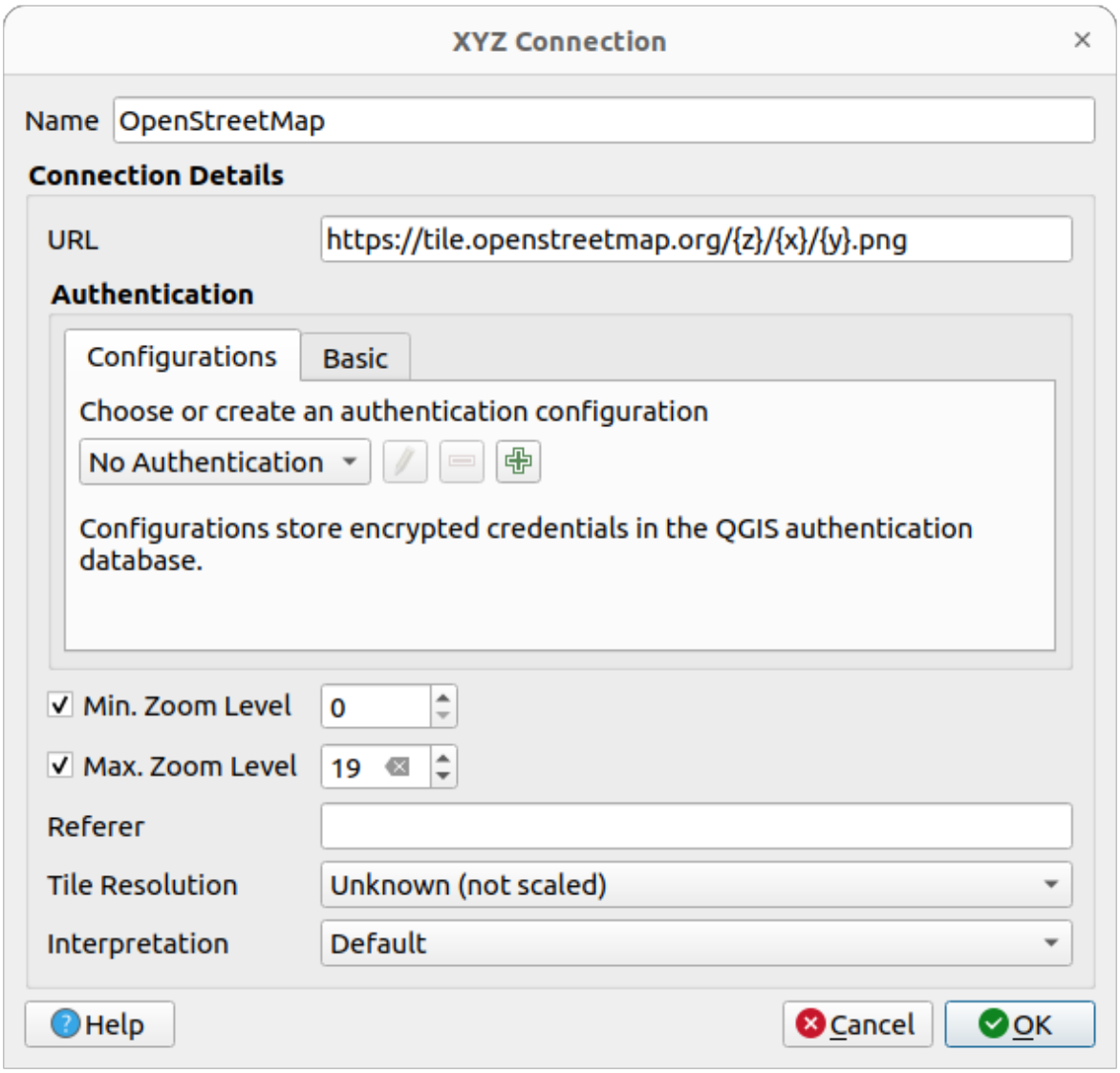


Abb. 11.19: XYZ Tiles - OpenStreetMap Konfiguration

- einen *Namen*
- die *URL*, Sie können `http://example.com/{z}/{x}/{y}.png` oder `file:///local_path/{z}/{x}/{y}.png` hinzufügen
- the *authentication* configuration if necessary
- die *Min. Zoomstufe* und *Max. Zoomstufe*
- einen *Referer*
- die *Kachelauflösung*: mögliche Werte sind *Unbekannt (nicht skaliert)*, *Standard (256x256 / 96DPI)* und *Hoch (512x512 / 192DPI)*
- *Interpretation*: konvertiert WMTS/XYZ-Rasterdatensätze in einen Raster Layer vom Typ Single Band Float nach einem vordefinierten Kodierungsschema. Unterstützte Schemata sind *Default* (es wird keine Konvertierung durchgeführt), *MapTiler Gelände RGB* und *Terrarium Gelände RGB*. Der ausgewählte Konverter wandelt die RGB-Quellwerte in Float-Werte für jedes Pixel um. Sobald der Layer geladen ist, wird er als ein einbandiger Fließkomma-Raster-Layer dargestellt, der mit dem in QGIS üblichen *Raster Renderer* gestaltet werden kann.

Drücken Sie *OK*, um die Verbindung herzustellen. Danach ist es möglich:

- *Hinzufügen* des neuen Layers zum Projekt; er wird mit dem in den Einstellungen angegebenen Namen geladen.
- *Bearbeiten* der XYZ-Verbindungseinstellungen
- *Entfernen* der Verbindung
- Klicken Sie im *Browser* Bedienfeld mit der rechten Maustaste auf den Eintrag und Sie können auch:
  - *Layer exportieren…* ► *In Datei*, *Speichern als Raster*
  - *Layer zu Projekt hinzufügen*; ein Doppelklick fügt auch den Layer hinzu
  - Sehen Sie sich die *Layereigenschaften…* an und erhalten Sie Zugriff auf Metadaten und eine Vorschau auf die vom Dienst bereitgestellten Daten. Weitere Einstellungen sind verfügbar, wenn der Layer in das Projekt geladen wurde.

Konfigurationen können über den Eintrag *XYZ* im Dialog *Datenquellenverwaltung* oder dessen Kontextmenü im *Browser* Bedienfeld in einer `.XML`-Datei (*Speichern*) gespeichert werden. Ebenso können sie aus einer Datei hinzugefügt werden (*Laden*).

Die XML-Datei für OpenStreetMap sieht wie folgt aus:

```
<!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>
```

Tipp

Laden von XYZ-Kacheln, ohne eine Verbindung herzustellen

Es ist auch möglich, XYZ-Kacheln zu einem Projekt hinzuzufügen, ohne die Verbindungseinstellungen in Ihrem Benutzerprofil zu speichern (z. B. für einen Datensatz, den Sie nur einmal benötigen). In der Registerkarte *Datenquellenverwaltung* ► *XYZ* bearbeiten Sie alle Eigenschaften in der Gruppe *Verbindungsdetails*. Das obige Feld *Name* sollte sich in `Benutzerdefiniert` ändern. Drücken Sie *Hinzufügen*, um den Layer in das Projekt zu laden. Sie wird standardmäßig als `XYZ Layer` bezeichnet.

Beispiele für XYZ Tile Dienste:

- OpenStreetMap Monochrome: *URL*: `http://tiles.wmflabs.org/bw-mapnik/{z}/{x}/{y}.png`, *Min. Zoomstufe*: 0, *Max. Zoomstufe*: 19.
- Google Maps: *URL*: `https://mt1.google.com/vt/lyrs=m&x={x}&y={y}&z={z}`, *Min. Zoomstufe*: 0, *Max. Zoomstufe*: 19.
- Open Weather Map Temperature: *URL*: `http://tile.openweathermap.org/map/temp_new/{z}/{x}/{y}.png?appid={api_key}` *Min. Zoomstufe*: 0, *Max. Zoomstufe*: 19.

Verwendung von ArcGIS REST-Servern

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.

> **Bemerkung**
>
> ArcGIS Feature Service-Verbindungen, für die entsprechende Portal-Endpunkt-URLs festgelegt wurden, können im Bedienfeld des Browsers nach Inhaltsgruppen durchsucht werden.
>
> Wenn für eine Verbindung die Portal-Endpunkte festgelegt sind, wird beim Aufklappen der Verbindung im Browser ein Ordner „Gruppen“ und „Dienste“ angezeigt, anstatt der vollständigen Liste der Dienste, die normalerweise angezeigt wird. Wenn Sie den Ordner „Gruppen“ ausklappen, wird eine Liste aller Inhaltsgruppen angezeigt, in denen der Benutzer Mitglied ist, und jede dieser Gruppen kann ausgeklappt werden, um die zu dieser Gruppe gehörenden Dienste anzuzeigen.

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:
   - *Bearbeiten* der ArcGIS REST Server Verbindungseinstellungen
   - *Entfernen* der Verbindung
   - *Neu laden* der Verbindung
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.

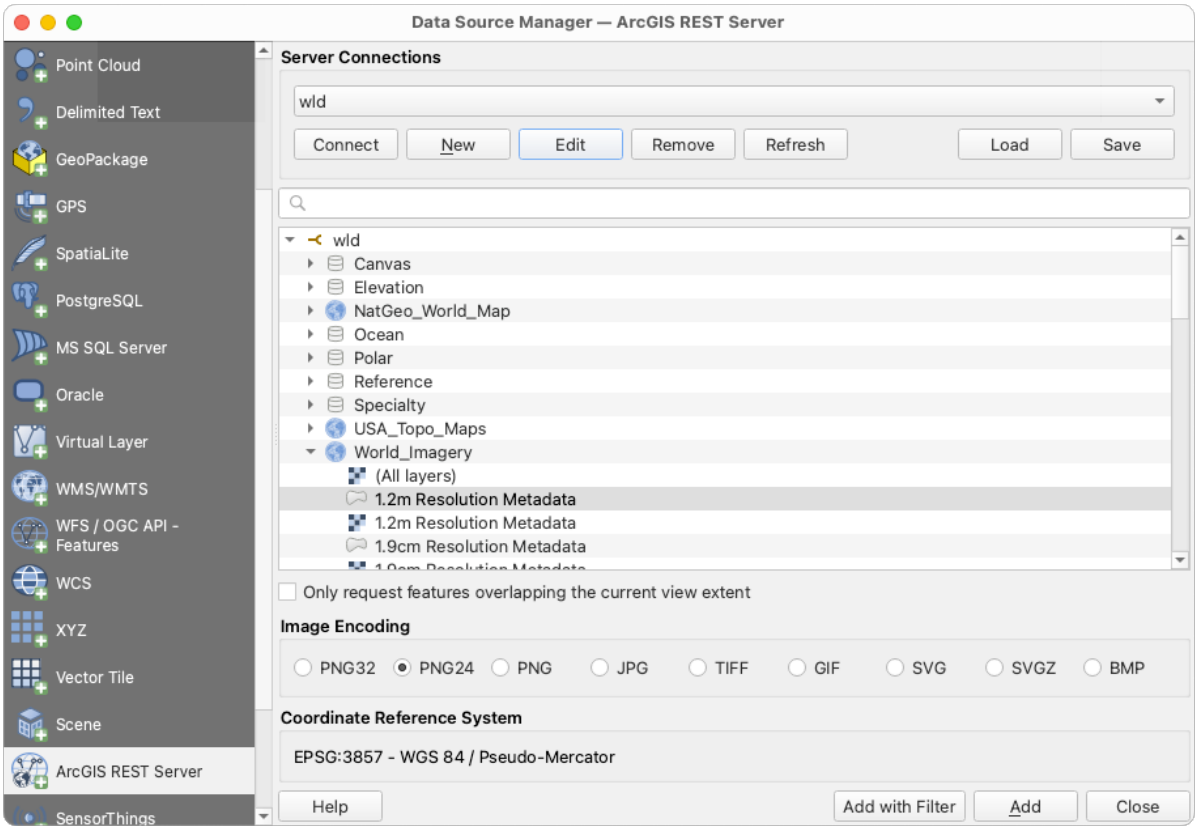


Abb. 11.20: Data Source Manager - ArcGIS REST Server

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 *Der Ausdruckseditor* 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.

Verwendung von 3D-Kachelszenendiensten

QGIS unterstützt mehrere Formate von 3D-Kachel-Datensätzen, die als „Kachelszenen" zusammengefasst sind. Dazu gehören Cesium 3D-Kacheln und quantisierte Mesh-Kacheln.

Um einen gekachelten Szenendatensatz in QGIS zu laden, verwenden Sie die Registerkarte *Szene* im Dialogfeld *Datenquellenverwaltung*.

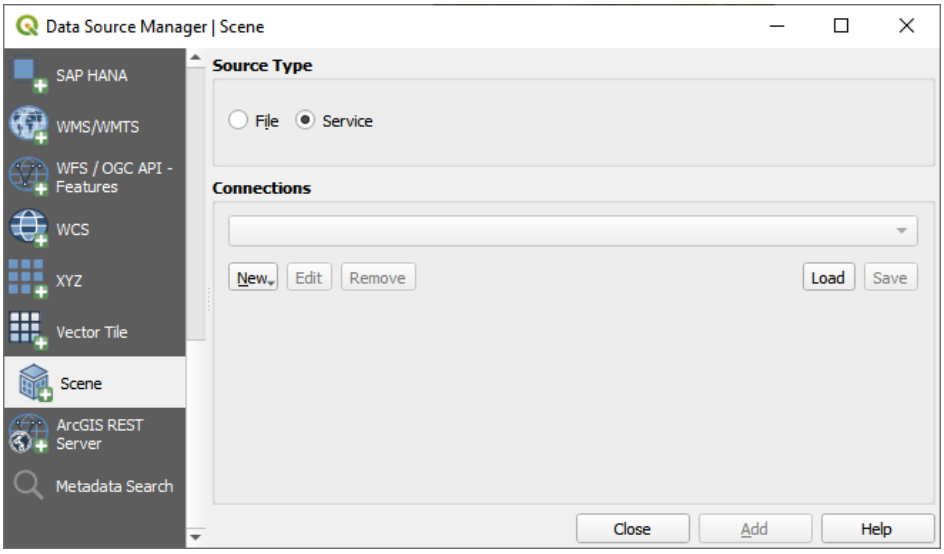


Abb. 11.21: Datenquellenverwaltung - Szene-Reiter

Erstellen Sie eine Verbindung, indem Sie auf *Neu* klicken. Sie können eine *Neue Cesium 3D-Kachelverbindung* oder eine *Neue quantifizierte Netzverbindung* hinzufügen.

Wählen Sie einen *Namen* und legen Sie die *URL* auf die URL einer JSON-Datei mit der Layer-Beschreibung fest.

Die URL kann remote (z. B. `http://example.com/tileset.json`) oder lokal (z. B. `file:///path/to/tiles/tileset.json`) sein.

The *authentication* configuration if necessary.

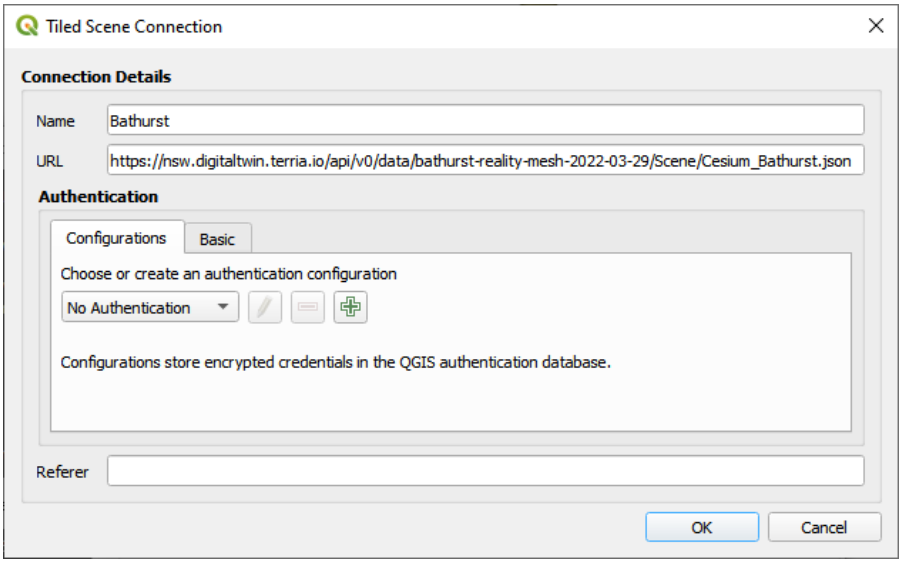


Abb. 11.22: Kachelszenenverbindung

Sie können den Dienst auch über das *Browser* Bedienfeld hinzufügen.

Nachdem Sie eine neue Verbindung erstellt haben, können Sie den neuen Layer zu Ihrer Karte hinzufügen.

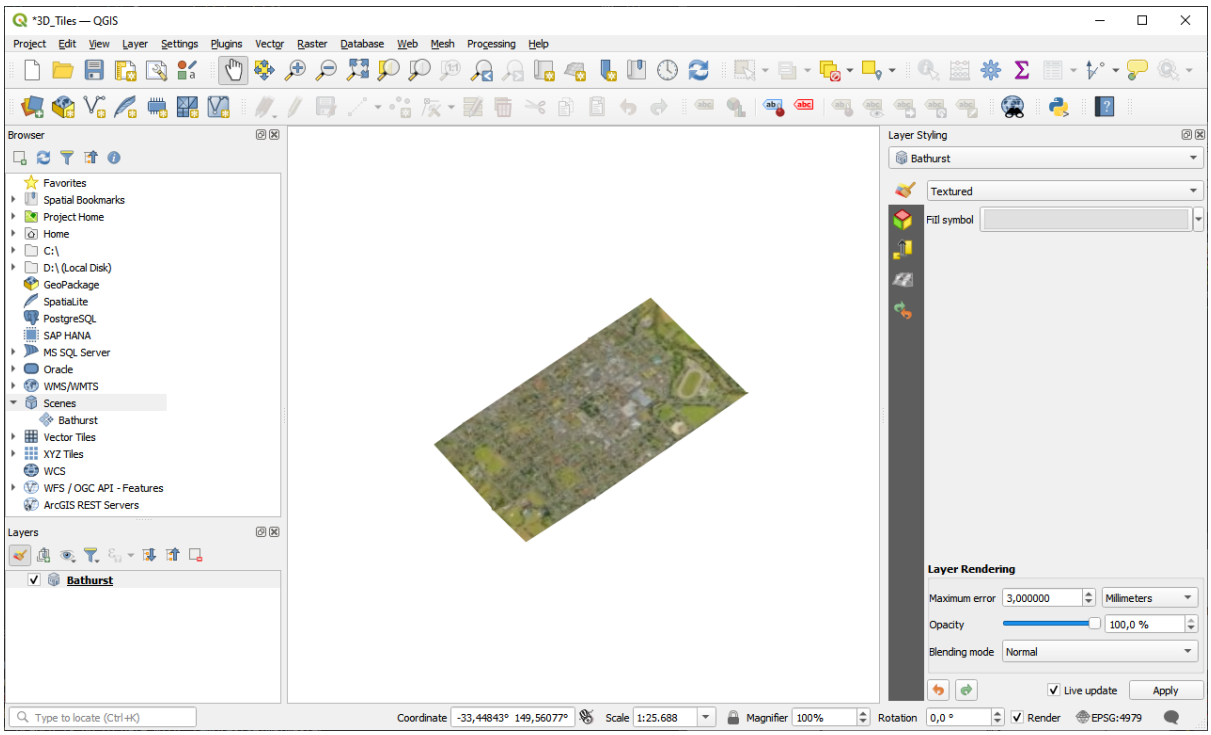


Abb. 11.23: 3D-Kachel-Layer - Texturiert

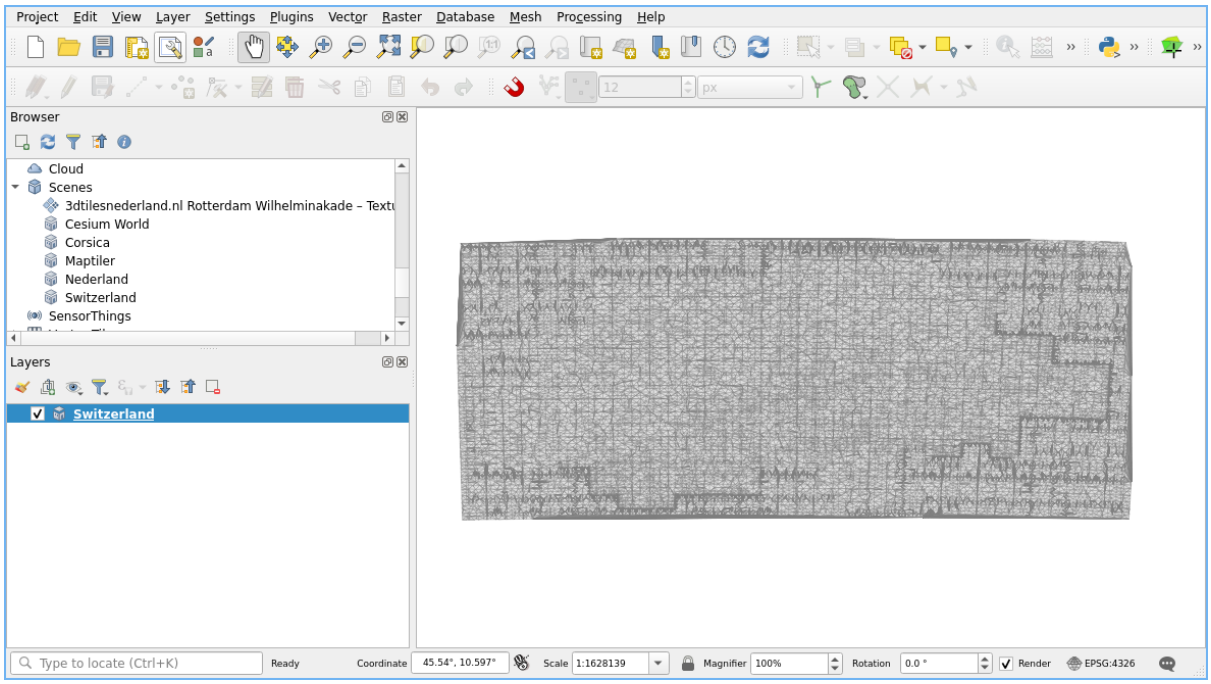


Abb. 11.24: Quantifizierter Netz-Layer

Verwendung von Cloud-Verbindungen

QGIS unterstützt Verbindungen zu Cloud-Diensten wie Alibaba Cloud OSS, Amazon S3, Google Cloud Storage, Microsoft Azure Blob Storage, Microsoft Azure Data Lake Storage und OpenStack Swift Object Storage. Sie können Vektor- und Rasterdaten aus diesen Diensten in QGIS laden. Richten Sie eine neue *Cloud*-Verbindung im *Browser*-Fenster ein, indem Sie mit der rechten Maustaste auf den Eintrag *Cloud* klicken und *Neue Verbindung* auswählen. Es wird eine Dropdown-Liste der verfügbaren Cloud-Dienste angezeigt. Wählen Sie den Dienst aus, mit dem Sie sich verbinden möchten, und füllen Sie die erforderlichen Felder aus:

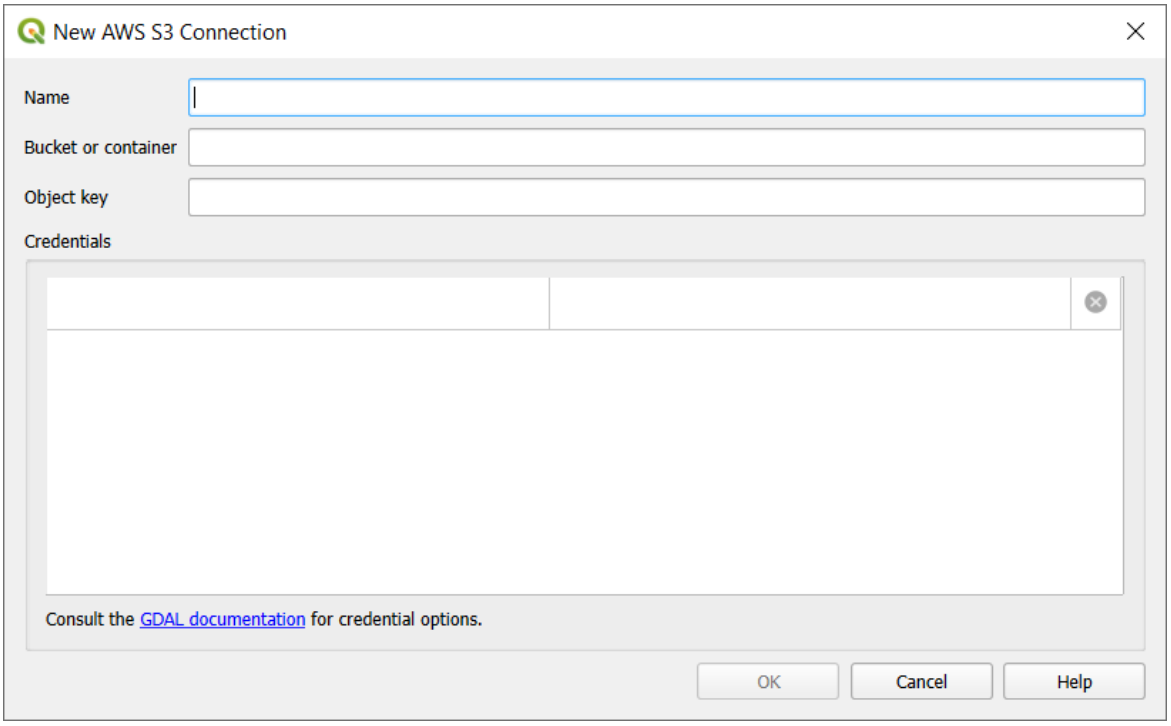


Abb. 11.25: Cloud-Verbindungsdialog

- *Name*: Ein Name für die Verbindung
- *Bucket oder Container*: Der Name des Buckets oder Containers im Cloud-Dienst
- *Objektschlüssel* (optional): Der Schlüssel des Objekts im Bucket oder Container
- *Zugangsdaten*: Die Zugangsdaten für den Zugriff auf den Cloud-Dienst

Sie können auch *Verbindung speichern* in einer XML-Datei oder *Verbindung laden* aus einer XML-Datei auswählen

11.2 Erstellung von Layern

Layer können auf viele Arten erstellt werden, unter anderem:

- leere Layer von Grund auf
- Layer von vorhandenen Layern
- Layer aus der Zwischenablage
- Layer als Ergebnis einer SQL-ähnlichen Abfrage, die auf einem oder mehreren Layern basiert: die (*virtuellen Layer*)

QGIS bietet auch Werkzeuge zum Importieren/Exportieren verschiedener Formate.

11.2.1 Neue Vektorlayer erstellen

QGIS erlaubt es Ihnen, neue Layer in verschiedenen Formaten zu erstellen. Es bietet Werkzeuge für die Erstellung von GeoPackage-, Shapefile-, SpatiaLite-, GPX-Format und flüchtige Layer (Temporärlayer). Die Erstellung eines *neuen GRASS-Layer* wird innerhalb des GRASS-Plugins unterstützt.

Einen neuen GeoPackage-Layer erstellen

Um einen neuen GeoPackage-Layer zu erstellen, klicken Sie auf die Schaltfläche *Neuer GeoPackage-Layer…* im Menü *Layer* ► *Layer erstellen* ► oder in der Symbolleiste *Datenquellenverwaltung*. Sie können einen neuen GeoPackage-Layer auch über das *Browser* Bedienfeld erstellen, indem Sie die Option *Datenbank und Layer erzeugen…* auswählen. Das Dialogfeld *Neuer GeoPackage-Layer* wird angezeigt, wie in Abb. 11.26 dargestellt.

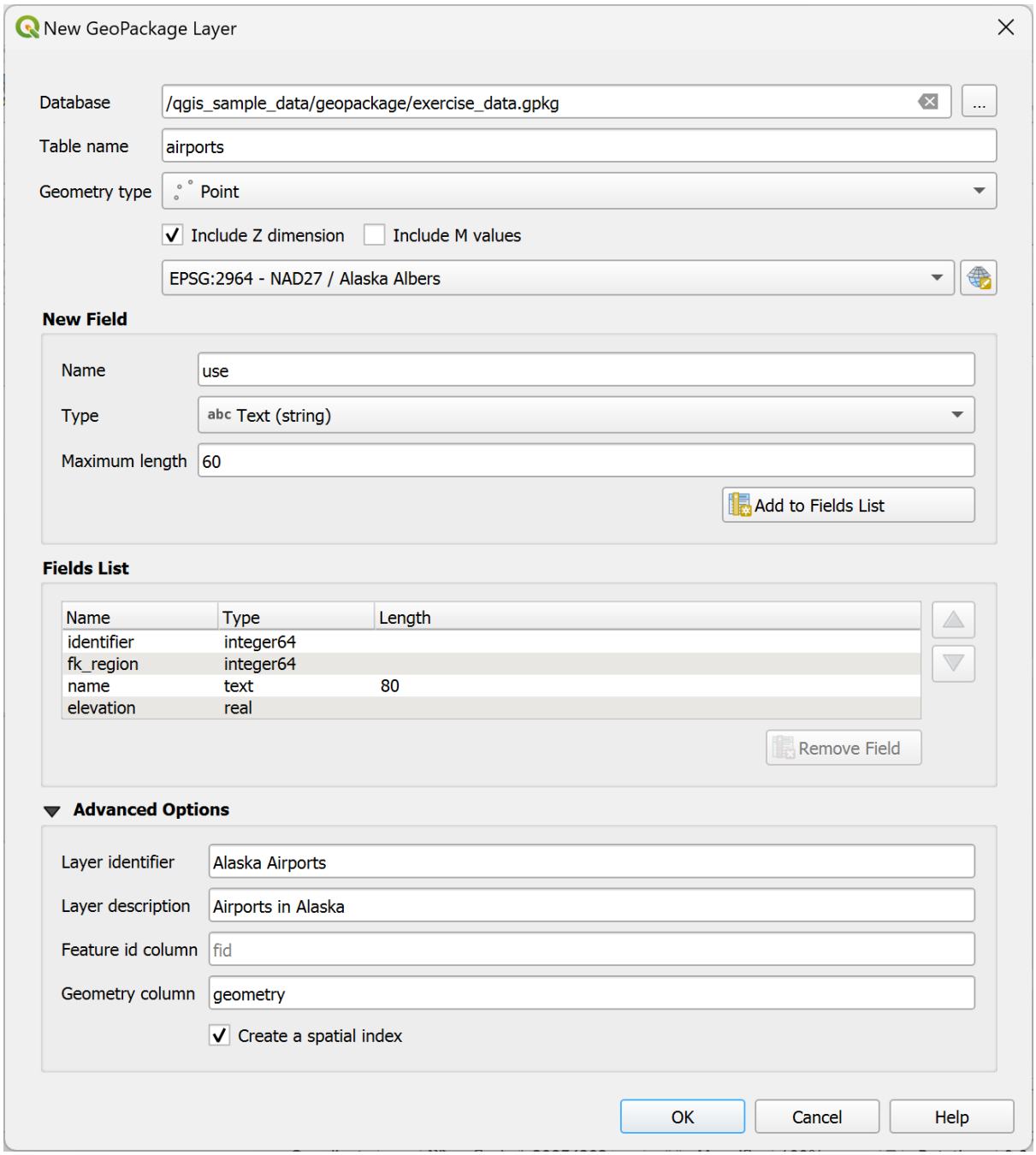


Abb. 11.26: Dialog zum Erstellen eines neuen GeoPackage-Layers

1. Der erste Schritt besteht darin, den Speicherort der Datenbankdatei anzugeben. Dies kann durch Drücken der Schaltfläche …. rechts neben dem Feld *Datenbank* erfolgen und eine vorhandene GeoPackage-Datei auswählen oder eine neue erstellen. QGIS wird automatisch die richtige Erweiterung zu dem von Ihnen angegebenen Namen hinzufügen.
2. Geben Sie dem neuen Layer/Tabelle einen Namen (*Tabellenname*)
3. Definieren Sie den *Geometrietyp*. Wenn es sich nicht um einen geometrielosen Layer handelt, können Sie angeben, ob er *Z-Dimension einschließen* und/oder *M-Werte einschließen* soll.
4. Geben Sie das Koordinatenreferenzsystem mit der Schaltfläche an

Zum Hinzufügen von Feldern zu dem Layer, den Sie erstellen:

1. Geben Sie *Name* des Feldes ein
2. Wählen Sie die Daten *Typ*. Unterstützte Typen sind *Text*, *Ganze Zahl* (sowohl Integer als auch Integer 64 bit), *Dezimalzahl*, *Datum* und *Datum & Zeit*, *Binärobjekt (BLOB)* und *Boolean*.
3. Je nach gewähltem Datenformat geben Sie die *Maximallänge* der Werte ein.
4. Klicken Sie auf die Schaltfläche *Zur Feldliste hinzufügen*.
5. Wiederholen Sie die obigen Schritte für jedes Feld, das Sie hinzufügen müssen
6. Sie können die Reihenfolge der Felder später mit den Schaltflächen Nach oben verschieben und Nach unten verschieben ändern.
7. Wenn Sie mit den Attributen zufrieden sind, klicken Sie auf *OK*. QGIS wird den neuen Layer zur Legende hinzufügen, und Sie können ihn wie in Abschnitt *Einen vorhandenen Layer editieren* beschrieben bearbeiten.

Standardmäßig erzeugt QGIS beim Erstellen eines GeoPackage-Layers eine *Objektkennungsspalte* namens `fid`, die als Primärschlüssel des Layers fungiert. Der Name kann geändert werden. Das Geometriefeld, falls verfügbar, heißt `geometry`, und Sie können wählen, ob Sie darauf einen räumlichen Index erstellen wollen. Diese Optionen sind unter dem *Erweiterte Optionen* zusammen mit dem *Layerkennung* (kurzer menschenlesbarer Name des Layers) und der *Layerbeschreibung* zu finden.

Die weitere Verwaltung von GeoPackage-Layern kann mit dem *DB-Verwaltung* erfolgen.

Einen neuen Shapefile-Layer erstellen

Um einen neuen Layer im ESRI Shapefile-Format zu erstellen, drücken Sie die Schaltfläche *Neuer Shapefile-Layer…* im Menü *Layer* ► *Layer erstellen* ► oder in der Symbolleiste *Datenquellenverwaltung*. Das Dialogfeld *Neuer Shapefile-Layer* wird angezeigt, wie in Abb. 11.27 dargestellt.

1. Geben Sie einen Pfad und einen Dateinamen an, indem Sie die Schaltfläche … neben *Dateiname* verwenden. QGIS fügt automatisch die richtige Erweiterung zu dem von Ihnen angegebenen Namen hinzu.
2. Als nächstes geben Sie die *Dateikodierung* der Daten an
3. Wählen Sie den *Geometrietyp* des Layers: Keine Geometrie (was zu einer Datei im Format `.DBF` führt), Punkt, Multipunkt, Linie oder Polygon
4. Geben Sie an, ob die Geometrie zusätzliche Dimensionen haben soll: *Keine*, *Z (+ M Werte)* oder *M Werte*.
5. Geben Sie das Koordinatenreferenzsystem mit der Schaltfläche an

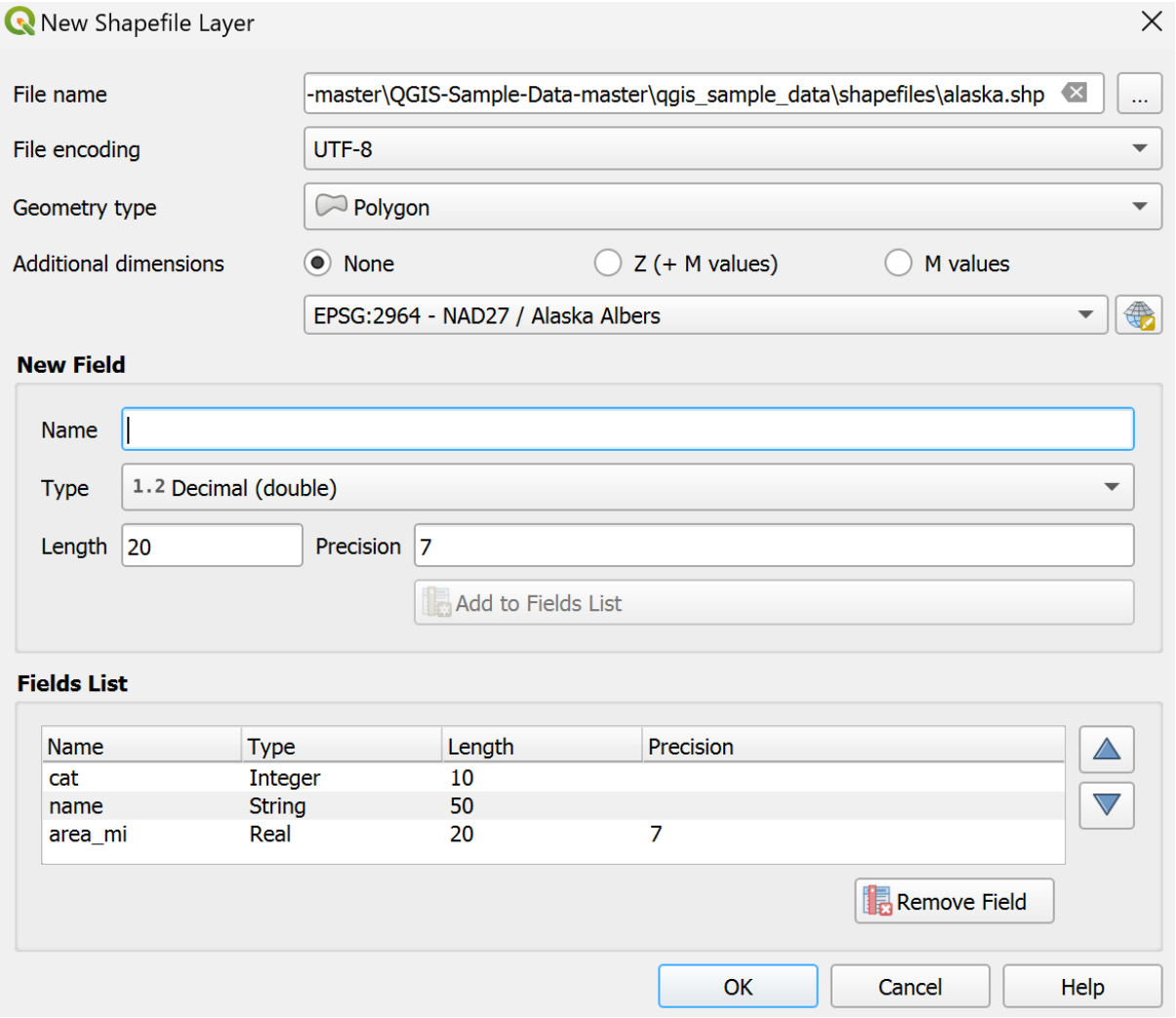


Abb. 11.27: Dialog zum Erstellen eines neuen Shapefile-Layers

Zum Hinzufügen von Feldern zu dem Layer, den Sie erstellen:

1. Geben Sie *Name* des Feldes ein
2. Wählen Sie den *Datentyp* aus. Es werden die Datentypen guilabel:*Dezimalzahl*, guilabel:*Ganzzahl*, guilabel:*Text*, guilabel:*Datum* und guilabel:*Boolean* unterstützt.
3. Je nach gewähltem Datenformat geben Sie *Länge* und *Genauigkeit* ein.
4. Klicken Sie auf die Schaltfläche *Zur Feldliste hinzufügen*.
5. Wiederholen Sie die obigen Schritte für jedes Feld, das Sie hinzufügen müssen
6. Sie können die Reihenfolge der Felder später mit den Schaltflächen Nach oben verschieben und Nach unten verschieben ändern.
7. Wenn Sie mit den Attributen zufrieden sind, klicken Sie auf *OK*. QGIS wird den neuen Layer zur Legende hinzufügen, und Sie können ihn wie in Abschnitt *Einen vorhandenen Layer editieren* beschrieben bearbeiten.

Standardmäßig wird eine erste ganzzahlige `id`-Spalte hinzugefügt, die jedoch entfernt werden kann.

Erstellen eines neuen SpatiaLite Layers

Um einen neuen SpatiaLite Layer zu erstellen, drücken Sie die Schaltfläche *Neuer SpatiaLite Layer…* im Menü *Layer* ► *Layer erstellen* ► oder in der Symbolleiste *Datenquellenverwaltung*. Das Dialogfeld *Neuer SpatiaLite Layer* wird angezeigt, wie in Abb. 11.28 dargestellt.

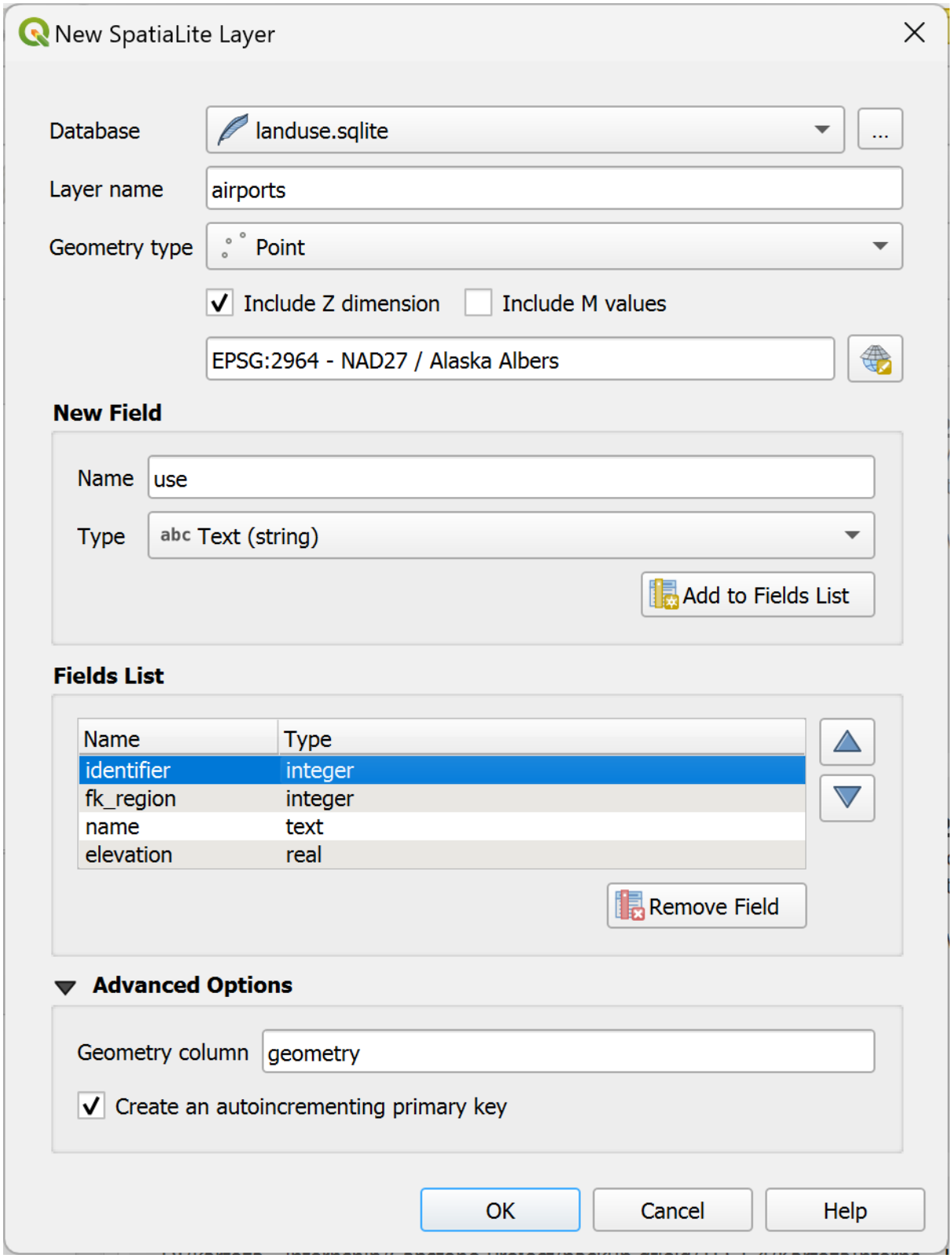


Abb. 11.28: Dialog zum Erstellen eines neuen SpatiaLite-Layers

1. Der erste Schritt besteht darin, den Speicherort der Datenbankdatei anzugeben. Klicken Sie dazu auf die Schaltfläche … rechts neben dem Feld *Datenbank* und wählen Sie eine vorhandene SpatiaLite-Datei aus oder erstellen Sie eine neue. QGIS fügt automatisch die richtige Erweiterung an den von Ihnen angegebenen Namen an.
2. Geben Sie einen Namen (*Layername*) für den neuen Layer an

3. Definieren Sie den *Geometrietyp*. Wenn es sich nicht um einen geometrielosen Layer handelt, können Sie angeben, ob er *Z-Dimension einschließen* und/oder *M-Werte einschließen* soll.
4. Geben Sie das Koordinatenreferenzsystem über die Schaltfläche an.

Zum Hinzufügen von Feldern zu dem Layer, den Sie erstellen:

1. Geben Sie *Name* des Feldes ein
2. Wählen Sie den Datentyp *Typ*. Unterstützte Typen sind *Textdaten*, *Ganzzahl*, *Dezimalzahl*, *Datum* und *Datum-Zeit*.
3. Klicken Sie auf die Schaltfläche *Zur Feldliste hinzufügen*.
4. Wiederholen Sie die obigen Schritte für jedes Feld, das Sie hinzufügen müssen
5. Sie können die Reihenfolge der Felder später mit den Schaltflächen Nach oben verschieben und Nach unten verschieben ändern.
6. Wenn Sie mit den Attributen zufrieden sind, klicken Sie auf *OK*. QGIS wird den neuen Layer zur Legende hinzufügen, und Sie können ihn wie in Abschnitt *Einen vorhandenen Layer editieren* beschrieben bearbeiten.

Falls gewünscht, können Sie unter dem Abschnitt *automatisch inkrementierenden Primärschlüssel erzeugen* das Kontrollkästchen *Erweiterte Optionen* aktivieren. Sie können auch die Spalte *Geometry* umbenennen (standardmäßig `Geometry`).

Die weitere Verwaltung von SpatiaLite Layern kann mit *DB Manager* erfolgen.

Erstellen eines neuen Netzlayers

Um einen neuen Netzlayer zu erstellen, drücken Sie die Schaltfläche *Neuer Netzlayer…* im Menü *Layer* ► *Layer erstellen* ► oder in der Symbolleiste *Datenquellenverwaltung*. Der Dialog *Neuer Netzlayer* wird angezeigt, wie in Abb. 11.29 dargestellt.

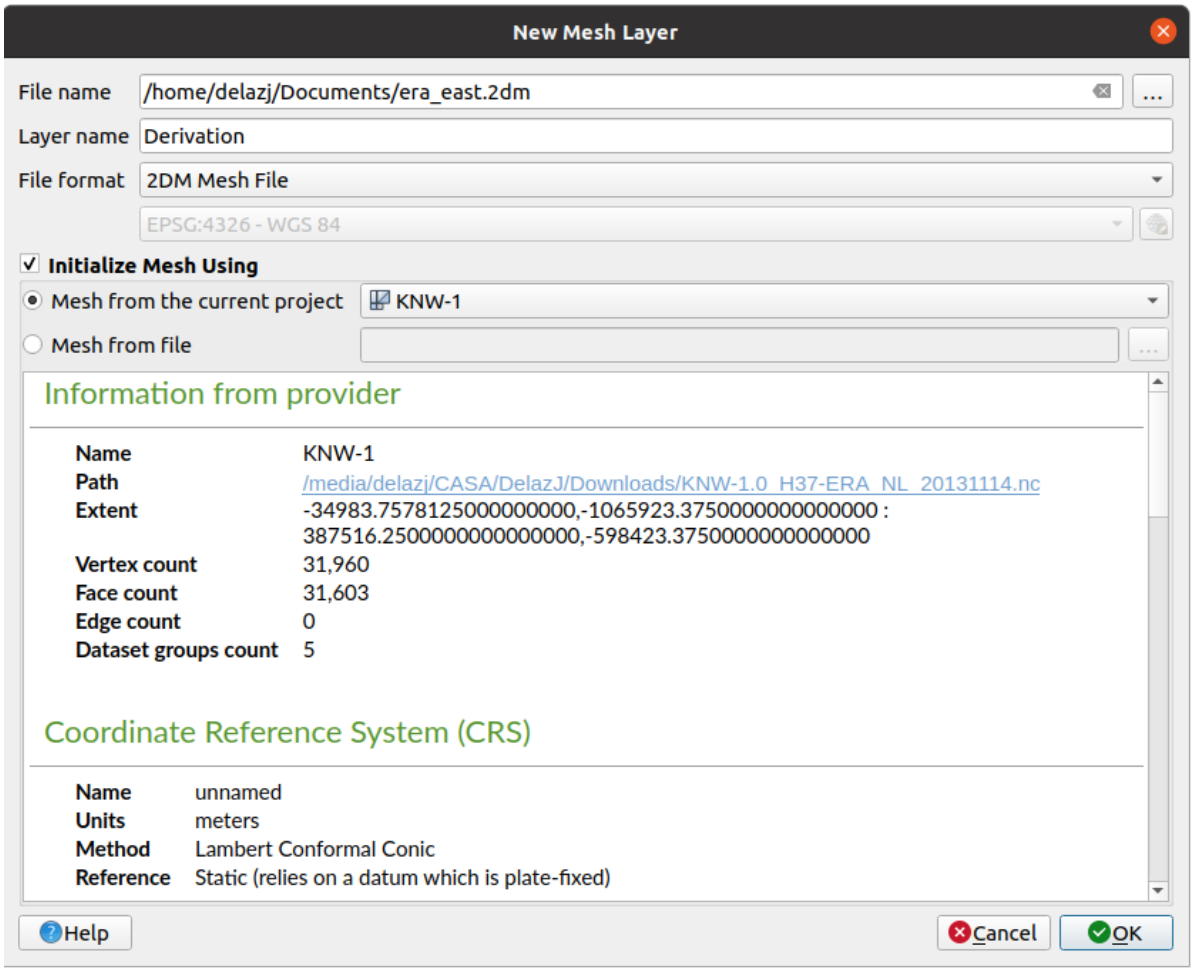


Abb. 11.29: Dialog zum Erstellen eines neuen Netzlayers

1. Der erste Schritt besteht darin, den Speicherort der Netzdatei anzugeben. Dazu drücken Sie die Schaltfläche … rechts neben dem Feld *Dateiname* und wählen eine vorhandene Netzdatei oder erstellen eine neue.
2. Geben Sie einen Namen an (*Layername*), d.h. den Namen, unter dem der Layer im Bedienfeld *Layers* angezeigt wird
3. Wählen Sie das *Dateiformat*: derzeit unterstützte Mesh-Dateiformate sind `2DM Mesh File (*.2dm)`, `Selafin File (*.slf)` und `UGRID (*.nc)`.
4. Geben Sie das *Koordinatenreferenzsystem* an, das dem Datensatz zugeordnet werden soll
5. Mit den obigen Schritten wird ein leerer Layer erzeugt, dem Sie anschließend Knotenpunkte digitalisieren und Datensatzgruppen hinzufügen können. Es ist jedoch auch möglich, den Layer mit einem bestehenden Netzlayer zu initialisieren, d.h. den neuen Layer mit Eckpunkten oder Flächen des anderen Layers zu bestücken. Um dies zu tun:

   1. Aktivieren Sie *Netz initialisieren mit*
   2. und wählen entweder ein *Netz aus dem aktuellen Projekt* oder *Netz aus Datei*. Informationen über die ausgewählte Netzdatei werden zur Überprüfung angezeigt.

   Beachten Sie, dass nur der Rahmen des Netzlayers in den neuen Layer übertragen wird; die Datensätze werden nicht kopiert.

Erstellen eines neuen GPX-Layers

Um eine neue GPX-Datei zu erstellen:

1. Wählen Sie *Layer erstellen* ► *Neuer GPX Layer…* aus dem Menü *Layer*.
2. Wählen Sie im Dialogfeld den Speicherort für die neue Datei, geben Sie ihr einen Namen und drücken Sie *Speichern*.
3. Dem Bedienfeld *Layer* werden drei neue Layer hinzugefügt:
   - ein Punkt-Layer zur Digitalisierung von Orten (`waypoints`) mit Feldern, die den Namen, die Höhe, den Kommentar, die Beschreibung, die Quelle, die Url und den Url-Namen speichern
   - ein Layer zur Digitalisierung von Routenabfolgen, die eine geplante Strecke bilden (`routes`), mit Feldern, die den Namen, das Symbol, die Nummer, den Kommentar, die Beschreibung, die Quelle, die Url und den Url-Namen speichern
   - und einen Linien-Layer, um die Bewegung des Empfängers über die Zeit zu verfolgen (`Tracks`) mit Feldern, die den Namen, das Symbol, die Nummer, den Kommentar, die Beschreibung, die Quelle, die URL und den URL-Namen speichern.
4. Sie können nun jeden dieser Layer bearbeiten, wie im Abschnitt *Einen vorhandenen Layer editieren* beschrieben.

Erstellen eines neuen temporären Scratch Layers

Temporäre Scratch Layer sind In-Memory Layer, d.h. sie werden nicht auf der Festplatte gespeichert und werden verworfen, wenn QGIS geschlossen wird. Sie können nützlich sein, um Features zu speichern, die Sie vorübergehend benötigen, oder als Zwischenlayer bei Geoverarbeitungsoperationen.

Um einen neuen Temporärerlayer zu erstellen, wählen Sie den Eintrag *Neuer Temporärerlayer…* im Menü *Layer* ► *Layer erstellen* ► oder in der Symbolleiste *Datenquellenverwaltung*. Der Dialog *Neuer Temporärerlayer* wird angezeigt, wie in Abb. 11.30 dargestellt. Dann:

1. Geben Sie den *Layernamen* an
2. Wählen Sie den *Geometrietyp*. Hier können Sie eine erstellen:
   - Layer vom Typ „Keine Geometrie“, der als einfache Tabelle dient,
   - Layer `Punkt` oder `Multipunkt`,
   - Layer `CompoundCurve` oder `MultiCurve`,
   - `Polygon/CurvePolygon` oder `MultiPolygon/MultiSurface` Layer.
3. Bei geometrischen Typen geben Sie die Dimensionen des Datensatzes an: Prüfen Sie, ob er *Z-Dimension einschließen* und/oder *M-Werte einschließen* soll.
4. Geben Sie das Koordinatenreferenzsystem über die Schaltfläche an.
5. Fügen Sie Felder zu dem Layer hinzu. Beachten Sie, dass im Gegensatz zu vielen anderen Formaten temporäre Layer ohne Felder erstellt werden können. Dieser Schritt ist daher optional.
   1. Geben Sie *Name* des Feldes ein
   2. Wählen Sie die Daten *Typ*: *Text*, *Ganzzahl*, *Dezimalzahl*, *Boolean*, *Datum*, *Zeit*, *Datum & Zeit*, *Binär (BLOB)* und andere werden unterstützt.
   3. Je nach gewähltem Datenformat geben Sie die *Länge* und *Genauigkeit* ein
   4. Klicken Sie auf die Schaltfläche *Zur Feldliste hinzufügen*.
   5. Wiederholen Sie die obigen Schritte für jedes Feld, das Sie hinzufügen möchten

6. Sie können die Reihenfolge der Felder später mit den Schaltflächen Nach oben verschieben und Nach unten verschieben ändern.

6. Wenn Sie mit den Einstellungen zufrieden sind, klicken Sie auf *OK*. QGIS fügt den neuen Layer zum *Layer* Bedienfeld hinzu, und Sie können ihn wie im Abschnitt *Einen vorhandenen Layer editieren* beschrieben bearbeiten.

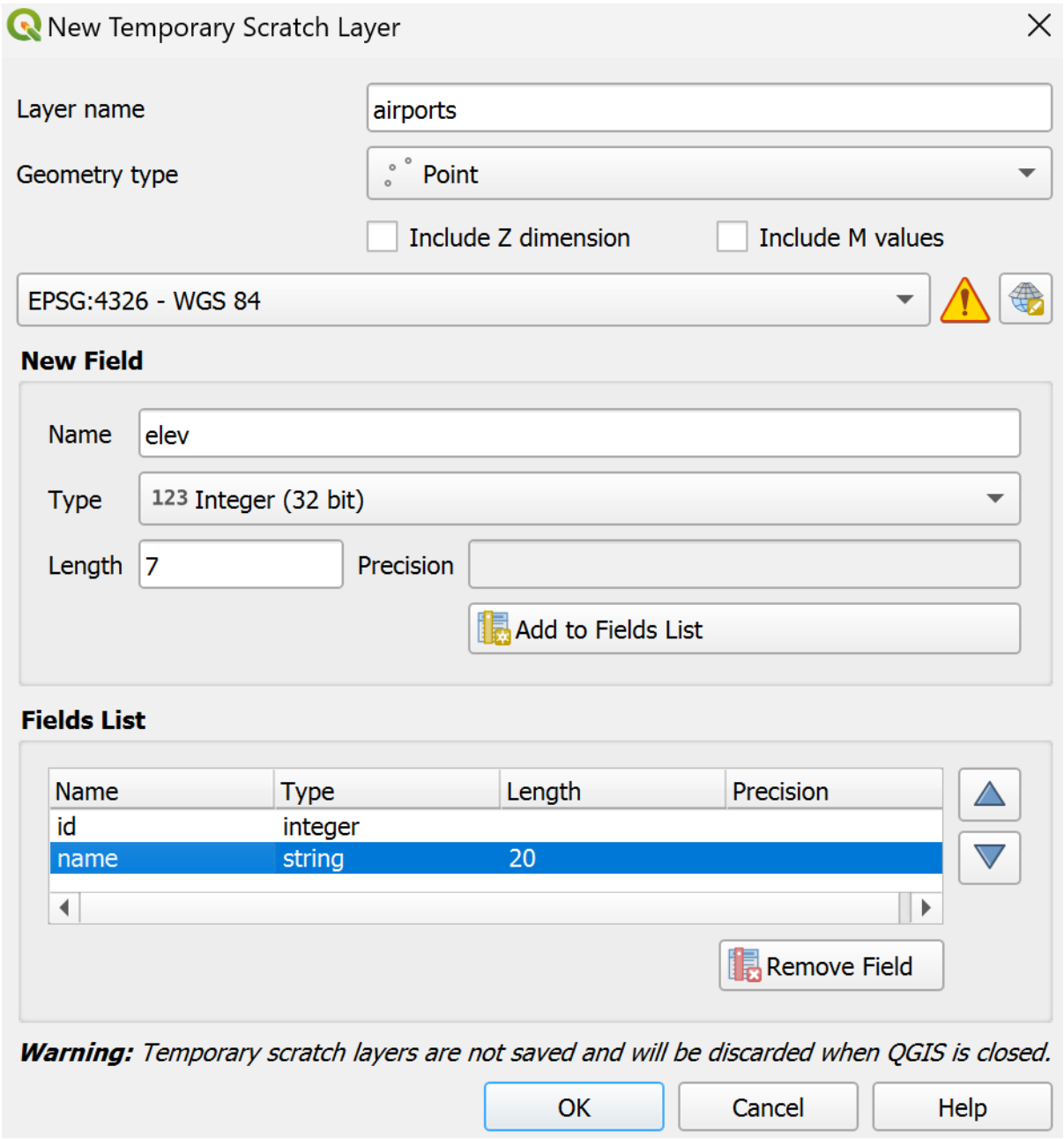


Abb. 11.30: Dialog zum Erstellen eines neuen Temporärlayers

Sie können auch vorgefertigte Temporärlayer erstellen, z. B. über die Zwischenablage (siehe *Erstellen neuer Layer über die Zwischenablage*) oder als Ergebnis eines *Verarbeitungsalgorithmus*.

Tipp

Permanente Speicherung eines Layers auf der Festplatte

Um Datenverluste beim Schließen eines Projekts mit Temporärlayern zu vermeiden, können Sie diese Layer in jedem von QGIS unterstützten Vektorformat speichern:

- Klicken Sie auf das Symbol neben dem Layer;
- Auswahl des Eintrags *Permanent machen* im Kontextmenü des Layers;
- über den Eintrag *Export* ► aus dem Kontextmenü oder das Menü *Layer* ► *Speichern unter…*.

Jeder dieser Befehle öffnet den Dialog *Vektorlayer speichern unter*, der im Abschnitt *Erstellen neuer Layer aus einem vorhandenen Layer* beschrieben ist, und die gespeicherte Datei ersetzt die temporäre Datei im *Layer* Bedienfeld.

11.2.2 Erstellen neuer Layer aus einem vorhandenen Layer

Layer (Raster, Vektor und Punktwolke) können in einem anderen Format gespeichert und/oder in ein anderes Koordinatenreferenzsystem (CRS) umprojiziert werden, indem Sie das Menü *Layer* ► *Speichern als:* verwenden oder mit der rechten Maustaste auf den Layer im *Layer-Panel* klicken und folgende Option auswählen:

- *Export –Export* ► *Speichern unter…* für Raster- und Punktwolkenebenen
- *Export* ► *Objekte speichern unter…* oder *Export* ► *Ausgewählte Objekte speichern unter…* für Vektor-Layer.
- 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*.

Gemeinsame Parameter

Der *Layer speichern unter…* Dialog zeigt verschiedene Parameter an, um das Verhalten beim Speichern des Layers zu ändern. Zu den gemeinsamen Parametern für Raster und Vektor gehören:

- *Dateiname*: der Speicherort der Datei auf der Festplatte. Er kann sich auf den ausgegebenen Layer oder auf einen Container beziehen, der den Layer speichert (z. B. datenbankähnliche Formate wie GeoPackage, SpatiaLite oder Open Document Spreadsheets).
- *KBS*: kann geändert werden, um die Daten neu zu projizieren
- *Ausdehnung*: schränkt die Ausdehnung der Eingabe ein, die mit dem *extent_selector* Widget exportiert werden soll
- *Gespeicherte Datei zur Karte hinzufügen*: zum Hinzufügen des neuen Layers zur Kartenansicht

Einige Parameter sind jedoch für bestimmte Formate spezifisch:

Rasterspezifische Parameter

Je nach Exportformat sind einige dieser Optionen möglicherweise nicht verfügbar:

- *Ausgabemodus* (kann **Rasterrohdaten** oder **Wie dargestellt** sein)
- *Format*: exportiert in jedes Rasterformat, in das GDAL schreiben kann, wie GeoTiff, GeoPackage, MBTiles, Geospatial PDF, SAGA GIS Binary Grid, Intergraph Raster, ESRI .hdr Labelled…
- *Auflösung*
- *Erzeugungsoptionen*: verwenden Sie erweiterte Optionen (Dateikomprimierung, Blockgrößen, Farbmetrik…) beim Erzeugen von Dateien, entweder aus den *vordefinierten Erstellungsprofilen*, die sich auf das Ausgabeformat beziehen, oder indem Sie jeden Parameter einstellen.
- *Pyramiden* Erstellung
- *VRT Kacheln* falls Sie sich für *VRT erzeugen* entschieden haben

- *Leerwerte*

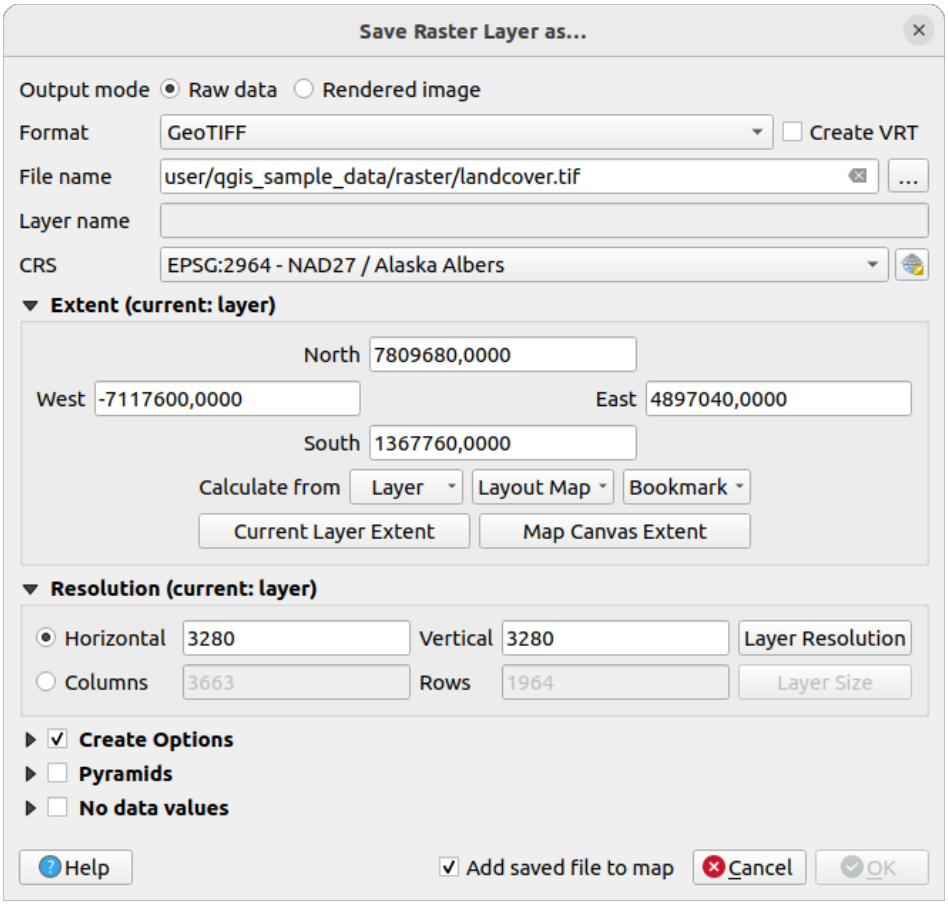


Abb. 11.31: Speichern als neuer Raster Layer

Vektorspezifische Parameter

Je nach Exportformat können einige dieser Optionen verfügbar sein:

- *Format*: exportiert in jedes Vektorformat, in das GDAL schreiben kann, wie GeoPackage, GML, ESRI Shapefile, AutoCAD DXF, ESRI FileGDB, Mapinfo TAB oder MIF, SpatiaLite, CSV, KML, ODS, …
- *Layername*: Dieser Eintrag steht zur Verfügung, wenn der *Dateiname* auf ein containerähnliches Format verweist und stellt den Layer der Ausgabe dar.
- *Kodierung*
- *Nur gewählte Objekte speichern*
- *Felder zum Export und deren Exportoptionen wählen*: bietet die Möglichkeit, Felder mit benutzerdefinierten *Namen und Einstellungen* zu exportieren:
  - Markieren Sie die Zeilen unter der Spalte *Name*, um die Felder auszuwählen, die im Output Layer bleiben sollen, oder drücken Sie die Schaltflächen *Alle wählen* oder *Alle abwählen*.
  - Aktivieren Sie das Kontrollkästchen *Aliase für exportierten Namen verwenden*, um die Spalte *Exportname* mit entsprechenden Feldaliasen zu füllen oder auf den ursprünglichen Feldnamen zurückzusetzen. Durch einen Doppelklick auf eine Zelle kann der Name ebenfalls bearbeitet werden.
  - Je nachdem, ob benutzerdefinierte Widgets für Attributformulare verwendet werden, können Sie *Alle ausgewählten Rohfeldwerte durch angezeigte Werte ersetzen*. Wenn z.B. ein `value map` Widget auf ein Feld angewandt wird, wird der Output Layer die Darstellungswerte anstelle der Originalwerte enthalten. Die Ersetzung kann auch feldweise in der Spalte *Ersetzen durch angezeigte Werte* vorgenommen werden.

- *Layermetadaten erhalten*: stellt sicher, dass alle im Quell-Layer vorhandenen *Metadaten* kopiert und gespeichert werden:
  - in dem neu erstellten Layer, wenn die Ausgabe im GeoPackage-Format erfolgt
  - bei anderen Formaten als `.qmd`-Datei zusammen mit dem Output Layer. Beachten Sie, dass dateibasierte Formate, die mehr als einen Datensatz unterstützen (z.B. SpatiaLite, DXF,…) ein unbeabsichtigtes Verhalten aufweisen können.
- *Symbologieexport*: kann hauptsächlich für den DXF-Export und für alle Dateiformate verwendet werden, die OGR-Objektstile (siehe Hinweis unten) als DXF-, KML-, Tabulator-Dateiformate verwenden:
  - **Keine Symbologie**: Standardstil der Anwendung, die die Daten liest
  - **Feature-Symbologie**: Stil mit OGR-Feature-Styles speichern (siehe Hinweis unten)
  - **Symbol-Layer-Symbologie**: Speichern mit OGR-Objektstilen (siehe Hinweis unten), aber mehrfaches Exportieren der gleichen Geometrie, wenn mehrere Symbol-Layer verwendet werden
  - Ein **Maßstab**-Wert kann auf die letzten Optionen angewendet werden

Bemerkung

OGR Feature Styles sind eine Möglichkeit, den Stil direkt in den Daten als verborgenes Attribut zu speichern. Nur einige Formate können diese Art von Informationen verarbeiten. Die Dateiformate KML, DXF und TAB sind solche Formate. Weitere Einzelheiten finden Sie in der Spezifikation der *OGR Feature Styles <https://gdal.org/en/latest/user/ogr_feature_style.html>`_.

- *Geometrie*: Sie können die Geometriefähigkeiten des Output Layers konfigurieren
  - *Geometrietyp*: behält die ursprüngliche Geometrie der Objekte bei, wenn sie auf **Automatisch** gesetzt ist, andernfalls wird sie entfernt oder durch einen beliebigen Typ ersetzt. Sie können eine leere Geometriespalte zu einer Attributtabelle hinzufügen und die Geometriespalte eines räumlichen Layers entfernen.
  - guilabel:*Mehrfachtyp erzwingen*: Erzwingt die Erstellung von Objekten mit mehreren Geometrien in dem Layer.
  - *Z-Dimension einschließen*

Tipp

Das Überschreiben des Layer-Geometrietyps macht es möglich, eine geometriefreie Tabelle (z.B. eine `.csv`-Datei) in ein Shapefile MIT beliebiger Geometrie (Punkt, Linie, Polygon) zu speichern, sodass Geometrien dann manuell mit dem Werkzeug Teil hinzufügen zu Zeilen hinzugefügt werden können.

- *Datenquellen-Optionen*, *Layeroptionen* oder *Benutzeroptionen*, mit denen Sie je nach Ausgabeformat erweiterte Parameter konfigurieren können. Einige sind in *Datenformate und Felder erkunden* beschrieben, aber für alle Details siehe die GDAL Treiberdokumentation. Jedes Dateiformat hat seine eigenen benutzerdefinierten Parameter, z.B. für das `GeoJSON`-Format schauen Sie in die GDAL GeoJSON Dokumentation.

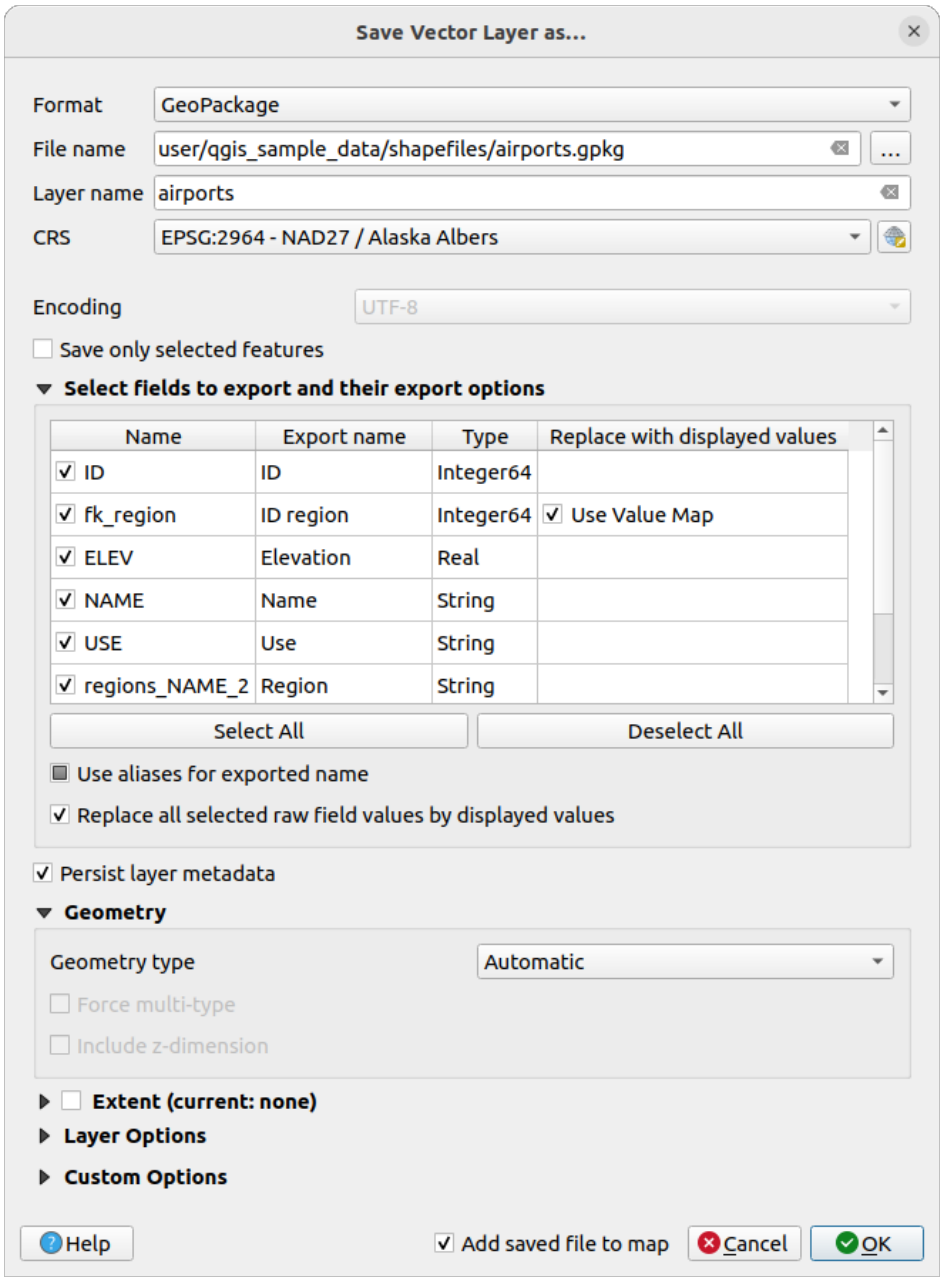


Abb. 11.32: Speichern als neuer Vektor Layer

Beim Speichern eines Vector Layers in eine bestehende Datei kann der Benutzer, abhängig von den Möglichkeiten des Ausgabeformats (Geopackage, SpatiaLite, FileGDB…), entscheiden, ob er:

- die gesamte Datei überschreiben
- nur den Ziel-Layer überschreiben (der Name des Layers ist konfigurierbar)
- Objekte an den bestehenden Layer anhängen
- Objekte anhängen, neue Felder hinzufügen, wenn es welche gibt.

Für Formate wie ESRI Shapefile, MapInfo .tab ist das Anhängen von Objekten ebenfalls möglich.

Punktwolken-spezifische Parameter

Ähnlich wie Raster- und Vektor-Layer können Punktwolken-Layer in einem anderen Format gespeichert und/oder auf ein anderes Koordinatenreferenzsystem (KBS) umprojiziert werden. Auf diese Weise können Sie einen Punktwolken-Layer in Vektor- oder Punktwolkenformate exportieren. Derzeit werden folgende Formate unterstützt: Temporärer Arbeitsspeicher (temporärer Layer), GeoPackage, ESRI Shapefile, DXF und LAS/LAZ-Punktwolke. Zusätzlich zu den oben aufgeführten allgemeinen Parametern stehen beim Exportieren von Punktwolkenebenen folgende Optionen zur Verfügung:

- *Über Polygon-Layer filtern*: Ermöglicht es Ihnen, die Punktwolkendaten basierend auf einem Polygon-Layer zu filtern.
- *Höhenbereich*: Ermöglicht das Filtern der Punktwolkendaten basierend auf einem angegebenen Z-Bereich.
- *Anzahl der Punkte begrenzen*: Bietet eine Option zum Begrenzen der Anzahl der aus der Punktwolkenebene exportierten Punkte.

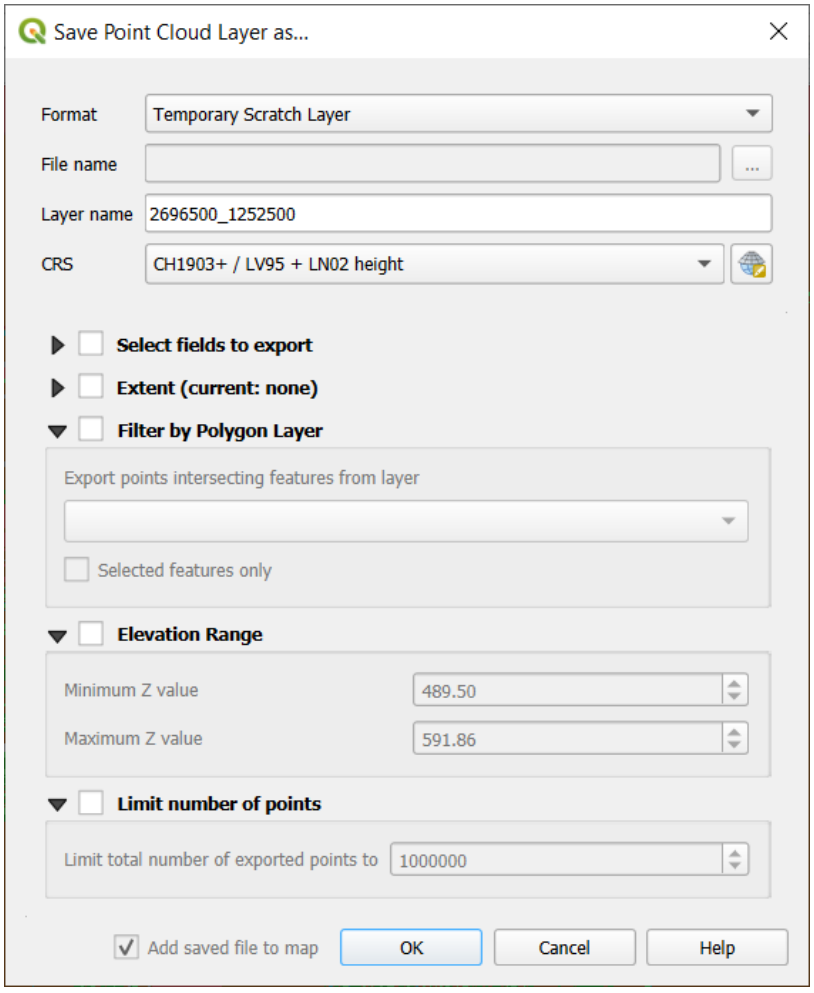


Abb. 11.33: Speichern eines Punktwolkenlayers als neuer Layer

11.2.3 Erstellen neuer DXF-Dateien

Neben dem *Speichern unter…* Dialog, der Optionen zum Export eines einzelnen Layers in ein anderes Format bietet QGIS ein weiteres Werkzeug, um mehrere Layer als eine einzelne DXF-Datei zu exportieren. Dies ist über das Menü *Projekt* ► *Import/Export* ► *Projekt als DXF speichern…* möglich.

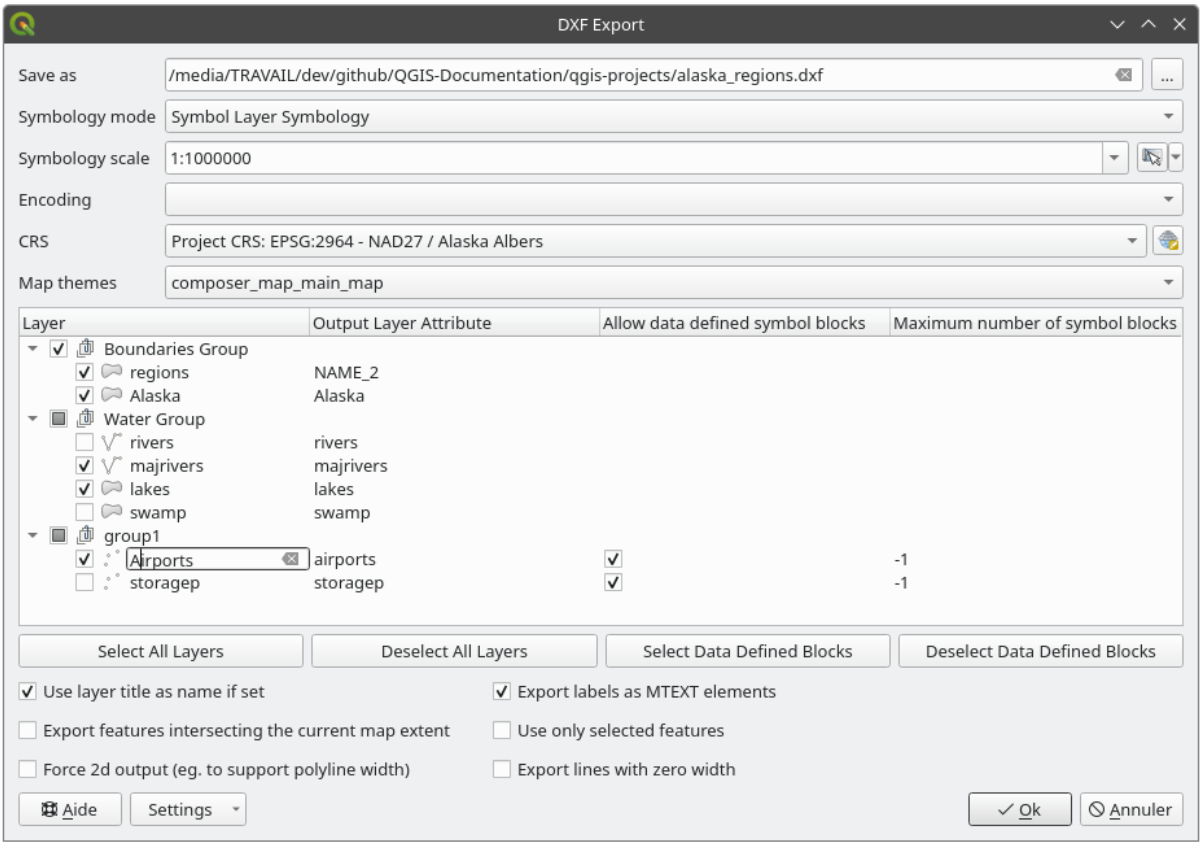


Abb. 11.34: Exportieren eines Projekts in den DXF-Dialog

Im Dialogfeld *DXF Export*:

1. Geben Sie die Zieldatei an.
2. Wählen Sie den Symbolisierungsmodus und den Maßstab (siehe Hinweis *OGR Feature Styles*), falls zutreffend.
3. Wählen Sie die Daten *Kodierung*.
4. Wählen Sie das anzuwendende *KBS*: Die ausgewählten Layer werden auf das angegebene KBS neu projiziert.
5. Wählen Sie die Layer aus, die in die DXF-Dateien aufgenommen werden sollen, indem Sie sie entweder in der Tabelle ankreuzen oder sie automatisch aus einem vorhandenen *Kartenthema* auswählen. Die Schaltflächen *Alle wählen* und *Alle abwählen* können helfen, die zu exportierenden Daten schnell festzulegen.

   Für jeden Layer können Sie:

   - den ursprünglichen Namen des Layers überschreiben (ohne ihn im Projekt zu verändern). Klicken Sie dazu im Dialogfeld auf den Namen in der Spalte *Layer* und geben Sie den gewünschten Ausgabennamen ein.
   - *Ausgabelayerattribut*: Wählen Sie aus, ob alle Objekte in einem einzigen DXF-Layer exportiert werden sollen oder ob ein Feld verwendet werden soll, dessen Werte zum Aufteilen der Objekte in Layer in der DXF-Ausgabe dienen. Im letzteren Fall erhält jeder Layer seinen Namen aus dem entsprechenden Feldwert.
   - *Erlaube datendefinierte Symbolblöcke*:

- *Maximalzahl der Symbolblöcke*: Erstellt Symbolblöcke bis zur angegebenen Grenze, beginnend mit denen, die die höchste Anzahl an Referenzen enthalten. Die anderen Symbole werden unverändert geschrieben. `-1` bedeutet keine Begrenzung.

Optional können Sie auch Folgendes wählen:

- ☐ *Layertitel als Namen verwenden, wenn vorhanden* (anstelle des Layernamens selbst): Der Titel wird aus den Eigenschaften *Metadaten* oder *Servereigenschaften* des Layers übernommen.
- ☐ *Objekte exportieren, die den aktuellen Kartenausschnitt schneiden*;
- ☐ *2d-Ausgabe erzwingen (z.B. um Polylinienstärke zu unterstützen)*;
- ☑ *Beschriftung als MTEXT exportieren* oder TEXT-Elemente;
- ☐ *Nur gewählte Objekte nutzen*;
- ☐ *Linien mit Länge 0 exportieren*: Wenn diese Option aktiviert ist, werden alle Linien mit der minimalen Breite `0` (Haarlinie) exportiert. Dadurch bleiben die Linien in der Datei unabhängig vom Zoomfaktor minimal und dies kann für die weitere CAD-Bearbeitung mit der exportierten DXF-Datei nützlich sein, insbesondere wenn viele Elemente auf der Karte dicht beieinander liegen.

Bemerkung

Die Priorität für die Definition des Namens des Ausgabelayers ist wie folgt:

1. Der Feldwert aus *Ausgabelayerattribut*
2. Der überschriebene Name in der Spalte *Layer*
3. Die Option *Layertitel als Namen verwenden, wenn vorhanden*
4. Der Layername

Die aktuellen Einstellungen, die im Dialogfeld *DXF-Export* definiert sind, können in einer XML-Datei gespeichert werden, um sie in anderen Sitzungen wiederzuverwenden. Zu diesem Zweck verfügt das Kombinationsfeld *Einstellungen* über zwei Optionen: *Einstellungen aus Datei laden…* und *Einstellungen in Datei speichern…*.

11.2.4 Erstellen neuer Layer über die Zwischenablage

Objekte, die sich in der Zwischenablage befinden, können in einen neuen Layer eingefügt werden. Wählen Sie dazu einige Objekte aus, kopieren Sie sie in die Zwischenablage und fügen Sie sie dann in einen neuen Layer ein, indem Sie „Bearbeiten –> Objekte einfügen als –>“ wählen:

- :guilabel:` Neuer Vektorlayer…`: der *Vektorlayer speichern als…* Dialog erscheint (siehe *Erstellen neuer Layer aus einem vorhandenen Layer* für Parameter)
- oder *Temporärlayer…*: müssen Sie einen Namen für den Layer angeben

Ein neuer Layer mit ausgewählten Objekten und ihren Attributen wird erstellt (und zur Kartenansicht hinzugefügt).

Bemerkung

Das Erstellen von Layern aus der Zwischenablage ist sowohl mit Objekten möglich, die in QGIS ausgewählt und kopiert wurden, als auch mit Objekten aus anderen Anwendungen, sofern deren Geometrien mit bekanntem Text (WKT) definiert wurden.

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.

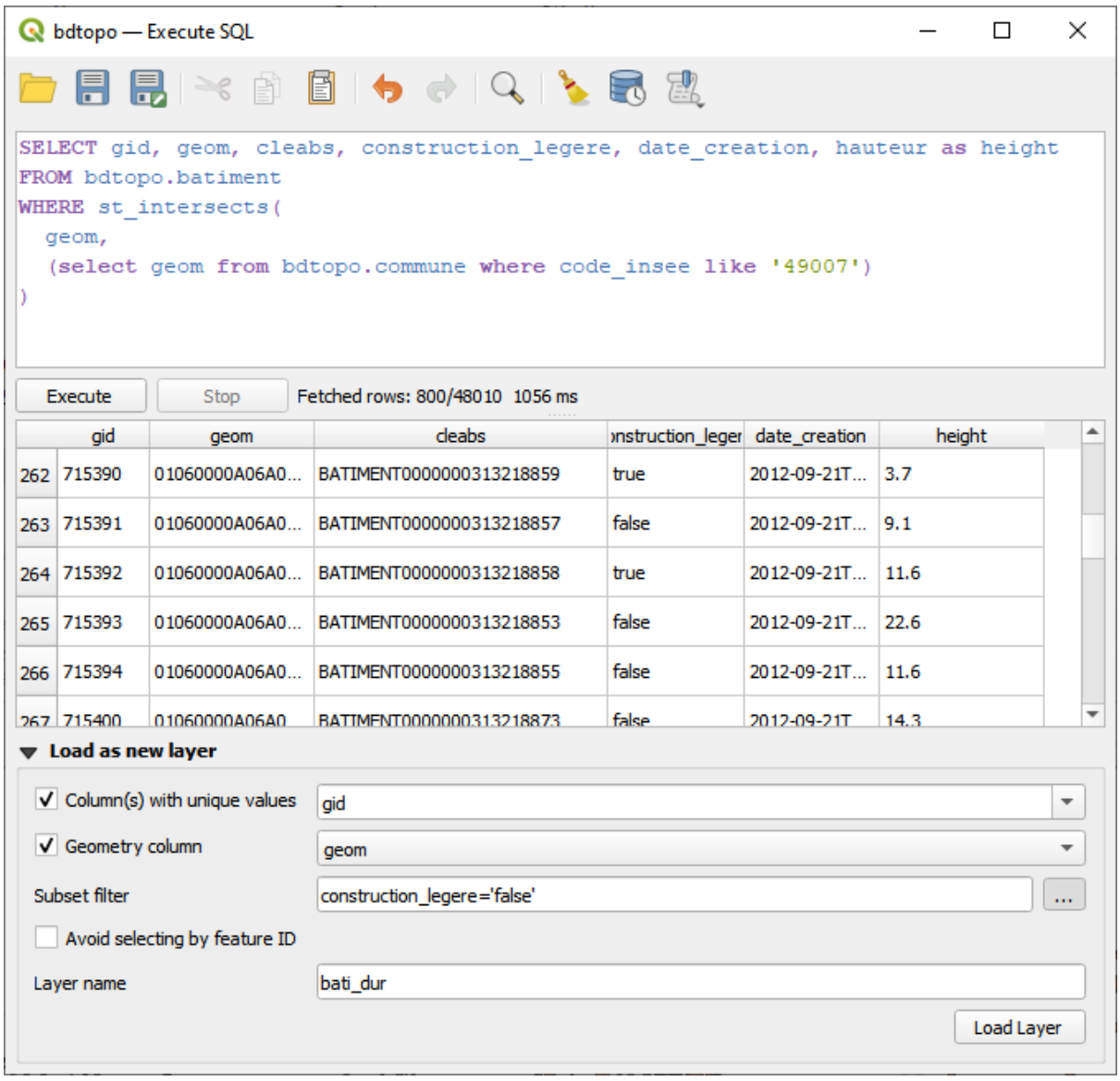


Abb. 11.35: Executing SQL queries in the Execute SQL window

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.

> **Tipp**
>
> **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.

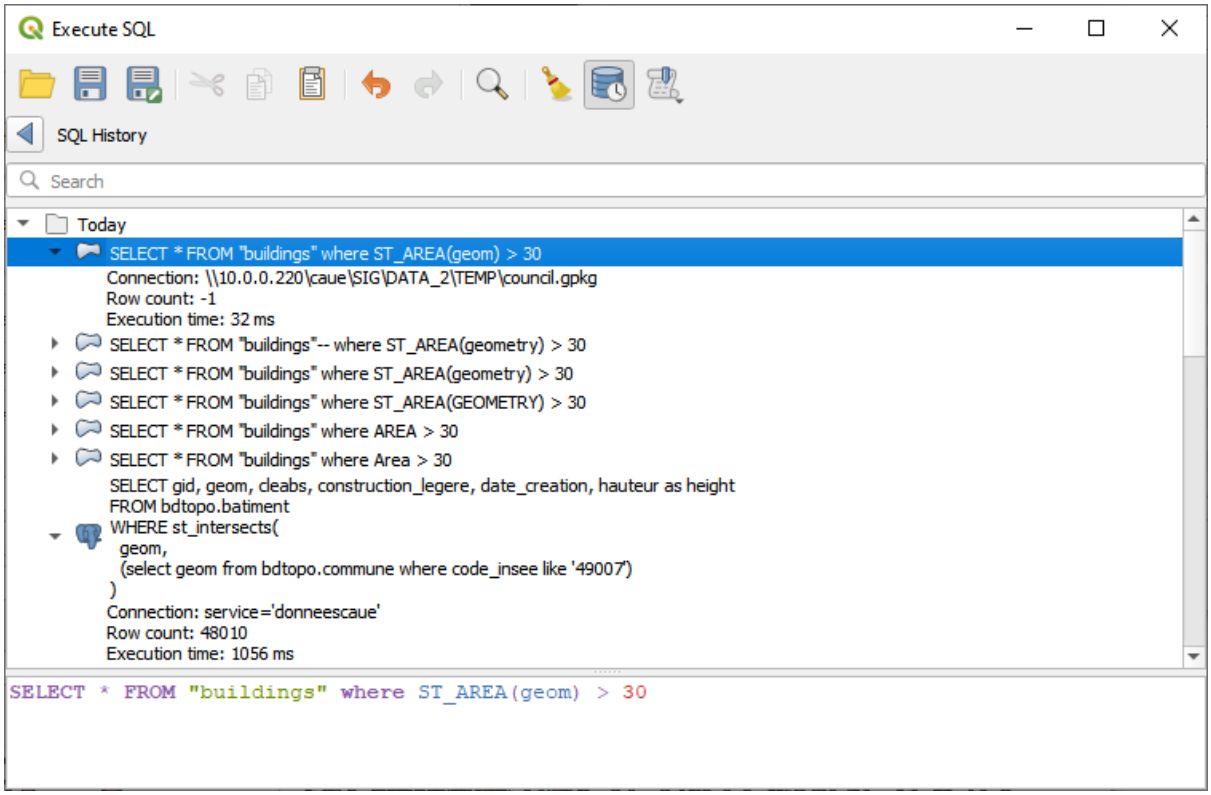


Abb. 11.36: History of the executed SQL queries

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 Erstellen von virtuellen Layern

Ein virtueller Layer ist eine besondere Art von Vektor Layer. Er ermöglicht es Ihnen, einen Layer als Ergebnis einer SQL-Abfrage zu definieren, die eine beliebige Anzahl anderer Vektorlayer einbezieht, die QGIS öffnen kann. Virtuelle Layer enthalten selbst keine Daten und können als Sichten auf andere Daten betrachtet werden.

Um einen virtuellen Layer zu erstellen, öffnen Sie den Dialog zur Erstellung eines virtuellen Layers:

- Wählen Sie den Eintrag *Virtueller Layer hinzufügen/bearbeiten ` im Menü :menuselection:`Layer –> Layer hinzufügen –>*;
- Aktivierung der Registerkarte *Virtuellen Layer hinzufügen* im Dialog *Datenquellenverwaltung*;
- oder über den entsprechenden Datenanbieter-Eintrag in der *DB-Verwaltung*.

Im Dialogfeld können Sie einen *Layernamen* und eine SQL *Abfrage* angeben. Die Abfrage kann den Namen (oder die ID) der geladenen Layer als Tabellen sowie deren Feldnamen als Spalten verwenden.

Wenn Sie zum Beispiel einen Layer mit dem Namen `airports` haben, können Sie einen neuen virtuellen Layer mit dem Namen `public_airports` mit einer SQL-Abfrage wie dieser erstellen:

```
SELECT *
FROM airports
WHERE USE = "Civilian/Public"
```

Die SQL-Abfrage wird ausgeführt, unabhängig vom zugrundeliegenden Provider des `airports` Layers, auch wenn dieser Provider SQL-Abfragen nicht direkt unterstützt.

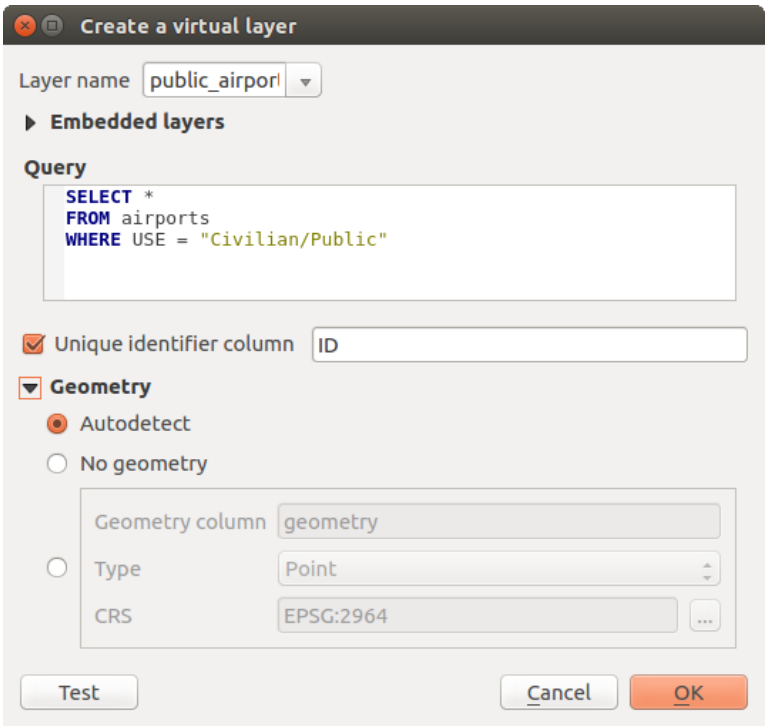


Abb. 11.37: Dialog Virtuelle Layer erstellen

Es können auch Verknüpfungen und komplexe Abfragen erstellt werden, z. B. zur Verknüpfung von Flughäfen und Länderinformationen:

```
SELECT airports.*, country.population
FROM airports
JOIN country
ON airports.country = country.name
```

Bemerkung

Es ist auch möglich, virtuelle Layer über das SQL-Fenster des *DB Verwaltung* zu erstellen.

Einbetten von Layern zur Verwendung in Abfragen

Neben den Vektorlayern, die in der Kartenansicht zur Verfügung stehen, kann der Benutzer in der Liste *Eingebettete Layer* Layer hinzufügen, die in Abfragen verwendet werden können, ohne dass sie in der Kartenansicht oder im Bedienfeld Layers angezeigt werden müssen.

Um einen Layer einzubetten, klicken Sie auf *Hinzufügen* und geben Sie den *Lokalen Namen*, *Datenanbieter*, *Kodierung* und den Pfad zur *Quelle* an.

Die Schaltfläche *Import* ermöglicht das Hinzufügen von Layern aus der Kartenansicht in die Liste der eingebetteten Layer. Diese Layer können dann aus dem *Layer* Bedienfeld entfernt werden, ohne dass bestehende Abfragen unterbrochen werden.

Unterstützte Abfragesprache

Die zugrundeliegende Engine verwendet SQLite und SpatiaLite zum Betrieb.

Das bedeutet, dass Sie können alle SQL-Ausdrücke verwenden, die auf Ihrer lokalen Installation von SQLite bereitstehen.

Funktionen aus SQLite und räumliche Funktionen aus SpatiaLite können auch in einer Abfrage für virtuelle Layer verwendet werden. Die Erstellung eines Punkt-Layers aus einem reinen Attribut-Layer kann z. B. mit einer Abfrage wie dieser erfolgen:

```
SELECT id, MakePoint(x, y, 4326) as geometry
FROM coordinates
```

:ref:`Funktionen der QGIS-Ausdrücke <functions_list> ` können auch in einer virtuellen Layer-Abfrage verwendet werden.

Um auf die Geometriespalte eines Layers zu verweisen, verwenden Sie den Namen `geometry`.

Im Gegensatz zu einer reinen SQL-Abfrage müssen alle Felder einer virtuellen Layer-Abfrage benannt werden. Vergessen Sie nicht, das Schlüsselwort `as` zu verwenden, um Ihre Spalten zu benennen, wenn sie das Ergebnis einer Berechnung oder eines Funktionsaufrufs sind.

Leistungsprobleme

Mit den Standardparametern versucht die Engine für virtuelle Layer, den Typ der verschiedenen Spalten der Abfrage zu erkennen, einschließlich des Typs der Geometriespalte, falls eine vorhanden ist.

Dies geschieht, wenn möglich, durch Introspektion der Abfrage oder, als letztes Mittel, durch Abrufen der ersten Zeile der Abfrage (LIMIT 1). Das Abrufen der ersten Zeile des Ergebnisses, nur um den Layer zu erstellen, kann aus Leistungsgründen unerwünscht sein.

Die Parameter des Erstellungsdialogs:

- *Eindeutiges Schlüsselfeld*: gibt ein Feld der Abfrage an, das eindeutige Integer-Werte darstellt, die QGIS als Zeilenbezeichner verwenden kann. Standardmäßig wird ein autoinkrementierender Integer-Wert verwendet. Die Definition einer eindeutigen Identifizierungsspalte beschleunigt die Auswahl von Zeilen nach der ID.

- guilabel:*Keine Geometrie*: erzwingt, dass der virtuelle Layer alle Geometriefelder ignoriert. Der resultierende Layer ist ein reiner Attribut-Layer.
- Geometrie *Geometriespalte*: gibt den Namen der Geometriespalte an.
- Geometrie *Typ*: gibt den Typ der Geometrie an.
- Geometrie *KBS*: gibt das Koordinatenreferenzsystem des virtuellen Layers an.

Besondere Bemerkungen

Die Engine des virtuellen Layers versucht, den Typ jeder Spalte der Abfrage zu bestimmen. Gelingt dies nicht, wird die erste Zeile der Abfrage abgerufen, um die Spaltentypen zu bestimmen.

Der Typ einer bestimmten Spalte kann direkt in der Abfrage angegeben werden, indem einige spezielle Kommentare verwendet werden.

Die Syntax lautet wie folgt: `/*: type*/`. Die Angabe muss direkt nach dem Namen einer Spalte platziert werden. `type` kann entweder `int` für Ganzzahlen, `real` für Gleitkommazahlen oder `text` sein.

Zum Beispiel:

```
SELECT id+1 as nid /*:int*/
FROM table
```

Der Typ und das Koordinatenreferenzsystem der Geometriespalte können auch durch spezielle Kommentare mit der folgenden Syntax festgelegt werden: `/*:gtype:srid*/`, wobei `gtype` der Geometrietyp ist (`point`, `linestring`, `polygon`, `multipoint`, `multilinestring` oder `multipolygon`) und `` `srid `` eine ganze Zahl, die den EPSG-Code eines Koordinatenreferenzsystems darstellt.

Verwendung von Indizes

Wenn ein Layer über einen virtuellen Layer angefordert wird, werden die Indizes der Quell-Layer wie folgt verwendet:

- wenn ein =-Prädikat für die Primärschlüsselspalte des Layers verwendet wird, wird der zugrunde liegende Datenanbieter nach einer bestimmten ID (FilterFid) gefragt
- für alle anderen Prädikate (>, <=, !=, usw.) oder für eine Spalte ohne Primärschlüssel wird eine aus einem Ausdruck gebildete Anfrage verwendet, um den zugrunde liegenden Vektordatenanbieter anzufordern. Das bedeutet, dass Indizes bei Datenbankanbietern verwendet werden können, wenn sie existieren.

Es gibt eine spezielle Syntax, um räumliche Prädikate in Anfragen zu behandeln und die Verwendung eines räumlichen Indexes auszulösen: eine versteckte Spalte namens `_search_frame_` existiert für jeden virtuellen Layer. Diese Spalte kann auf Gleichheit mit einer Bounding Box verglichen werden. Beispiel:

```
SELECT *
FROM vtab
WHERE _search_frame_=BuildMbr(-2.10,49.38,-1.3,49.99,4326)
```

Räumliche binäre Prädikate wie `ST_Intersects` werden erheblich beschleunigt, wenn sie in Verbindung mit dieser räumlichen Indexsyntax verwendet werden.

11.3 Georeferenzierung

Der Georeferenzierer ist ein Werkzeug zum Ausrichten von nicht referenzierten Raster- oder Vektorlayern auf bekannte Koordinatensysteme unter Verwendung von sogenannten Passpunkte (oder auch „GCP“ von eng.: „Ground Control Points“). Er unterstützt den Export transformierter Raster (GeoTIFF) und Vektoren (Shapefiles, Geopakete usw.) oder das Schreiben begleitender Weltdateien (nur für Raster). Der grundlegende Ansatz für die Georeferenzierung eines Layers besteht darin, Punkte festzulegen, für die Sie die Koordinaten möglichst genau bestimmen können.

Funktionen

Tab. 11.1: Georeferencer Tools

Icon	Zweck	Icon	Zweck
	Raster öffnen		Vektor öffnen
	Georeferenzierung starten		GDAL-Skript erzeugen
	Passpunkte laden		Passpunkte speichern
	Transformationseinstellungen		
	GCP-Punkt hinzufügen		GCP-Punkt löschen
	Passpunkt verschieben		Snapping Type
	Show Advanced Digitizing Dock		Verschieben
	Hinein zoomen		Heraus zoomen
	Auf den Layer zoomen		Zoom zurück
	Zoom vor		Georeferenzierung mit QGIS verbinden
	QGIS mit Georeferenzierung verbinden		Volle Histogrammstreckung
	Lokale Histogrammstreckung		

11.3.1 Georeferenzierung von Rasterlayern

Als X- und Y-Koordinaten (Grad, Minuten, Sekunden (dd mm ss.ss), Dezimalgrad (dd.dd) oder projizierte Koordinaten (mmmm.mm)), die dem ausgewählten Punkt auf dem Bild entsprechen, können zwei alternative Verfahren verwendet werden:

- Das Raster selbst liefert manchmal Kreuze mit Koordinaten, die auf das Bild „geschrieben“ sind. In diesem Fall können Sie die Koordinaten manuell eingeben.
- Verwendung von bereits georeferenzierten Layern. Dies können entweder Vektor- oder Rasterdaten sein, die dieselben Objekte/Features enthalten, die Sie auf dem Bild haben, das Sie georeferenzieren möchten, und mit der Projektion, die Sie für Ihr Bild wünschen. In diesem Fall können Sie die Koordinaten eingeben, indem Sie auf den in der QGIS Kartenansicht geladenen Referenzdatensatz klicken.

Das übliche Verfahren zur Georeferenzierung eines Bildes umfasst die Auswahl mehrerer Punkte auf dem Raster, die Angabe ihrer Koordinaten und die Auswahl eines entsprechenden Transformationstyps. Auf der Grundlage der eingegebenen Parameter und Daten berechnet der Georeferencer die Parameter der Weltdatei. Je mehr Koordinaten Sie angeben, desto besser wird das Ergebnis sein.

Wählen Sie aus der Menüleiste *Layer* ► *Georeferenzierung*. Der Georeferenzierung-Dialog erscheint wie in Abb. 11.38 gezeigt.

Für dieses Beispiel verwenden wir ein Rasterkarte von South Dakota von SDGS. Es kann später zusammen mit den Daten aus der GRASS `spearfish60` Location visualisiert werden. Sie können das Topo-Blatt hier herunterladen: https://grass.osgeo.org/sampledata/spearfish_toposheet.tar.gz.

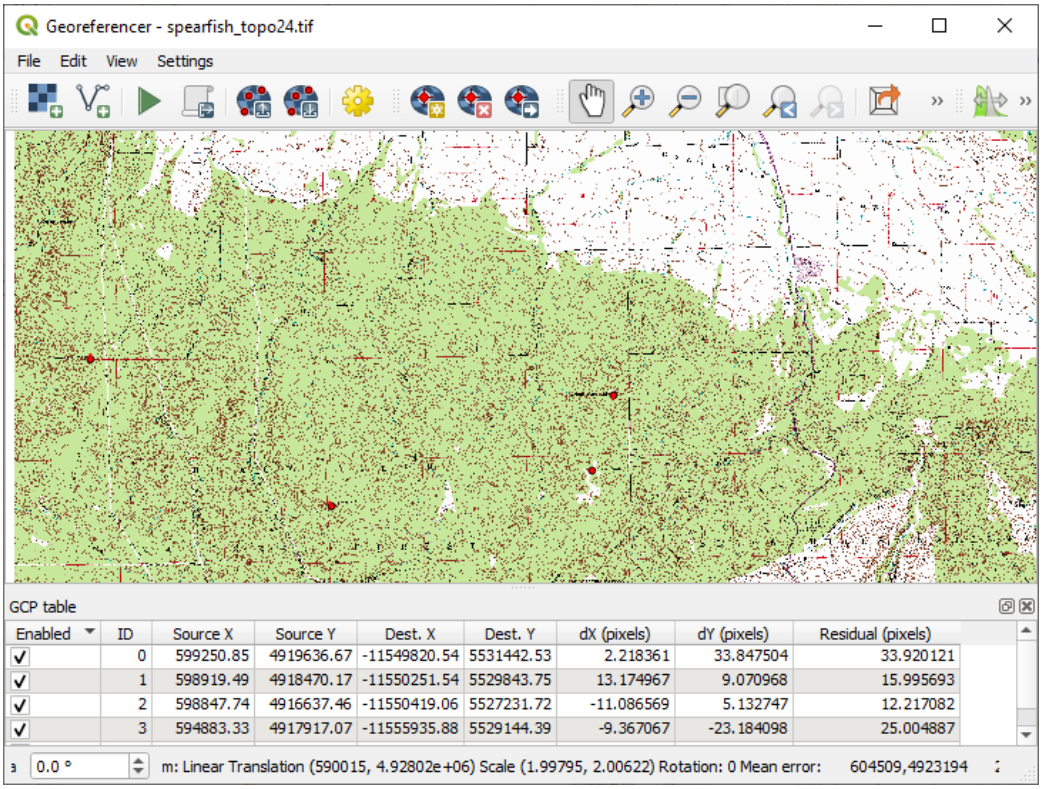


Abb. 11.38: Georeferenzierung Dialog

Eingabe von Passpunkten (GCPs)

1. Um mit der Georeferenzierung eines nicht referenzierten Rasters zu beginnen, müssen wir es über die Schaltfläche Raster öffnen… laden. Das Raster wird im Hauptarbeitsbereich des Dialogs angezeigt. Sobald das Raster geladen ist, können wir mit der Eingabe von Referenzpunkten beginnen.

2. Fügen Sie mit der Schaltfläche GCP-Punkt hinzufügen Punkte zum Hauptarbeitsbereich hinzu und geben Sie deren Koordinaten ein (siehe Abbildung Abb. 11.39). Für diesen Vorgang stehen Ihnen folgende Optionen zur Verfügung:

   - Klicken Sie auf einen Punkt im Rasterbild und geben Sie die X- und Y-Koordinaten zusammen mit dem KBS des Punktes manuell ein.
   - Klicken Sie auf einen Punkt im Rasterbild und wählen Sie die Schaltfläche Aus Kartenansicht, um die X- und Y-Koordinaten mithilfe einer bereits in die QGIS Kartenansicht geladenen georeferenzierten Karte hinzuzufügen. Das KBS wird automatisch gesetzt.
   - Bei der Eingabe von GCPs aus der Kartenansicht haben Sie die Möglichkeit, das Georeferenzierungsfenster auszublenden, während Sie Punkte aus der Kartenansicht auswählen. Wenn das Kontrollkästchen *Georeferenzierungsfensters automatisch verbergen* aktiviert ist, wird nach dem Klicken auf *Aus Kartenansicht* das Hauptfenster des Georeferenzierers ausgeblendet, bis ein Punkt auf der Kartenansicht hinzugefügt wird. Der Dialog *Kartenkoordinaten eingeben* bleibt geöffnet. Ist das Kontrollkästchen nicht markiert, bleiben beide Fenster geöffnet, während ein Punkt auf der Kartenansicht ausgewählt wird. Diese Option ist nur wirksam, wenn das Georeferenzierungsfenster nicht an der Hauptschnittstelle angedockt ist.

3. Fahren Sie mit der Eingabe der Punkte fort. Sie sollten mindestens vier Punkte haben, und je mehr Koordinaten Sie angeben können, desto besser wird das Ergebnis sein. Es gibt zusätzliche Werkzeuge zum Zoomen und Schwenken des Arbeitsbereichs, um einen relevanten Satz von GCP-Punkten zu finden.

   Tipp

   Um nicht ständig zwischen den Schaltflächen Verschieben, GCP-Punkt hinzufügen und GCP-Punkt verschieben wechseln zu müssen, können Sie die Pfeiltasten der Tastatur zum Verschieben und das Mausrad zum Skalieren der georeferenzierten Karte verwenden.

4. Nachdem Sie einige Punkte angegeben haben, können Sie die Schaltflächen QGIS mit Georeferenzierung verbinden und/oder Georeferenzierung mit QGIS verbinden verwenden, um den Kartenausschnitt des QGIS-Hauptfensters an die aktuelle Ansicht in der Georeferenzierung anzupassen und/oder umgekehrt.

5. Mit dem Werkzeug Passpunkt verschieben können Sie die GCPs sowohl in der Kartenansicht als auch im Georeferenzierungsfenster verschieben, wenn Sie sie korrigieren müssen.

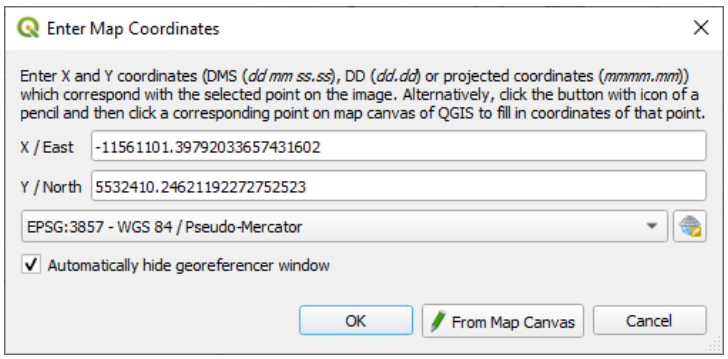


Abb. 11.39: Punkte zum Rasterbild hinzufügen

Die zur Karte hinzugefügten Punkte werden in einer separaten Textdatei (`[Dateiname].points`) gespeichert, normalerweise zusammen mit dem Rasterbild. Dies ermöglicht es uns, den Georeferencer zu einem späteren Zeitpunkt erneut zu öffnen und neue Punkte hinzuzufügen oder bestehende zu löschen, um das Ergebnis zu optimieren. Die Punktedatei enthält Werte in der Form: `mapX, mapY, pixelX, pixelY`. Sie können die Schaltflächen GCP-Punkte laden und GCP-Punkte speichern als verwenden, um die Dateien zu verwalten.

Definieren der Transformationseinstellungen

Nachdem Sie Ihre Passpunkte zum Rasterbild hinzugefügt haben, müssen Sie die Transformationseinstellungen für den Georeferenzierungsprozess festlegen.

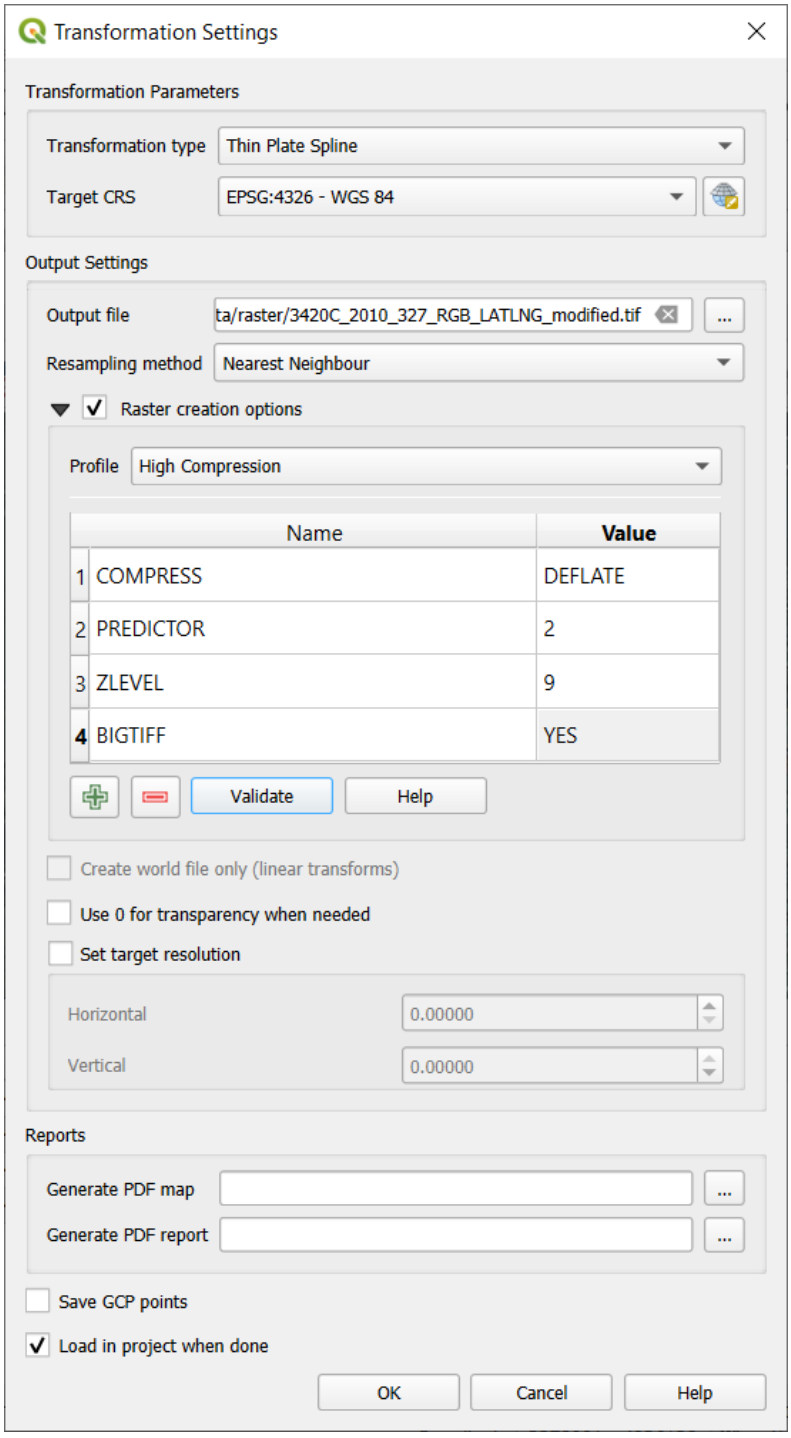


Abb. 11.40: Festlegen der Einstellungen für die Transformation der Georeferenzierung

Verfügbare Transformations-Algorithmen

Es stehen mehrere Transformationsalgorithmen zur Verfügung, die von der Art und Qualität der Eingabedaten, der Art und dem Ausmaß der geometrischen Verzerrung, die Sie für das Endergebnis in Kauf nehmen, und der Anzahl der Passpunkte (GCPs) abhängen.

Derzeit sind die folgenden *Transformationstypen* verfügbar:

- Der **Lineare** Algorithmus wird zur Erstellung einer Weltdatei verwendet und unterscheidet sich von den anderen Algorithmen, da er die Rasterpixel nicht wirklich transformiert. Er ermöglicht die Positionierung (Verschiebung) des Bildes und eine einheitliche Skalierung, aber keine Drehung oder andere Transformationen. Dieser Algorithmus ist am besten geeignet, wenn Ihr Bild eine qualitativ gute Rasterkarte in einem bekannten KBS ist, aber nur die Georeferenzierungsinformationen fehlen. Es werden mindestens 2 GCPs benötigt.
- Die **Helmert**-Transformation ermöglicht auch eine Drehung. Sie ist besonders nützlich, wenn es sich bei Ihrem Raster um eine qualitativ gute lokale Karte oder ein orthorektifiziertes Luftbild handelt, das jedoch nicht mit dem Raster in Ihrem KBS ausgerichtet ist. Es werden mindestens 2 GCPs benötigt.
- Der **Polynomial 1**-Algorithmus ermöglicht eine allgemeinere affine Transformation, insbesondere auch eine gleichmäßige Scherung. Gerade Linien bleiben gerade (d. h. kollineare Punkte bleiben kollinear) und parallele Linien bleiben parallel. Dies ist besonders nützlich für die Georeferenzierung von Datenkartogrammen, die möglicherweise mit unterschiedlichen Bodenpixelgrößen in verschiedenen Richtungen aufgezeichnet (oder erfasst) wurden. Es sind mindestens 3 GCPs erforderlich.
- Die **Polynomial**-Algorithmen 2-3 verwenden allgemeinere Polynome 2. oder 3. Grades statt nur affiner Transformationen. Dadurch können sie Krümmungen oder andere systematische Verzerrungen des Bildes berücksichtigen, z. B. fotografierte Karten mit gekrümmten Kanten. Es werden mindestens 6 (bzw. 10) GCPs benötigt. Winkel und lokaler Maßstab werden nicht beibehalten oder gleichmäßig über das Bild behandelt. Insbesondere können gerade Linien gekrümmt werden, und es kann zu erheblichen Verzerrungen an den Rändern oder weit entfernt von den GCPs kommen, wenn die an die Daten angepassten Polynome zu weit extrapoliert werden.
- Der **Projektiv** Algorithmus verallgemeinert das Polynom 1 auf eine andere Art und Weise und erlaubt Transformationen, die eine zentrale Projektion zwischen zwei nicht parallelen Ebenen, dem Bild und der Kartenansicht, darstellen. Gerade Linien bleiben gerade, aber die Parallelität wird nicht beibehalten und der Maßstab des Bildes ändert sich entsprechend der Änderung der Perspektive. Diese Transformationsart eignet sich am besten für die Georeferenzierung von schrägen Fotos (und nicht von flachen Scans) von Karten guter Qualität oder von schrägen Luftbildern. Ein Minimum von 4 GCPs ist erforderlich.
- Der Algorithmus **Thin Plate Spline** (TPS) schließlich „ verformt“ das Raster unter Verwendung mehrerer lokaler Polynome, um es an die angegebenen GCPs anzupassen, wobei die Oberflächenkrümmung insgesamt minimiert wird. Bereiche, die von den GCPs entfernt sind, werden in der Ausgabe verschoben, um die GCP-Anpassung zu ermöglichen, werden aber ansonsten nur minimal lokal deformiert. TPS ist am nützlichsten für die Georeferenzierung beschädigter, deformierter oder anderweitig leicht ungenauer Karten oder schlecht orthorektifizierter Luftbilder. Es ist auch nützlich für die ungefähre Georeferenzierung und implizite Reprojektion von Karten mit unbekanntem Projektionstyp oder unbekannten Parametern, bei denen jedoch ein regelmäßiges Raster oder ein dichter Satz von Ad-hoc-GCPs mit einer Referenzkarten-Layer abgeglichen werden kann. Technisch gesehen sind mindestens 10 GCPs erforderlich, in der Regel jedoch mehr, um erfolgreich zu sein.

Wenn bei allen Algorithmen mit Ausnahme von TPS mehr als die minimalen GCPs angegeben sind, werden die Parameter so angepasst, dass der Gesamtrestfehler minimiert wird. Dies ist hilfreich, um die Auswirkungen von Registrierungsfehlern, d. h. leichten Ungenauigkeiten bei Zeigerklicks oder getippten Koordinaten, oder anderen kleinen lokalen Bildverformungen zu minimieren. Ohne weitere GCPs zum Ausgleich könnten solche Fehler oder Verformungen zu erheblichen Verzerrungen führen, insbesondere in der Nähe der Ränder des georeferenzierten Bildes.␣ Wenn jedoch mehr als die minimalen GCPs angegeben werden, stimmen sie in der Ausgabe nur ungefähr überein. Im Gegensatz dazu passt TPS alle angegebenen GCPs genau an, kann aber zu erheblichen Verformungen zwischen nahegelegenen GCPs mit Registrierungsfehlern führen.

Definieren der Resampling-Methode

Welche Art der Neuabtastung Sie wählen, hängt wahrscheinlich von Ihren Eingabedaten und dem letztendlichen Ziel der Arbeit ab. Wenn Sie die statistischen Daten des Rasters nicht verändern wollen (abgesehen von der ungleichmäßigen geometrischen Skalierung bei Verwendung anderer Transformationen als Linear, Helmert oder Polynom 1), sollten Sie die Option „Nächster Nachbar“ wählen. Im Gegensatz dazu führt z. B. das „kubische Resampling“ in der Regel zu einem visuell glatteren Ergebnis.

Es ist möglich, zwischen fünf verschiedenen Resampling-Methoden zu wählen:

1. Nächster Nachbar
2. Bilinear (2x2 Kern)
3. Kubisch (4x4 Kern)
4. Kubisch B-Spline (4x4 Kern)
5. Lanczos (6x6 Kern)

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*.

> **Tipp**
>
> Select an empty entry if you want to create your own custom combination of parameters.

Definieren der Transformationseinstellungen

Für das georeferenzierte Ausgaberaster müssen mehrere Optionen festgelegt werden.

- Das Kontrollkästchen *Nur World-Datei erzeugen* ist nur verfügbar, wenn Sie sich für die lineare Transformation entscheiden, da dies bedeutet, dass das Rasterbild tatsächlich nicht transformiert wird. In diesem Fall ist das Feld *Ausgabedatei* nicht aktiviert, da nur eine neue Weltdatei erstellt wird.
- Für alle anderen Transformationsarten müssen Sie eine *Ausgabedatei* definieren. Standardmäßig wird eine neue Datei ([Dateiname]_modifiziert) im selben Ordner zusammen mit dem ursprünglichen Rasterbild erstellt.
- In einem nächsten Schritt müssen Sie das *Ziel-KBS* (Koordinatenreferenzsystem) für das georeferenzierte Raster definieren (siehe *Arbeiten mit Projektionen*).
- Bei Bedarf können Sie *PDF-Karte erzeugen* und auch *PDF-Bericht erzeugen* aktivieren. Der Bericht enthält Informationen über die verwendeten Transformationsparameter, ein Bild der Residuen und eine Liste mit allen GCPs und ihren RMS-Fehlern.
- Außerdem können Sie das Kontrollkästchen *Zielauflösung setzen* aktivieren und die Pixelauflösung des Ausgaberasters festlegen. Die Standardeinstellung für die horizontale und vertikale Auflösung ist 1.
- Die *Falls nötig 0 für Transparenz verwenden*-Option kann aktiviert werden, wenn Pixel mit dem Wert 0 transparent dargestellt werden sollen. In unserem Beispiel-Toposheet würden alle weißen Flächen transparent sein.
- Die *Passpunkte speichern* speichert die GCP Punkte in einer Datei neben dem Ausgaberaster.
- Schließlich lädt *Nach Fertigstellung ins Projekt laden* das Ausgaberaster automatisch in die QGIS Kartenansicht, wenn die Transformation abgeschlossen ist.

Durchführung der Transformation

Nachdem alle GCPs gesammelt und alle Transformationseinstellungen definiert wurden, drücken Sie einfach die Schaltfläche :sup:` Georeferenzierung starten`, um das neue georeferenzierte Raster zu erstellen.

11.3.2 Georeferenzierung des Vektor-Layers

Die Georeferenzierung von Vektor-Layern funktioniert ähnlich wie die Raster-Georeferenzierung, jedoch werden statt Bildpixeln Vektor-Geometrien (Punkte, Linien oder Polygone) mit bekannten räumlichen Referenzen abgeglichen.

Das Standardverfahren beginnt genauso wie bei der Raster-Georeferenzierung: Öffnen Sie QGIS und fügen Sie der Kartenfläche einen Layer hinzu, der als Referenz dienen soll. Dies kann ein georeferenzierter Raster- oder Vektor-Layer oder ein WMS-Layer sein.

Öffnen Sie das Dialogfeld „“Georeferenzierung“ über *Layer* ► *Georeferenzierung*.

Beginnen Sie mit der Georeferenzierung, indem Sie die folgenden Schritte ausführen (in diesem Beispiel verwenden wir eine nicht referenzierte Datei „alaska.shp”):

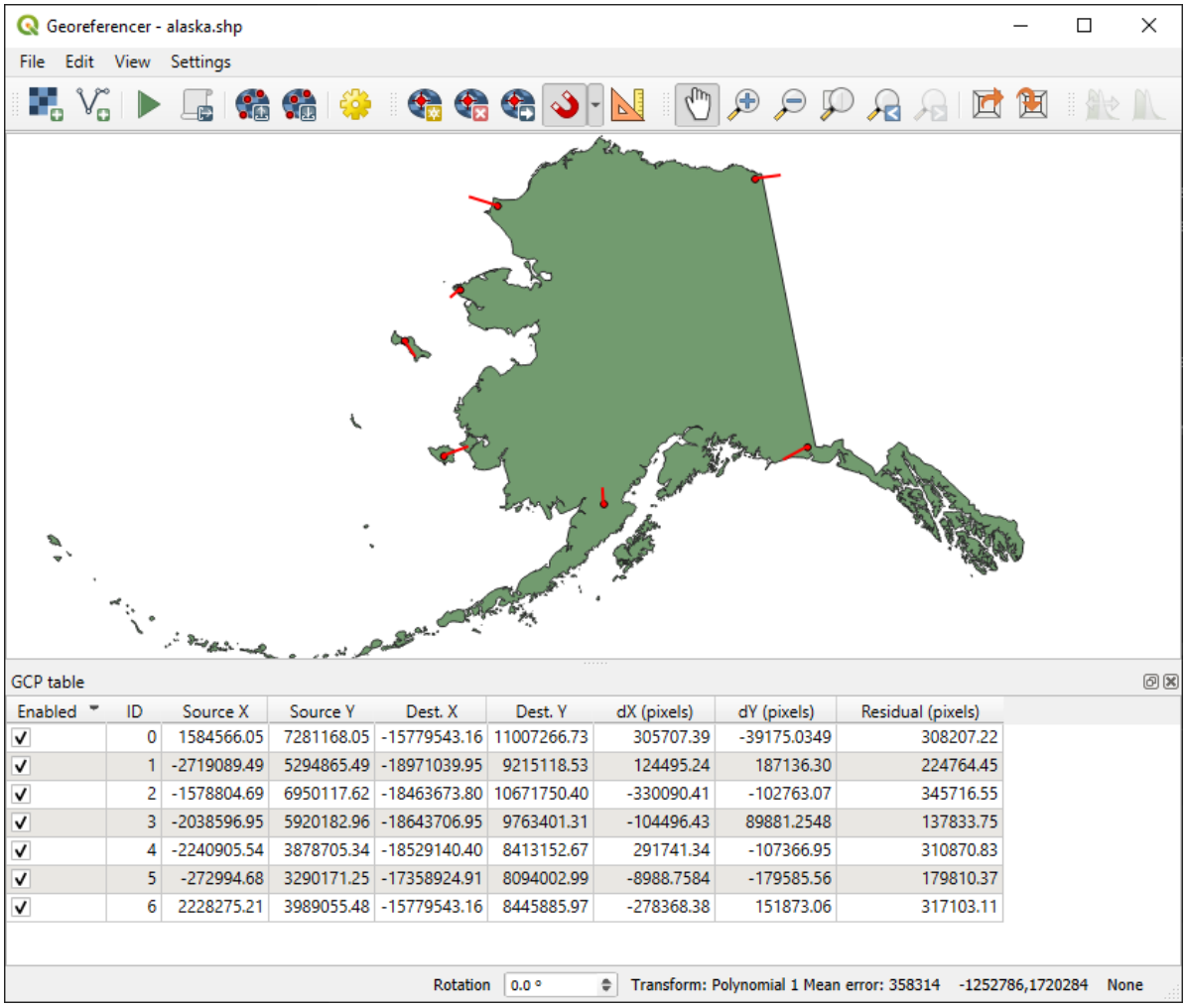


Abb. 11.41: Dialogfeld Vektor- Georeferenzierung

1. Laden Sie den nicht referenzierten Vektor-Layer mit der Schaltfläche . Der Vektor-Layer wird im Hauptarbeitsbereich des Dialogfelds angezeigt.

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. Verwenden Sie die Schaltfläche GCP-Punkt hinzufügen, um einen Punkt zum Arbeitsbereich hinzuzufügen. Geben Sie dessen Koordinaten manuell ein und legen Sie das KBS fest, oder klicken Sie auf die Schaltfläche Aus Kartenansicht, um die Koordinaten aus einem georeferenzierten Layers in der Hauptkartenansicht von QGIS auszuwählen. In diesem Fall wird das KBS automatisch festgelegt

4. Definieren Sie die Transformationseinstellungen:

   - Wählen Sie den *Transformationstyp* aus. Die Transformationsalgorithmen sind dieselben wie bei der Raster-Georeferenzierung. Weitere Informationen finden Sie unter *Definieren der Transformationseinstellungen*.
   - Definieren Sie das *Ziel-KBS* (Koordinatenbezugssystem) für den georeferenzierten Vektor (siehe *Arbeiten mit Projektionen*).
   - Legen Sie das Ausgabeformat und den Pfad der Datei fest (z. B. GeoPackage, Shapefile). Standardmäßig wird eine neue Datei mit der Endung „_modified“ im selben Ordner wie die ursprüngliche Vektordatei erstellt.
   - Optional können Sie **PDF-Karte erstellen** und **PDF-Bericht erstellen** aktivieren. Der Bericht enthält Transformationsparameter, GCP-Residuen und eine Zusammenfassung der RMS-Fehler.
   - Aktivieren Sie das Kontrollkästchen „GCP-Punkte speichern“, um die GCPs zusammen mit dem Ausgabevektor-Layer in einer Datei zu speichern.
   - Aktivieren Sie das Kontrollkästchen „Nach Fertigstellung ins Projekt laden“, um das Ergebnis direkt zur Kartenfläche hinzuzufügen.

5. Klicken Sie auf Georeferenzierung starten, um die Transformation auszuführen und den georeferenzierten Vektor-Layer zu generieren.

11.3.3 Layer-Eigenschaften anzeigen und anpassen

Durch Klicken auf die Option *Quelleneigenschaften* im Menü *Einstellungen* wird je nach Art des Layers, den Sie georeferenzieren, entweder *Rasterlayer-Eigenschaften* oder *Vektorlayer-Eigenschaften* geöffnet.

11.3.4 Konfiguration des Georeferenzierers

Sie können das Verhalten der Georeferenzierung unter *Einstellungen* ► *Georeferenzierung konfigurieren* anpassen (oder das Tastaturkürzel `Strg+P` verwenden).

- Unter *Punktspitze* können Sie die Anzeige von GCP-IDs und X/Y-Koordinaten sowohl im Georeferenzierungsfenster als auch in der Kartenansicht umschalten.
- *Resteinheiten* steuert, ob Resteinheiten in Pixeln oder Karteneinheiten angegeben werden
- *PDF-Bericht* ermöglicht es Ihnen, die Randgröße in mm für den Berichtsexport festzulegen
- *PDF-Karte* ermöglicht die Auswahl eines Papierformats für den Export der Karte
- Schließlich können Sie das Kontrollkästchen *Georeferenzierungsfenster docken* aktivieren. Dadurch wird das Georeferenzierungsfenster an das Hauptfenster von QGIS angedockt, anstatt es als separates Fenster anzuzeigen, das minimiert werden kann.

11.4 Datenformate und Felder erkunden

11.4.1 Rasterdaten

GIS-Rasterdaten sind Matrizen aus diskreten Zellen, die Objekte/Phänomene auf, über oder unter der Erdoberfläche darstellen. Jede Zelle im Rastergitter hat die gleiche Größe, und die Zellen sind normalerweise rechteckig (in QGIS sind sie immer rechteckig). Zu den typischen Rasterdatensätzen gehören Fernerkundungsdaten wie Luft- oder Satellitenbilder und modellierte Daten wie Höhenangaben oder Temperaturen.

Im Gegensatz zu Vektordaten gibt es bei Rasterdaten in der Regel nicht für jede Zelle einen zugehörigen Datenbankeintrag. Sie werden durch die Pixelauflösung und die X/Y-Koordinate eines Eckpixels des Rasterlayers geokodiert. Dies ermöglicht es QGIS, die Daten korrekt auf der Kartenansicht zu positionieren.

Das GeoPackage-Format ist für die Speicherung von Rasterdaten bei der Arbeit mit QGIS geeignet. Das gängige und leistungsstarke GeoTiff-Format ist eine gute Alternative.

QGIS nutzt die Georeferenzinformationen innerhalb des Layer (z.B. GeoTiff) oder eine zugehörige *World-Datei* zur korrekten Darstellung der Daten.

11.4.2 Vektordaten

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.

> **Bemerkung**
>
> QGIS unterstützt ((Multi)Punkt, (Multi)Linie, (Multi)Polygon, CircularString, CompoundCurve, CurvePolygon, MultiCurve, MultiSurface Objekttypen, alle optional mit Z und/oder M Werten.
>
> Sie sollten auch beachten, dass einige Treiber einige dieser Objekttypen nicht unterstützen, wie CircularString, CompoundCurve, CurvePolygon, MultiCurve, MultiSurface feature type. QGIS wird sie konvertieren.

GeoPackage

Das Format „GeoPackage <https://www.geopackage.org/>“ (GPKG) ist plattformunabhängig und wird als SQLite-Datenbankcontainer implementiert und kann sowohl für die Speicherung von Vektor- als auch von Rasterdaten verwendet werden. Das Format wurde vom Open Geospatial Consortium (OGC) definiert und im Jahr 2014 veröffentlicht.

GeoPackage kann verwendet werden, um Folgendes in einer SQLite-Datenbank zu speichern:

- **Vektor**-Objekte
- **Kachelmatrixsätze von Bildmaterial** und **Rasterkarten**
- Attribute (nicht raumbezogene Daten)
- Erweiterungen

Seit QGIS Version 3.8 kann GeoPackage auch QGIS-Projekte speichern. GeoPackage Layer können JSON Felder haben.

GeoPackage ist das Standardformat für Vektordaten in QGIS.

ESRI Shapefile-Format

Das ESRI Shapefile-Format ist nach wie vor eines der am häufigsten verwendeten Vektordateiformate, auch wenn es im Vergleich zu beispielsweise GeoPackage und SpatiaLite einige Einschränkungen aufweist.

Ein Datensatz im ESRI Shapefile-Format besteht aus mehreren Dateien. Die folgenden drei sind mindestens erforderlich:

1. `.shp` Datei mit den Geometrien der Objekte
2. `.dbf` Datei mit den Attributen im dBase-Format
3. `.shx` Indexdatei

Ein Datensatz im ESRI Shapefile-Format kann auch eine Datei mit dem Suffix `.prj` enthalten, die Projektionsinformationen enthält. Eine Projektionsdatei ist zwar sehr nützlich, aber nicht zwingend erforderlich. Ein Datensatz im Shapefile-Format kann zusätzliche Dateien enthalten. Weitere Einzelheiten finden Sie in der technischen Spezifikation von ESRI <https://www.esri.com/content/dam/esrisites/sitecore-archive/Files/Pdfs/library/whitepapers/pdfs/shapefile.pdf>`_.

GDAL bietet Lese- und Schreibunterstützung für das komprimierte ESRI Shapefile-Format (`shz` und `shp.zip`).

Verbesserung der Leistung für Datensätze im ESRI Shapefile-Format

Um die Zeichengeschwindigkeit für einen Datensatz im ESRI Shapefile-Format zu verbessern, können Sie einen räumlichen Index erstellen. Ein räumlicher Index verbessert die Geschwindigkeit beim Zoomen und Schwenken. Räumliche Indizes, die von QGIS verwendet werden, haben die Erweiterung `.qix`.

Gehen Sie folgendermaßen vor, um den Index zu erstellen:

1. Laden eines Datensatzes im ESRI Shapefile-Format (siehe: *Das Browser-Bedienfeld*)
2. Öffnen Sie den Dialog *Eigenschaften des Layers*, indem Sie auf den Namen des Layers in der Legende doppelklicken oder indem Sie mit der rechten Maustaste klicken und im Kontextmenü *Eigenschaften…* wählen
3. Klicken Sie auf der Registerkarte *Quelle* auf die Schaltfläche *Räumlichen Index erstellen*.

Problem beim Laden einer .prj-Datei

Wenn Sie einen Datensatz im ESRI Shapefile Format mit einer `.prj` Datei laden und QGIS nicht in der Lage ist, das Koordinatenreferenzsystem aus dieser Datei zu lesen, müssen Sie die richtige Projektion manuell im *Layer Eigenschaften* ► *Quelle* Tab des Layers definieren, indem Sie den KBS wählen Button klicken. Dies liegt daran, dass `.prj`-Dateien oft nicht die vollständigen Projektionsparameter enthalten, wie sie in QGIS verwendet werden und im *KBS*-Dialog aufgeführt sind.

Wenn Sie mit QGIS einen neuen Datensatz im ESRI Shapefile-Format erstellen, werden aus demselben Grund zwei verschiedene Projektionsdateien erstellt: eine `.prj`-Datei mit eingeschränkten Projektionsparametern, die mit ESRI-Software kompatibel ist, und eine `.qpj`-Datei, die alle Parameter des KBS enthält. Wann immer QGIS eine `.qpj`-Datei findet, wird diese anstelle der `.prj`-Datei verwendet.

Getrennte Textdateien (CSV)

Getrennte Textdateien sind sehr verbreitet und werden wegen ihrer Einfachheit und Lesbarkeit häufig verwendet - die Daten können in einem einfachen Texteditor angezeigt und bearbeitet werden. Bei einer durch Trennzeichen getrennten Textdatei handelt es sich um tabellarische Daten mit durch ein bestimmtes Zeichen getrennten Spalten und durch Zeilenumbrüche getrennten Zeilen. Die erste Zeile enthält normalerweise die Spaltennamen. Ein gängiger Typ einer durch Trennzeichen getrennten Textdatei ist eine CSV-Datei (Comma Separated Values), bei der die Spalten durch Kommas getrennt sind. Begrenzte Textdateien können auch Positionsangaben enthalten (siehe *Speichern von Geometrieinformationen in getrennten Textdateien*).

QGIS erlaubt es Ihnen, eine getrennte Textdatei als Layer oder eine gewöhnliche Tabelle zu laden (siehe *Das Browser-Bedienfeld* oder *Importieren einer Textdatei (mit Trennzeichen)*). Prüfen Sie zunächst, ob die Datei die folgenden Anforderungen erfüllt:

1. Die Datei muss eine abgegrenzte Kopfzeile mit Feldnamen enthalten. Dies muss die erste Zeile der Daten sein (idealerweise die erste Zeile in der Textdatei).
2. Wenn die Geometrie aktiviert werden soll, muss die Datei Felder enthalten, die die Geometrie definieren. Diese Felder können einen beliebigen Namen haben.
3. Die X- und Y-Koordinatenfelder (wenn die Geometrie durch Koordinaten definiert ist) müssen als Zahlen angegeben werden. Das Koordinatensystem ist nicht wichtig.
4. Wenn Sie eine CSV-Datei mit Nicht-String-Spalten haben, können Sie eine zugehörige CSVT-Datei haben (siehe Abschnitt *Verwendung der CSVT-Datei zur Steuerung der Feldformatierung*).

Die Höhendaten-Datei `elevp.csv` im QGIS-Beispieldatensatz (siehe Abschnitt *Beispieldaten herunterladen*) ist ein Beispiel für eine gültige Textdatei:

```
X;Y;ELEV
-300120;7689960;13
-654360;7562040;52
1640;7512840;3
[...]
```

Zur Textdatei sind einige Dinge zu beachten:

1. Die Beispiel-Textdatei verwendet `;` (Semikolon) als Trennzeichen (jedes beliebige Zeichen kann zur Abgrenzung der Felder verwendet werden).
2. Die erste Zeile ist die Kopfzeile. Sie enthält die Felder `X`, `Y` und `ELEV`.
3. Zur Abgrenzung von Textfeldern werden keine Anführungszeichen (`"`) verwendet.
4. Die X-Koordinaten sind in dem Feld `X` enthalten
5. Die Y-Koordinaten sind in dem Feld `Y` enthalten

Speichern von Geometrieinformationen in getrennten Textdateien

Getrennte Textdateien können Geometrieinformationen in zwei Hauptformen enthalten:

- Als Koordinaten in separaten Spalten (z.B. `Xcol`, `Ycol`...), für punktgeometrische Daten;
- Als Well-known Text (WKT) Darstellung der Geometrie in einer einzigen Spalte, für jeden Geometrietyp.

Objekte mit gekrümmten Geometrien (CircularString, CurvePolygon und CompoundCurve) werden unterstützt. Hier sind einige Beispiele für Geometrietypen in einer begrenzten Textdatei mit als WKT kodierten Geometrien:

```
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))
```

Getrennte Textdateien unterstützen auch Z- und M-Koordinaten in Geometrien:

```
LINESTRINGZ(10.0 20.0 30.0, 11.0 21.0 31.0, 11.0 22.0 30.0)
```

Verwendung der CSVT-Datei zur Steuerung der Feldformatierung

Beim Laden von CSV-Dateien geht der GDAL-Treiber davon aus, dass es sich bei allen Feldern um Strings (d.h. Text) handelt, sofern ihm nichts anderes mitgeteilt wird. Sie können eine CSVT-Datei erstellen, um GDAL (und QGIS) den Datentyp der verschiedenen Spalten mitzuteilen:

Typ	Name	Beispiel
Ganze Zahl	Ganzzahl	4
Boolean	Integer(Boolean)	true
Dezimalzahl	Real	3.456
Datum	Date (YYYY-MM-DD)	2016-07-28
Zeit	Time (HH:MM:SS+nn)	18:33:12+00
Datum & Zeit	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)

Die CSVT-Datei ist eine **einzeilige** reine Textdatei mit den Datentypen in Anführungszeichen und durch Kommas getrennt, z. B.:

```
"Integer","Real","String"
```

Sie können auch die Breite und Genauigkeit jeder Spalte angeben, z. B.:

```
"Integer(6)","Real(5.5)","String(22)"
```

Diese Datei wird im gleichen Ordner wie die `.csv`-Datei gespeichert, mit dem gleichen Namen, aber der Erweiterung `.csvt`.

Weitere Informationen finden Sie unter *GDAL CSV Driver <https://gdal.org/en/latest/drivers/vector/csv.html>*.

Tipp

Feldtypen erkennen

Anstatt eine CSVT-Datei zur Bestimmung der Datentypen zu verwenden, bietet QGIS die Möglichkeit, die Feldtypen automatisch zu erkennen und die angenommenen Feldtypen zu ändern.

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.

Tipp

PostGIS-Layer

Normalerweise wird ein PostGIS Layer durch einen Eintrag in der geometry_columns Tabelle identifiziert. QGIS kann auch Layer laden, die keinen Eintrag in der geometry_columns Tabelle haben. Dies schließt sowohl Tabellen als auch Views ein. Informationen zur Erstellung von Views finden Sie in Ihrem PostgreSQL-Handbuch.

Dieser Abschnitt enthält einige Details darüber, wie QGIS auf PostgreSQL Layer zugreift. In den meisten Fällen sollte QGIS Ihnen einfach eine Liste von Datenbanktabellen zur Verfügung stellen, die geladen werden können, und es wird sie auf Anfrage laden. Wenn Sie jedoch Probleme haben, eine PostgreSQL-Tabelle in QGIS zu laden, können die folgenden Informationen Ihnen helfen, die QGIS-Meldungen zu verstehen und Ihnen Hinweise geben, wie Sie die PostgreSQL-Tabellen- oder View-Definition modifizieren können, damit QGIS sie laden kann.

Bemerkung

Eine PostgreSQL-Datenbank kann auch QGIS-Projekte speichern.

Primärschlüssel

QGIS verlangt, dass PostgreSQL Layer eine Spalte enthalten, die als eindeutiger Schlüssel für den Layer verwendet werden kann. Für Tabellen bedeutet dies normalerweise, dass die Tabelle einen Primärschlüssel oder eine Spalte mit einer eindeutigen Einschränkung benötigt. In QGIS muss diese Spalte vom Typ int4 sein (eine Ganzzahl der Größe 4 Bytes). Alternativ kann auch die Spalte ctid als Primärschlüssel verwendet werden. Wenn eine Tabelle nicht über diese Elemente verfügt, wird stattdessen die oid-Spalte verwendet. Die Leistung wird verbessert, wenn die Spalte indiziert ist (beachten Sie, dass Primärschlüssel in PostgreSQL automatisch indiziert werden).

QGIS bietet eine Checkbox **Auswahl nach ID**, die standardmäßig aktiviert ist. Diese Option liefert die IDs ohne die Attribute, was in den meisten Fällen schneller ist.

Sicht (View)

Wenn der PostgreSQL Layer eine Datenbank-Sicht (View) ist, besteht die gleiche Anforderung, aber Views haben nicht immer Primärschlüssel oder Spalten mit eindeutigen Beschränkungen für sie. Sie müssen ein Primärschlüsselfeld (muss eine ganze Zahl sein) im QGIS Dialog definieren, bevor Sie den View laden können. Wenn keine passende Spalte in der Ansicht vorhanden ist, wird QGIS den Layer nicht laden. In diesem Fall müssen Sie die Ansicht so ändern, dass sie eine geeignete Spalte enthält (einen Integer-Typ und entweder einen Primärschlüssel oder eine eindeutige Einschränkung, vorzugsweise indiziert).

Für die Tabelle ist standardmäßig ein Kontrollkästchen **Auswahl nach ID** aktiviert (zur Bedeutung des Kontrollkästchens siehe oben). Es kann sinnvoll sein, diese Option zu deaktivieren, wenn Sie aufwendige Views verwenden.

Bemerkung

PostgreSQL Foreign Table

PostgreSQL-Foreign-Tables werden vom PostgreSQL-Anbieter nicht explizit unterstützt und werden wie ein View behandelt.

QGIS layer_style Tabelle und Datenbank Backup

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. Erstellen Sie ein PLAIN-Backup der Tabelle `layer_style`.
2. Öffnen Sie die Datei in einem Texteditor
3. Ändern Sie die Zeile `SET xmloption = content;` in `SET XML OPTION DOCUMENT;`
4. Speichern Sie die Datei
5. Verwenden Sie psql, um die Tabelle in der neuen Datenbank wiederherzustellen

Filtern der Datenbankseite

QGIS erlaubt es, Objekte bereits serverseitig zu filtern. Aktivieren Sie dazu *Einstellungen* ► *Optionen* ► *Datenquellen* ► *Ausdrücke serverseitig ausführen wenn möglich*. Es werden nur unterstützte Ausdrücke an die Datenbank gesendet. Ausdrücke, die nicht unterstützte Operatoren oder Funktionen verwenden, werden automatisch auf die lokale Auswertung zurückgesetzt.

Unterstützung von PostgreSQL-Datentypen

Folgende Datentypen werden vom PostgreSQL-Anbieter unterstützt: Integer, Float, Boolean, Binärobjekt, Varchar, Geometrie, Zeitstempel, Array, hstore und json.

Daten in PostgreSQL importieren

Data can be imported into PostgreSQL using several tools, including the Browser, DB Manager plugin and the command line tools shp2pgsql and ogr2ogr.

DB Manager

QGIS wird mit einem Kern-Plugin namens DB-Verwaltung geliefert. Es kann zum Laden von Daten verwendet werden und bietet Unterstützung für Schemata. Siehe Abschnitt *DB Verwaltung* für weitere Informationen.

shp2pgsql

PostGIS enthält ein Dienstprogramm namens **shp2pgsql**, das verwendet werden kann, um Datensätze im Shapefile-Format in eine PostGIS-aktivierte Datenbank zu importieren. Um zum Beispiel einen Datensatz im Shapefile-Format mit dem Namen `lakes.shp` in eine PostgreSQL-Datenbank mit dem Namen `gis_data` zu importieren, verwenden Sie den folgenden Befehl:

```
shp2pgsql -s 2964 lakes.shp lakes_new | psql gis_data
```

Dies erzeugt einen neuen Layer mit dem Namen `lakes_new` in der Datenbank `gis_data`. Der neue Layer hat einen spatial reference identifier (SRID) von 2964. Siehe Abschnitt *Arbeiten mit Projektionen* für weitere Informationen über räumliche Bezugssysteme und Projektionen.

> **Tipp**
>
> **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
```

Wenn Sie es vorziehen, den **COPY**-Befehl von PostgreSQL anstelle der Standardmethode **INSERT INTO** zu verwenden, können Sie die folgende Umgebungsvariable exportieren (zumindest unter und verfügbar):

```
export PG_USE_COPY=YES
```

ogr2ogr erstellt keine räumlichen Indizes wie **shp2pgsl**. Sie müssen sie manuell erstellen, indem Sie anschließend den normalen SQL-Befehl **CREATE INDEX** als zusätzlichen Schritt verwenden (wie im nächsten Abschnitt *Verbesserung der Leistung* beschrieben).

Verbesserung der Leistung

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).

> **Tipp**
>
> Sie können die DB-Veraltung verwenden, um einen Index für Ihren Layer zu erstellen. Wählen Sie zunächst den Layer aus und klicken Sie auf *Tabelle* ► *Tabelle bearbeiten*, gehen Sie auf die Registerkarte *Indizes* und klicken Sie auf *Räumlichen Index hinzufügen*.

Die Syntax für die Erstellung eines GiST-Index lautet:

```
CREATE INDEX [indexname] ON [tablename]
  USING GIST ( [geometryfield] GIST_GEOMETRY_OPS );
```

Beachten Sie, dass bei großen Tabellen die Erstellung des Indexes sehr lange dauern kann. Sobald der Index erstellt ist, sollten Sie eine `VACUUM ANALYZE` durchführen. Siehe die PostGIS-Dokumentation (POSTGIS-PROJECT in *Literatur und Internetreferenzen*) für weitere Informationen.

Das folgende Beispiel erstellt einen GiST-Index:

```
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

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$
```

SpatiaLite-Layer

Wenn Sie einen Layer im SpatiaLite-Format speichern wollen, können Sie dies mit Hilfe der Anweisungen unter *Erstellen neuer Layer aus einem vorhandenen Layer* tun. Sie wählen `SpatiaLite` als *Format* und geben sowohl *Dateiname* als auch *Layername* ein.

Außerdem können Sie als Format `SQLite` auswählen und dann `SPATIALITE=YES` im Feld *Benutzerdefinierte Optionen* ► *Datenquelle* hinzufügen. Dadurch wird GDAL angewiesen, eine SpatiaLite-Datenbank zu erstellen. Siehe auch https://gdal.org/en/latest/drivers/vector/sqlite.html.

QGIS unterstützt auch editierbare Ansichten in SpatiaLite. Für die Verwaltung von SpatiaLite-Daten können Sie auch das Kern-Plugin *DB-Verwaltung* verwenden.

Wenn Sie einen neuen SpatiaLite Layer erstellen wollen, lesen Sie bitte den Abschnitt *Erstellen eines neuen SpatiaLite Layers*.

GeoJSON-spezifische Parameter

Beim *Export von Layern* nach GeoJSON stehen einige spezielle *Layeroptionen* zur Verfügung. Diese Optionen kommen von GDAL, das für das Schreiben der Datei verantwortlich ist:

- *COORDINATE_PRECISION* die maximale Anzahl von Ziffern nach dem Dezimaltrennzeichen, die in Koordinaten geschrieben werden. Der Standardwert ist 15 (Hinweis: für Lat-Lon-Koordinaten reichen 6). Nachfolgende Nullen werden abgeschnitten.
- *RFC7946* Standardmäßig wird GeoJSON 2008 verwendet. Bei Einstellung auf YES wird der aktualisierte Standard RFC 7946 verwendet. Die Standardeinstellung ist NO (also GeoJSON 2008). Die wichtigsten Unterschiede finden Sie unter https://gdal.org/en/latest/drivers/vector/geojson.html#rfc-7946-write-support. Kurz gesagt: Nur EPSG:4326 ist zulässig, andere KBS werden transformiert, Polygone werden so geschrieben, dass sie der Rechtsregel für die Ausrichtung folgen, die Werte eines „bbox”-Arrays sind [west, south, east, north] und nicht [minx, miny, maxx, maxy]. Einige Erweiterungsmitgliedernamen sind in FeatureCollection-, Feature- und Geometry-Objekten verboten, die Standardkoordinatengenauigkeit beträgt 7 Dezimalstellen.
- *WRITE_BBOX* auf YES gesetzt, um den Begrenzungsrahmen der Geometrien auf der Ebene der Objekte und der Objektsammlung einzuschließen

Neben GeoJSON gibt es auch eine Option zum Exportieren nach „GeoJSON – Newline Delimited” (siehe https://gdal.org/en/latest/drivers/vector/geojsonseq.html). Anstelle einer FeatureCollection mit Features können Sie einen Typ (wahrscheinlich nur Features) sequenziell durch Zeilenumbrüche getrennt streamen.

GeoJSON - Newline Delimited verfügt auch über einige spezifische Layer-Optionen:

- *COORDINATE_PRECISION* siehe oben (gleich wie bei GeoJSON)

- *RS*, ob Datensätze mit dem Zeichen RS=0x1E beginnen sollen. Der Unterschied besteht darin, wie die Objekte getrennt werden: nur durch ein Zeilenumbruchzeichen (LF) (Newline Delimited JSON, geojsonl) oder durch Voranstellen eines Datensatz-Trennzeichens (RS) (bei GeoJSON-Text-Sequenzen, geojsons). Die Voreinstellung ist NO. Die Dateien erhalten die Erweiterung `.json`, wenn keine Erweiterung angegeben wird.

SAP HANA Spatial Layers

Dieser Abschnitt enthält einige Details darüber, wie QGIS auf SAP HANA Layer zugreift. In den meisten Fällen sollte QGIS Ihnen einfach eine Liste von Datenbanktabellen und -ansichten zur Verfügung stellen, die geladen werden können, und es wird sie auf Anfrage laden. Wenn Sie jedoch Probleme haben, eine SAP HANA-Tabelle oder -Ansicht in QGIS zu laden, können die folgenden Informationen Ihnen helfen, die Ursache zu verstehen und das Problem zu lösen.

Identifizierung von Objekten

Wenn Sie alle Möglichkeiten von QGIS zur Bearbeitung von Objekten nutzen möchten, muss QGIS in der Lage sein, jedes Objekt in einem Layer eindeutig zu identifizieren. Intern verwendet QGIS eine 64-Bit-Ganzzahl mit Vorzeichen, um Objekte zu identifizieren, während der negative Bereich für spezielle Zwecke reserviert ist.

Daher benötigt der SAP HANA-Anbieter einen eindeutigen Schlüssel, der auf eine positive 64-Bit-Ganzzahl abgebildet werden kann, um die Bearbeitungsfunktionen von QGIS für Objekte vollständig zu unterstützen. Wenn es nicht möglich ist, ein solches Mapping zu erstellen, können Sie die Objekte zwar noch anzeigen, aber möglicherweise nicht bearbeiten.

Hinzufügen von Tabellen

Wenn Sie eine Tabelle als Layer hinzufügen, verwendet der SAP HANA-Anbieter den Primärschlüssel der Tabelle, um sie einer eindeutigen Objekt-ID zuzuordnen. Um eine vollständige Unterstützung für die Bearbeitung von Objekten zu erhalten, müssen Sie daher einen Primärschlüssel für Ihre Tabellendefinition haben.

Der SAP HANA-Anbieter unterstützt mehrspaltige Primärschlüssel, aber wenn Sie die beste Leistung erzielen möchten, sollte Ihr Primärschlüssel eine einzelne Spalte vom Typ `INTEGER` sein.

Hinzufügen von Views

Wenn eine Ansicht als Layer hinzugefügt wird, kann der SAP HANA-Anbieter nicht automatisch Spalten identifizieren, die ein Objekt eindeutig kennzeichnen. Außerdem sind einige Ansichten schreibgeschützt und können nicht bearbeitet werden.

Um die volle Unterstützung für die Bearbeitung von Objekten zu haben, muss die Ansicht aktualisierbar sein (überprüfen Sie die Spalte `IS_READ_ONLY` in der Systemansicht `SYS.VIEWS` für die betreffende Ansicht) und Sie müssen QGIS manuell eine oder mehrere Spalten zur Verfügung stellen, die ein Objekt identifizieren. Die Spalten können mit Hilfe von *Layer* ► *Layer hinzufügen* ► *SAP HANA Spatial Layer hinzufügen* angegeben werden und dann die Spalten in der Spalte *Feature id* ausgewählt werden. Um eine optimale Leistung zu erzielen, sollte der *Feature id*-Wert eine einzelne `INTEGER`-Spalte sein.

11.4.3 Layer, die den 180° Längengrad kreuzen

In vielen GIS-Paketen werden Layer mit einem geografischen Bezugssystem (Breitengrad/Längengrad), das die 180-Grad-Linie überschreitet, nicht erfasst. Wenn man einen solchen Layer in QGIS öffnet, kann man zwei weit voneinander entfernte Orte sehen, die eigentlich nahe beieinander liegen sollten. In Abb. 11.42 sollte der winzige Punkt ganz links auf der Kartenansicht (Chatham-Inseln) innerhalb des Gitters liegen, rechts von den neuseeländischen Hauptinseln.

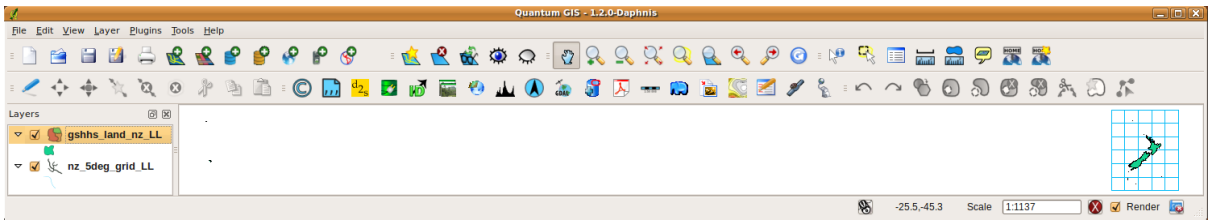


Abb. 11.42: Karte in Breiten- und Längengraden, die die 180°-Linie kreuzen

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.

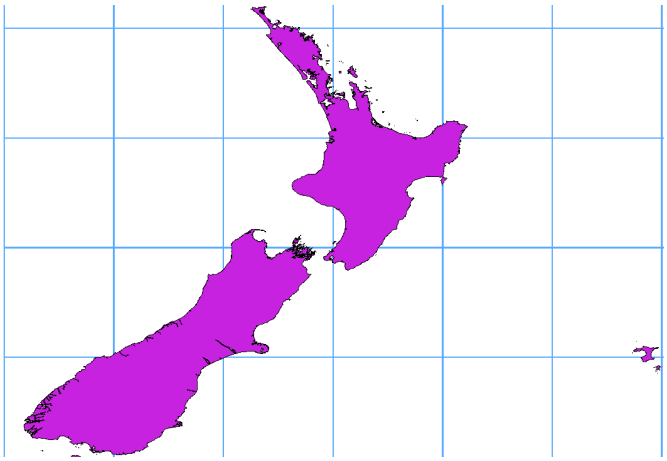


Abb. 11.43: Überschreiten des 180°-Längengrads unter Anwendung der Funktion **ST_ShiftLongitude**.

1. Import data into PostgreSQL (*Daten in PostgreSQL importieren*) 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. Wenn alles gut gegangen ist, sollten Sie eine Bestätigung über die Anzahl der Objekte erhalten, die aktualisiert wurden. Dann können Sie die Karte laden und den Unterschied sehen (*Figure_vector_crossing_map*).

KAPITEL 12

Arbeiten mit Vektordaten

12.1 Der Eigenschaften-Dialog von Vektorlayern

Der *Layereigenschaften* Dialog für Vektorlayer enthält generelle Einstellungen, um das Aussehen der Layerobjekte in der Karte (Symbologie, Beschriftung, Diagramme) und die Interaktion mit der Maus (Aktionen, Kartentipps, …) zu verwalten. Er bietet auch Informationen über den Layer.

Sie können den *Layereigenschaften* Dialog öffnen, indem Sie:

- im *Layer `Bedienfeld einen Doppelklick auf den Layer machen oder einen Rechtsklick und im Kontextmenü :guilabel:`Eigenschaften …* auswählen
- in der Menüleiste *Layer* ► *Layereigenschaften …* auswählen wenn der Layer aktiv ist

Der *Layereigenschaften* Dialog für Vektorlayer bietet die folgenden Bereiche:

Information	*Quelle*	*Symbology* ([1])
Labels ([1])	*Mask* ([1])	*3D View* ([1])
Diagramme	*Felder*	*Attributformular*
Verknüpfungen	*Hilfsspeicher*	*Aktionen*
Anzeige	*Darstellung*	*Zeitlich*
Variablen	*Höhe*	*Metadaten*
Abhängigkeiten	*Legende*	*QGIS Server*
Digitalisieren	*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.

Tipp

Layereigenschaften speichern und teilen

Mit dem *Stil* Menü unten im Dialogfenster ist es möglich die Layereigenschaften (oder Teile davon) zu speichern und zu laden . Für mehr Informationen s. *Verwalten von benutzerdefinierten Stilen*.

Bemerkung

Da die Eigenschaften von eingebetteten Layern (see *Layer aus externen Projekten einbinden*) aus der ursprünglichen QGIS-Projektdatei stammen und um Änderungen zu vermeiden, die mit diesen Einstellungen Konflikte verursachen könnten, ist der Ebeneneigenschaften-Dialog für eingebetteten Ebenen nicht verfügbar.

12.1.1 Information

Die Registerkarte *Information* ist nur zum Lesen vorgesehen und stellt einen interessanten Ort dar, um schnell zusammengefasste Informationen und Metadaten über den aktuellen Layer zu erhalten. Die bereitgestellten Informationen sind:

- allgemeine Angaben wie Name im Projekt, Quellpfad, Liste der Hilfsdateien, letzte Speicherzeit und -größe, der verwendete Datenanbieter
- Benutzerdefinierte Eigenschaften, die verwendet werden, um zusätzliche Informationen über den Layer im aktiven Projekt zu speichern. Zu den standardmäßigen benutzerdefinierten Eigenschaften können gehören: *Layer-Notizen*, *Legenden-Widgets*, *Layer-Variablen*, *Formulareigenschaften*… Weitere benutzerdefinierte Eigenschaften können mit PyQGIS erstellt und verwaltet werden, insbesondere mit der Methode setCustomProperty().
- abhänigig vom Daten-Anbieter des Layers: Speicherformat, Geometrietyp, Kodierung der Datenquelle, Ausdehnung, Anzahl der Objekte…
- das Koordinatenbezugssystem: Name, Einheiten, Methode, Genauigkeit, Referenz (d.h., ob es statisch oder dynamisch ist)
- aus den *ausgefüllten Metadaten*: Zugriff, Ausdehnung, Links, Kontakte, Protokoll…
- und mit seiner Geometrie (räumliche Ausdehnung, KBS…) oder seinen Attributen (Anzahl der Felder, Typ jedes Feldes…) zusammenhängen.

12.1.2 Quelle

Auf dieser Registerkarte können Sie allgemeine Einstellungen für die Vektorebene festlegen.

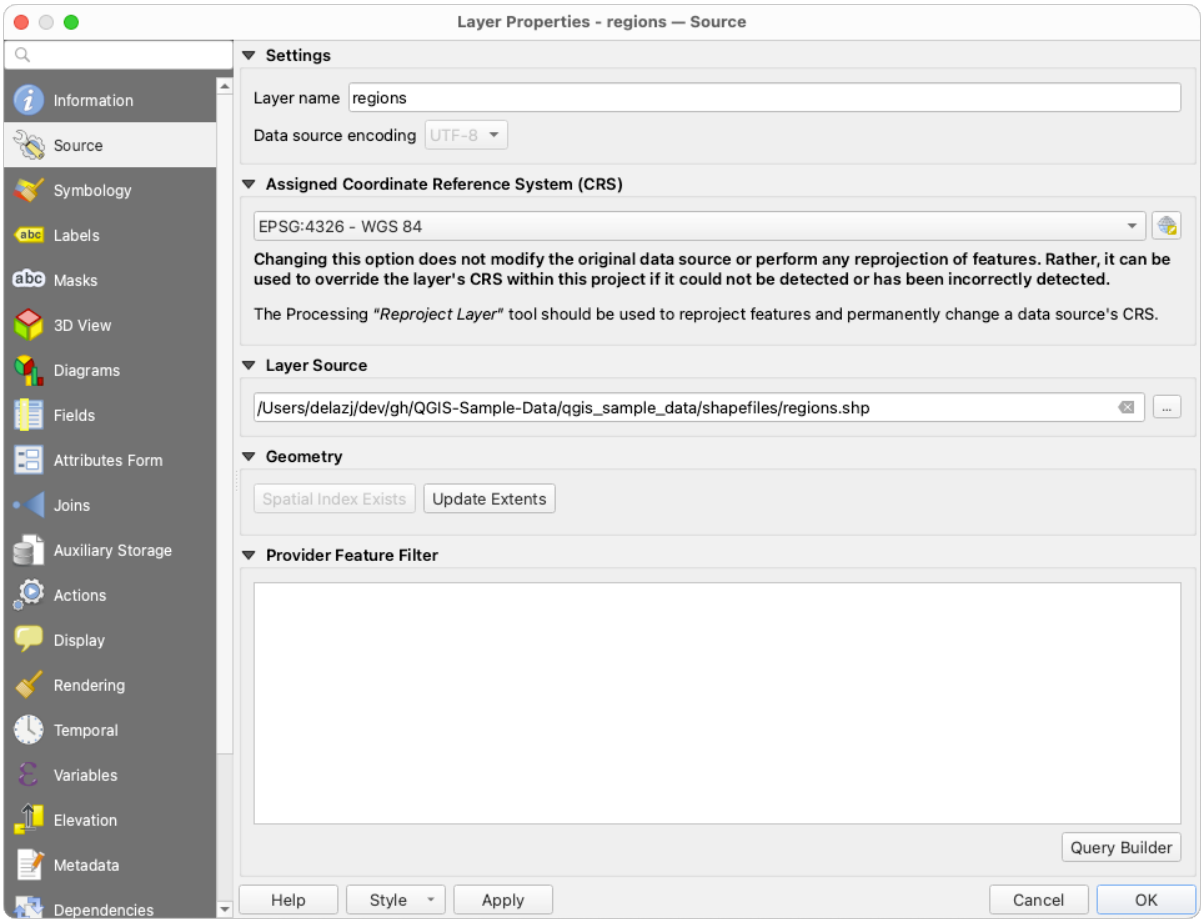


Abb. 12.1: Registerkarte Quelle im Menu Vektorlayereigenschaften

Einstellungen

- Lege einen *layernamen* fest, der sich vom Dateinamen der Ebene unterscheidet und zur Identifizierung der Ebene im Projekt verwendet wird (im *Layerbedienfeld*, mit Ausdrücken, in der Drucklayoutlegende, …)
- Wählen Sie je nach Datenformat die *Datenquellenkodierung*, wenn sie von QGIS nicht korrekt erkannt wird.

Coordinate Reference System

- Zeigt das *Zugeordnete Koordinatenbezugssystem (KBS)* des Layers an. Sie können das KBS des Layers ändern, indem Sie ein kürzlich verwendetes KBS aus der Dropdown-Liste auswählen oder auf die Schaltfläche KBS auswählen klicken (siehe *Auswahl des Koordinatenbezugssystems*). Verwenden Sie dieses Verfahren nur, wenn das auf den Layer angewendete KBS ein falsches KBS ist oder wenn kein KBS angewendet wurde. Wenn Sie Ihre Daten in ein anderes KBS projizieren möchten, verwenden Sie stattdessen die Algorithmen zur Layer-Reprojektion von der Verarbeitung oder *Speichern Sie die Daten in einem anderen 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.
- *Ausdehnung aktualisieren* für den Layer.

Abfrageerstellung

Der Abfrage-Dialog ist über die Schaltfläche *Abfrageerstellung* unten auf der Registerkarte *Quelle* im Dialogfeld Layer-Eigenschaften im Bereich *Objektfilter auf Datenanbieter* zu erreichen.

Die Abfrageerstellung bietet eine Schnittstelle, mit der Sie eine Teilmenge der Objekte im Layer mit einer SQL-ähnlichen WHERE-Klausel definieren und das Ergebnis im Hauptfenster anzeigen können. Solange die Abfrage aktiv ist, sind nur die Objekte, die ihrem Ergebnis entsprechen, im Projekt verfügbar.

Sie können ein oder mehrere Layer-Attribute verwenden, um den Filter in der Abfrageerstellung zu definieren. Die Verwendung von mehr als einem Attribut wird in Abb. 12.2 gezeigt. In diesem Beispiel kombiniert der Filter die Attribute

- `toa` (`DateTime` Feld: `cast("toa" as character) > '2017-05-17'` und `cast("toa" as character) < '2019-12-24T18:00:00'`),
- `name` (`String` Feld: `"name" > 'S'`) und
- `FID` (`Integer` Feld: `FID > 10`)

unter Verwendung der Operatoren AND, OR und NOT und Klammern. Diese Syntax (einschließlich des DateTime-Formats für das Feld `toa`) funktioniert für GeoPackage-Datensätze.

Der Filter wird auf der Ebene des Datenanbieters (OGR, PostgreSQL, MS SQL Server…) erstellt. Die Syntax hängt also vom Datenanbieter ab (DateTime wird z. B. nicht für das ESRI Shapefile-Format unterstützt). Der vollständige Ausdruck:

```
cast("toa" as character) > '2017-05-17' AND
cast("toa" as character) < '2019-12-24T18:00:00' AND
NOT ("name" > 'S' OR FID > 10)
```

Sie können den Dialog *Abfrageerstellung* auch mit der Option *Filter…* aus dem Menü *Layer* oder dem Layer-Kontextmenü öffnen. Die Abschnitte *Felder*, *Werte* und *Operatoren* im Dialog helfen Ihnen, die SQL-ähnliche Abfrage zu konstruieren, die im Feld *Datenanbieterspezifischer Filterausdruck* angezeigt wird.

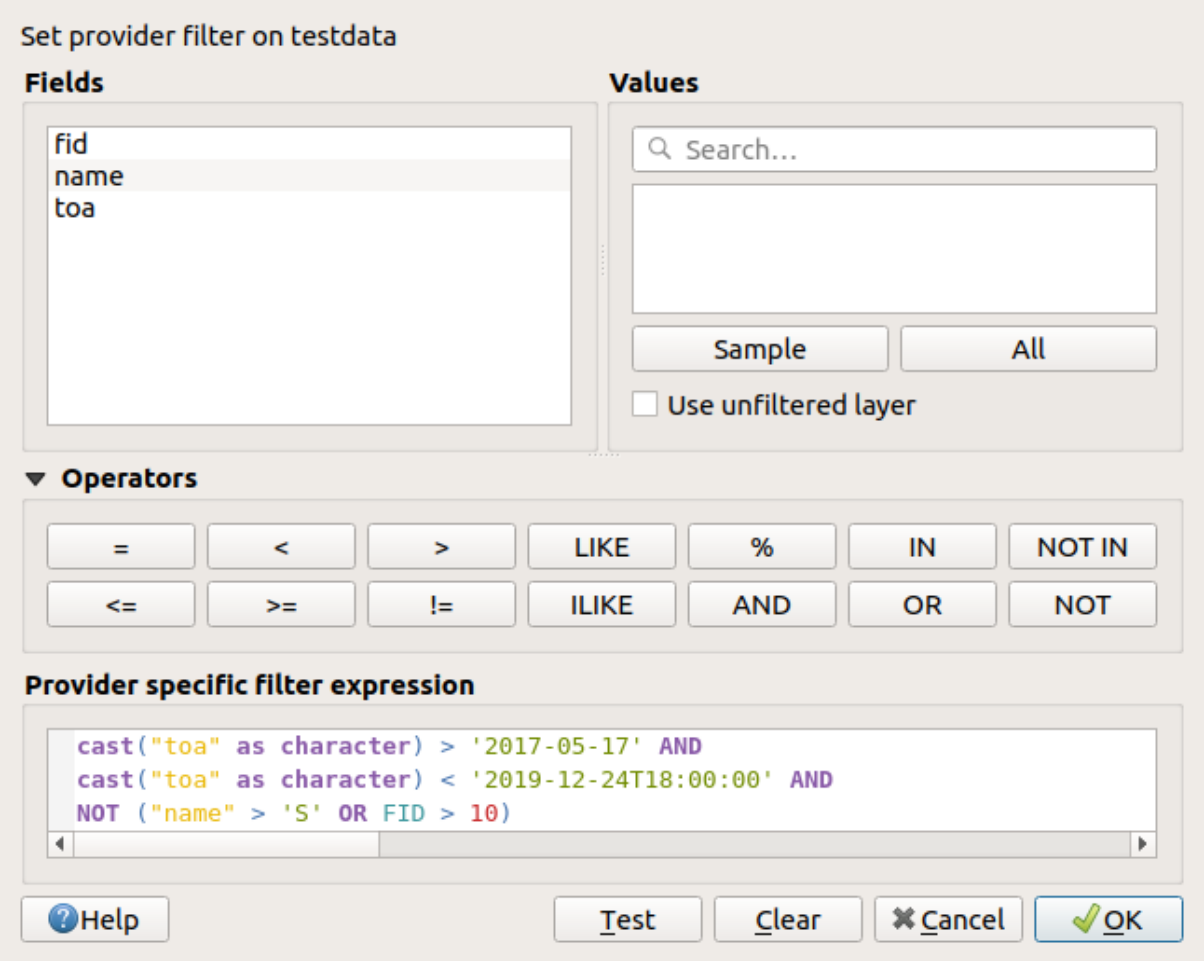


Abb. 12.2: Abfrageerstellung

Die Liste **Felder** enthält alle Felder des Layers. Um einen Feldnamen hinzuzufügen, doppelklicken Sie einfach auf den Namen oder geben Sie ihn in das Ausdrucks-Feld ein.

Der Bereich **Werte** listet die Werte des aktuell ausgewählten Feldes auf. Um alle eindeutigen Werte eines Feldes aufzulisten, klicken Sie auf die Schaltfläche *Alle*. Um stattdessen die ersten 25 eindeutigen Werte der Spalte aufzulisten, klicken Sie auf die Schaltfläche *Stichprobe*. Um dem Feld SQL WHERE-Klausel einen Wert hinzuzufügen, doppelklicken Sie auf seinen Namen in der Liste Werte. Mit dem Suchfeld am oberen Rand des Rahmens „Werte“ können Sie die Attributwerte in der Liste leicht durchsuchen und finden.

Der Abschnitt **Operatoren** enthält alle verwendbaren Operatoren. Um einen Operator in das SQL WHERE-Klauselfeld einzufügen, klicken Sie auf die entsprechende Schaltfläche. Es stehen relationale Operatoren (=, >, …), String-Vergleichsoperatoren (`LIKE`) und logische Operatoren (`AND`, `OR`, …) zur Verfügung.

Die Schaltfläche *Test* hilft Ihnen bei der Überprüfung Ihrer Abfrage und zeigt in einem Meldungsfenster die Anzahl der Objekte an, die der aktuellen Abfrage entsprechen. Verwenden Sie die Schaltfläche *Löschen*, um die SQL-Abfrage zu löschen und den Layer in seinen ursprünglichen Zustand zurückzuversetzen (d.h. alle Objekte vollständig zu laden). Es ist möglich, die Abfrage als *.QQF* Datei zu speichern oder sie mit *Laden…* aus einer solchen Datei in den Dialog zu kopieren.

Wenn ein Filter angewendet wird, behandelt QGIS die resultierende Teilmenge so, als wäre sie der gesamte Layer. Wenn Sie z.B. einen Filter auf ein Feld für Verwaltungseinheiten angewendet haben (z.B. „`TYPE_2`“ = `'Stadtbezirk'`), können Sie keine Objekte anzeigen, abfragen, speichern oder bearbeiten, die in diesem Feld über ein anderes Attriubt verfügen, also z.B. wenn in dem Feld „`TYPE_2`“ das Attribut ``‘Gemeinde‘``eingetragen ist.

Tipp

Gefilterte Layer werden im „ Layer“ Bedienfeld gekennzeichnet

Im Bedienfeld *Layer* wird ein gefilterter Layer mit einem Symbol Filter daneben angezeigt, welches die verwendete Abfrage anzeigt, wenn der Mauszeiger über der Schaltfläche schwebt. Ein Doppelklick auf das Symbol öffnet den *Abfrageerstellung*-Dialog zur Bearbeitung. Dies kann auch über das Menü *Layer* ► *Filter…* erreicht werden.

12.1.3 Symbolisierung

Die Registerkarte *Symbolisierung* bietet Ihnen ein umfassendes Werkzeug zum Rendern und Symbolisieren Ihrer Vektordaten. Sie können sowohl Werkzeuge verwenden, die für alle Vektordaten gelten, als auch spezielle Symbolisierungswerkzeuge, die für die verschiedenen Arten von Vektordaten entwickelt wurden. Alle Typen haben jedoch die folgende Dialogstruktur gemeinsam: Im oberen Teil befindet sich ein Widget, das Ihnen hilft, die Klassifizierung und das Symbol für Objekte vorzubereiten, und im unteren Teil das Widget *Layerdarstellung*.

Tipp

Wechseln Sie schnell zwischen verschiedenen Layerdarstellungen

Über das Menü *Stile* ► *Hinzufügen* am unteren Rand des Dialogs *Layer-Eigenschaften* können Sie so viele Stile wie nötig speichern. Ein Stil ist die Kombination aller Eigenschaften eines Layers (z.B. Symbologie, Beschriftung, Diagramm, Formularfelder, Aktionen…), wie Sie sie wünschen. Wechseln Sie dann einfach über das Kontextmenü des Layers im *Layers Panel* zwischen den Stilen, um automatisch verschiedene Darstellungen Ihrer Daten zu erhalten.

Tipp

Symbologie exportierten

Sie haben die Option die Vektorsymbologie von QGIS nach Google *.kml, *.dxf und MapInfo .*.tab Dateien zu exportieren. Öffnen Sie einfach das Rechte-Maustasten-Menü des Layers und klicken Sie auf *Speichern als …* ` *um den Namen der Ausgabedatei und ihr Format festzulegen. Verwenden Sie im Dialog das :guilabel:`Darstellungsexport* Menü um die Symbologie entweder als *Objektdarstellung* ► oder *Symbollayerdarstellung* ► zu speichern. Wenn Sie Symbollayer verwendet haben wird empfohlen die zweite Einstellung zu benutzen.

Objekt Darstellung

Der Renderer ist dafür verantwortlich, ein Objekt zusammen mit dem richtigen Symbol zu zeichnen. Unabhängig vom Typ der Layer-Geometrie gibt es vier gängige Renderer-Typen: Einzelsymbol, kategorisiert, abgestuft und regelbasiert. Für Punkt-Layer stehen die Renderer Punktversatz, Punkthäufung und Heatmap zur Verfügung, während Polygon-Layer auch mit den Renderern Objekte verschmolzenen, invertierte Polygone und 2,5 D gerendert werden können.

Es gibt keinen kontinuirliche Farbe Renderer da es in der Tat einfach ein spezieller Fall des Abgestuft Renderers ist. Die Kategorisiert und Abgestuft Renderer können erstellt werden indem ein Symbol und ein Farbverlauf festgelegt werden - Sie werden die Farben für Symbole angemessen einsetzen. Für jeden Datentyp (Punkte, Linien und Polygone) sind Symbollayertypen erhältlich. Abhängig vom ausgesuchten Renderer gibt es verschiedene zusätzliche Bereiche.

Bemerkung

Wenn Sie den Darstellungstyp beim Einstellen des Stils eines Vektorlayers ändern werden die Einstellungen für das Symbol beibehalten. Beachten Sie dass dieses Vorgehen nur für eine Änderunge funktioniert. Wenn Sie den Darstellungstyp wiederholt ändern gehen die Einstellungen für das Symbol verloren.

Einzelsymbol Darstellung

Der Renderer *Einzelsymbol* wird verwendet, um alle Objekte des Layers mit einem einzigen benutzerdefinierten Symbol darzustellen. Siehe *Die Symbolauswahl* für weitere Informationen zur Symboldarstellung.

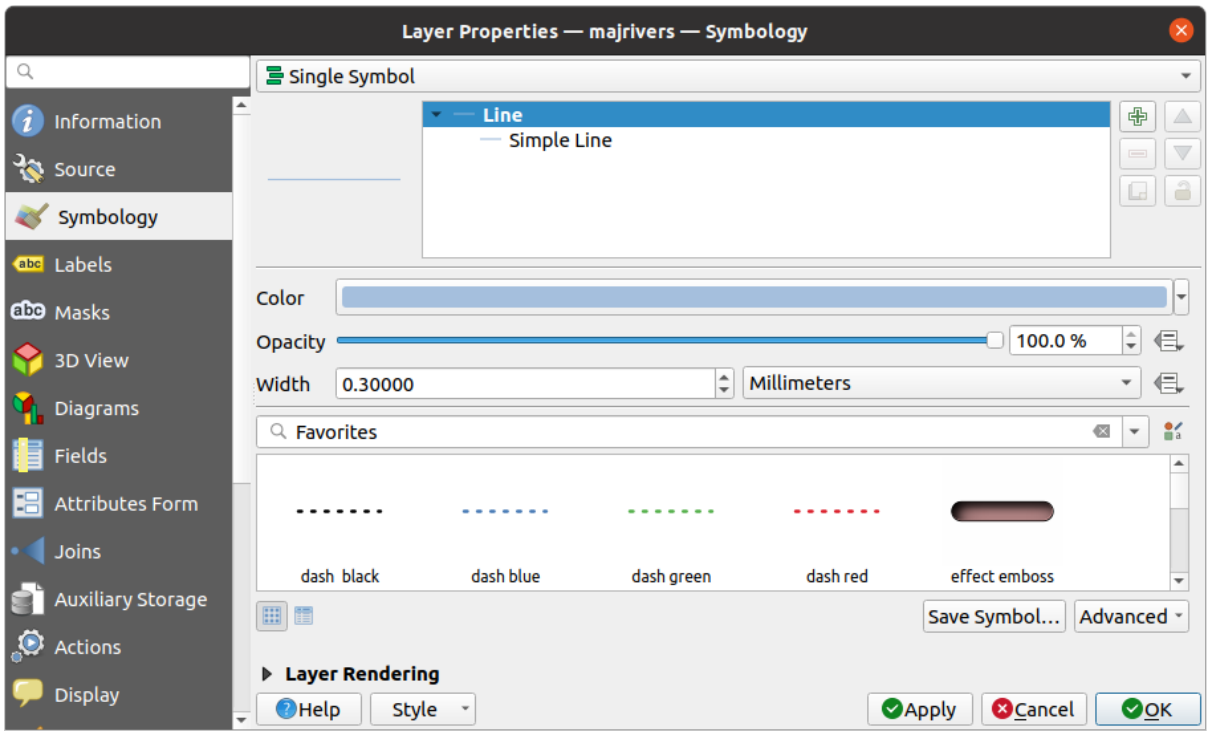


Abb. 12.3: Linieneigenschaften Einzelsymbol

Keine Symbole

Der Renderer *Keine Symbole* ist ein spezieller Anwendungsfall des Renderers Einzelsymbol, da er auf alle Objekte das gleiche Rendering anwendet. Mit diesem Renderer wird für Objekte kein Symbol gezeichnet, aber Beschriftungen, Diagramme und andere nicht-symbolische Teile werden weiterhin angezeigt.

Die Layer in der Kartenansicht können nach wie vor ausgewählt werden, und die ausgewählten Objekte werden mit einem Standardsymbol gerendert. Objekte, die gerade bearbeitet werden, werden ebenfalls angezeigt.

Dies ist eine praktische Möglichkeit für Layer, für die Sie nur Beschriftungen oder Diagramme anzeigen möchten, und vermeidet die Anforderung, Symbole mit vollständig transparenten Füllungen/Rändern zu rendern, um dies zu erreichen.

Kategorisierte Darstellung

Der Renderer *Kategorisiert* dient zur Darstellung der Objekte eines Layers unter Verwendung eines benutzerdefinierten Symbols, dessen Aspekt die diskreten Werte eines Feldes oder eines Ausdrucks widerspiegelt.

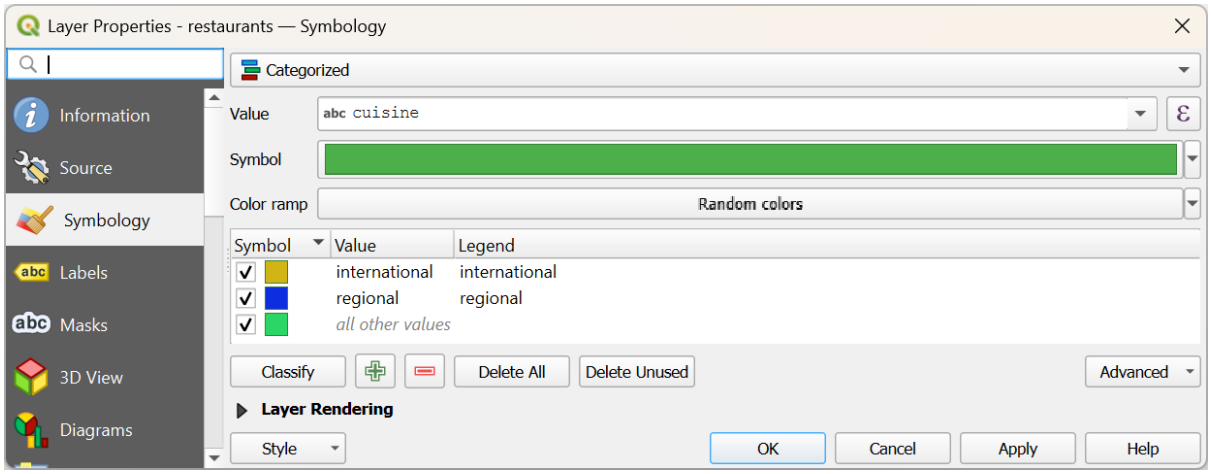


Abb. 12.4: Kategorisierte Symbolisierungsoptionen

Um eine kategorisierte Symbologie für einen Layer zu verwenden:

1. Wählen Sie den *Wert* der Klassifizierung: Es kann ein bestehendes Feld oder ein Ausdruck sein, den Sie in das Feld eingeben oder über die zugehörige Schaltfläche erstellen können. Die Verwendung von Ausdrücken für die Kategorisierung vermeidet die Notwendigkeit, ein Feld nur für Symbolzwecke zu erstellen (z. B. wenn Ihre Klassifizierungskriterien von einem oder mehreren Attributen abgeleitet sind).

   Der zur Klassifizierung von Objekten verwendete Ausdruck kann von beliebiger Art sein, z. B. kann er:

   - ein Vergleich sein. In diesem Fall liefert QGIS die Werte `1` (**True**) und `0` (**False**). Einige Beispiele:

     ```
     myfield >= 100
     $id = @atlas_featureid
     myfield % 2 = 0
     within( $geometry, @atlas_geometry )
     ```

   - verschiedene Bereiche kombinieren:

     ```
     concat( field_1, ' ', field_2 )
     ```

   - eine Berechnung auf Feldern sein:

     ```
     myfield % 2
     year( myfield )
     field_1 + field_2
     substr( field_1, -3 )
     ```

   - verwendet werden, um lineare Werte in diskrete Klassen umzuwandeln, z. B.:

     ```
     CASE WHEN x > 1000 THEN 'Big' ELSE 'Small' END
     ```

   - mehrere diskrete Werte zu einer einzigen Kategorie zusammenfassen, z. B.:

     ```
     CASE
     WHEN building IN ('residence', 'mobile home') THEN 'residential'
     WHEN building IN ('commercial', 'industrial') THEN 'Commercial and␣
     ```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
↪Industrial'
END
```

Tipp

Sie können zwar jede Art von Ausdruck verwenden, um Objekte zu kategorisieren, aber für einige komplexe Ausdrücke kann es einfacher sein, *Regelbasiertes Rendering* zu verwenden.

2. Konfigurieren Sie das *Symbol*, das als Basissymbol für alle Klassen verwendet werden soll;
3. Geben Sie den *Farbverlauf* an, d. h. den Farbbereich, aus dem die Farbe ausgewählt wird, die auf jedes Symbol angewendet wird.

   Neben den üblichen Optionen des *Farbverlauf-Werkzeug* können Sie die Kategorien mit einem *Zufälligen Farbverlauf* versehen. Wenn Sie mit der Auswahl der Farben nicht zufrieden sind, können Sie auf den Eintrag *Zufällige Farben mischen* klicken, um einen neuen Satz zufälliger Farben zu erzeugen.
4. Klicken Sie dann auf die Schaltfläche *Klassifizieren*, um Klassen aus den verschiedenen Werten des angegebenen Feldes oder Ausdrucks zu erstellen.
5. For each class, you can edit the *Legend* column to a more meaningful label (used in the *Layers* panel and the print layout).

Bemerkung

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. Wenn Sie auf *Anwenden* klicken, werden die Änderungen umgesetzt und jedes Objekt auf der Kartenansicht wird mit dem Symbol seiner Klasse dargestellt.

   Standardmäßig fügt QGIS eine Klasse *alle anderen Werte* an die Liste an. Diese Klasse ist zwar zu Beginn leer, wird aber als Standardklasse für alle Objekte verwendet, die nicht unter die anderen Klassen fallen (z.B. wenn Sie ein Objekte mit einem neuen Werten für das Klassifizierungsfeld erstellen).

Die Standardklassifizierung kann weiter angepasst werden:

- You can Add new categories, Remove selected categories, *Delete All* of them or *Delete Unused* categories.
- Eine Klasse kann durch Deaktivieren des Kontrollkästchens links neben dem Klassennamen deaktiviert werden; die entsprechenden Objekte werden auf der Karte ausgeblendet.
- Ziehen Sie die Zeilen per Drag-and-Drop, um die Klassen neu anzuordnen.
- Um das Symbol, den Wert oder die Legende einer Klasse zu ändern, doppelklicken Sie auf das Element.

Wenn Sie mit der rechten Maustaste auf ein ausgewähltes Element klicken, wird ein Kontextmenü angezeigt:

- *Symbol kopieren* und *Symbol einfügen*, eine bequeme Möglichkeit, die Darstellung des Elements auf andere anzuwenden
- *Farbe ändern…* des ausgewählten Symbols/der ausgewählten Symbole
- *Deckkraft ändern…* des/der ausgewählten Symbols/Symbole
- *Ausgabeeinheit ändern…* des gewählten Symbols/der gewählten Symbole
- *Breite ändern…* des/der ausgewählten Liniensymbole(s)

- *Größe ändern…* des/der ausgewählten Punktsymbole(s)
- *Winkel ändern…* des/der ausgewählten Punktsymbole(s)
- *Kategorien zusammenführen*: Fasst mehrere ausgewählte Kategorien zu einer einzigen Kategorie zusammen. Dies ermöglicht eine einfachere Gestaltung von Layern mit einer großen Anzahl von Kategorien, bei denen es möglich sein kann, zahlreiche unterschiedliche Kategorien zu einer kleineren und besser handhabbaren Gruppe von Kategorien zusammenzufassen, die für mehrere Werte gelten.

  Tipp

  Da das Symbol, das für die zusammengeführten Kategorien beibehalten wird, das der obersten ausgewählten Kategorie in der Liste ist, sollten Sie die Kategorie, deren Symbol Sie wiederverwenden möchten, vor der Zusammenführung an den Anfang stellen.

- *Aufheben der Zusammenführung von Kategorien*, die zuvor zusammengeführt wurden

Die erstellten Klassen erscheinen auch in einer Baumhierarchie im *Layer* Bedienfeld. Doppelklicken Sie auf einen Eintrag in der Kartenlegende, um das zugewiesene Symbol zu bearbeiten. Klicken Sie mit der rechten Maustaste und Sie erhalten einige *Weitere Optionen*.

Das Menü *Erweitert* ermöglicht den Zugriff auf Optionen zur Beschleunigung der Klassifizierung oder zur Feinabstimmung der Darstellung von Symbolen:

- *An gespeicherte Symbole angleichen*: Unter Verwendung der *Stilverwaltung* wird jeder Kategorie ein Symbol zugewiesen, dessen Name den Klassifizierungswert der Kategorie darstellt
- *Symbole aus Datei zuordnen…*: Wenn eine Datei mit Symbolen bereitgestellt wird, wird jeder Kategorie ein Symbol zugewiesen, dessen Name den Klassifizierungswert der Kategorie darstellt
- *Symbolebenen…* um die Reihenfolge der Wiedergabe von Symbolen festzulegen.

Abgestufte Darstellung

Der Renderer *Abgestuft* dient zur Darstellung aller Objekte eines Layers unter Verwendung eines benutzerdefinierten Symbols, dessen Farbe oder Größe die Zuordnung des Attributs eines ausgewählten Objekts zu einer Klasse widerspiegelt.

Wie der Renderer „Kategorisiert“ ermöglicht auch der abgestufte Renderer die Definition von Rotation und Größenskala anhand bestimmter Spalten.

Genauso können Sie -analog zum Kategorisierten Renderer - im Menü auswählen:

- Der *Wert* der Klassifizierung kann ein bestehendes Feld oder ein Ausdruck sein, den Sie in das Feld eingeben oder mit der zugehörigen Schaltfläche erstellen können. Durch die Verwendung von Ausdrücken für die Klassifizierung wird die Notwendigkeit vermieden, ein Feld nur für Symbologiezwecke zu erstellen (z. B. wenn Ihre Klassifizierungskriterien von einem oder mehreren Attributen abgeleitet sind).
- Das Symbol (nutzen des Symbolauswahl Dialogs)
- Die Legendenformatierung und die Genauigkeit
- Die Methode, um das Symbol zu ändern: Farbe oder Größe
- Die Farben (indem man die Farbverlaufsliste verwendet), wenn die Farbmethode ausgewählt ist
- Die Größe (unter Verwendung des Größenbereichs und seiner Einheit)

Dann können Sie die Registerkarte Histogramm verwenden, die ein interaktives Histogramm der Werte aus dem zugewiesenen Feld oder Ausdruck anzeigt. Klassenunterbrechungen können mit dem Histogramm-Widget verschoben oder hinzugefügt werden.

Bemerkung

Sie können die statistische Zusammenfassung nutzen, um mehr Informationen über Ihren Vektorlayer zu erhalten. Siehe *Das Statistik-Bedienfeld*.

Zurück zu dem Klassen Reiter, Sie können die Anzahl der Klassen und auch den Modus für das Klassifizieren von Objekten innerhalb der Klassen (indem man die Modus-Liste verwendet) einstellen. Die möglichen Modi sind:

- Gleiche Anzahl (Quantile): Jede Klasse hat die gleiche Anzahl von Elementen (die Idee eines Boxplots).
- Gleiches Intervall: Jede Klasse hat die gleiche Größe (z. B. mit den Werten von 1 bis 16 und vier Klassen hat jede Klasse eine Größe von vier).
- Festes Intervall: Jede Klasse hat einen festen Wertebereich (z. B. bei Werten von 1 bis 16 und einer Intervallgröße von 4 sind die Klassen 1-4, 5-8, 9-12 und 13-16).
- Logarithmischr Maßstab: geeignet für Daten mit einem großen Wertebereich. Enge Klassen für kleine Werte und breite Klassen für große Werte (z. B. für Dezimalzahlen mit einem Bereich [0..100] und zwei Klassen, wobei die erste Klasse von 0 bis 10 und die zweite Klasse von 10 bis 100 reicht).
- Natürliche Unterbrechungen (Jenks): Die Varianz innerhalb jeder Klasse wird minimiert, während die Varianz zwischen den Klassen maximiert wird.
- Schöne Unterbrechungen: Berechnet eine Folge von etwa n+1 gleichmäßig verteilten „schönen“ Werten, die den Bereich der Werte in x abdecken. Die Werte werden so gewählt, dass sie 1, 2 oder 5 mal eine Potenz von 10 sind. (basierend auf pretty aus der R-Statistikumgebung https://www.rdocumentation.org/packages/base/topics/pretty).
- Standardabweichung: Klassen werden in Abhängigkeit von der Standardabweichung der Werte gebildet.

Die Listbox in der Mitte der Registerkarte *Symbology* listet die Klassen mit ihren Bereichen, Bezeichnungen und Symbolen auf, die gerendert werden.

Klicken Sie auf den **Klassifizieren** Knopf um Klassen anhand des ausgewählten Modus zu erstellen. Jede Klasse kann anhand des Deaktivierens des Kontrollkästchens links neben dem Klassennamen ausgeschaltet werden.

Sie können das Symbol, den Wert oder die Beschriftung verändern, klicken Sie einfach auf das Element, das Sie ändern wollen.

Wenn Sie mit der rechten Maustaste auf ein ausgewähltes Element klicken, wird ein Kontextmenü angezeigt:

- *Symbol kopieren* und *Symbol einfügen*, eine bequeme Möglichkeit, die Darstellung des Elements auf andere anzuwenden
- *Farbe ändern…* des ausgewählten Symbols/der ausgewählten Symbole
- *Deckkraft ändern…* des/der ausgewählten Symbols/Symbole
- *Ausgabeeinheit ändern…* des gewählten Symbols/der gewählten Symbole
- *Breite ändern…* des/der ausgewählten Liniensymbole(s)
- *Größe ändern…* des/der ausgewählten Punktsymbole(s)
- *Winkel ändern…* des/der ausgewählten Punktsymbole(s)

Das Beispiel in Abb. 12.5 zeigt den abgestuften Rendering-Dialog für den Layer major_rivers des QGIS-Beispieldatensatzes.

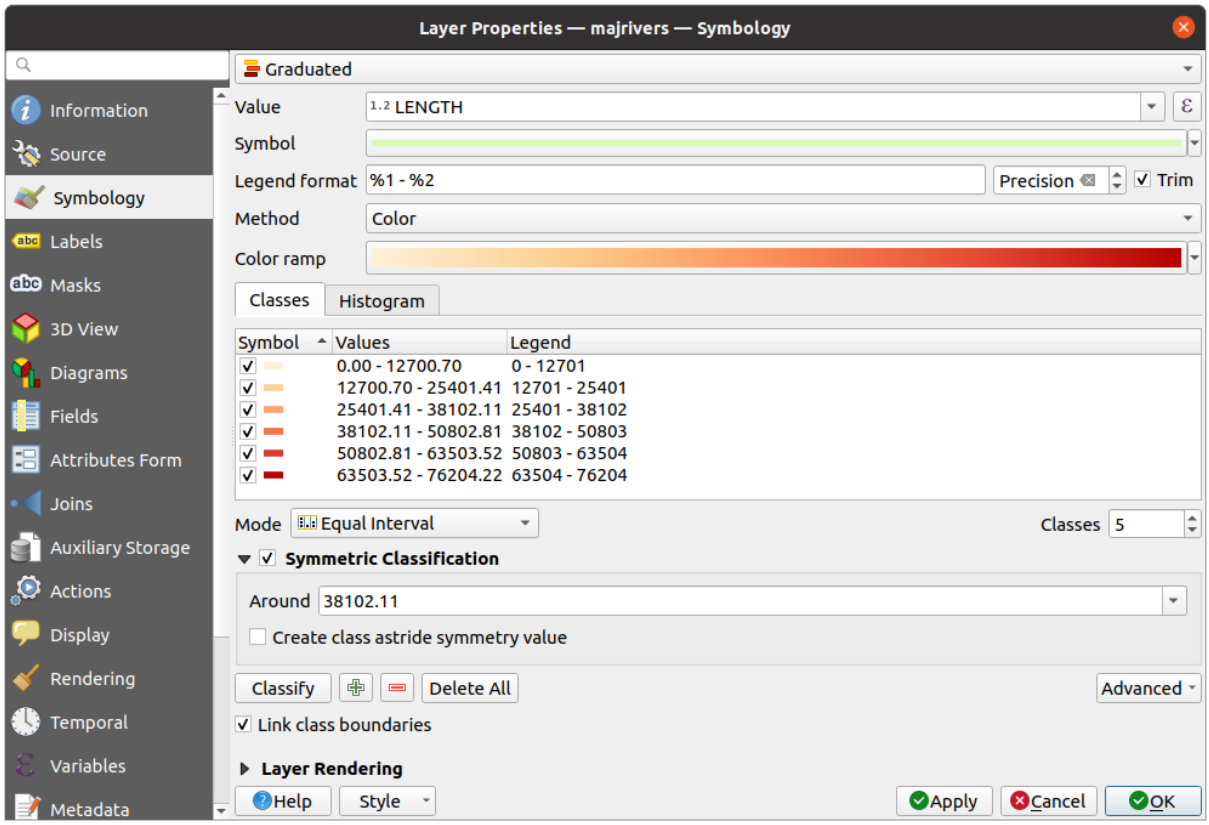


Abb. 12.5: Abgestufte Symbolisierungsoptionen

Die erstellten Klassen erscheinen auch in einer Baumhierarchie im *Layer* Bedienfeld. Doppelklicken Sie auf einen Eintrag in der Kartenlegende, um das zugewiesene Symbol zu bearbeiten. Klicken Sie mit der rechten Maustaste und Sie erhalten einige *Weitere Optionen*.

Proportionale Symbole und mehrdimensionale Analysen

Proportionale Symbole und Multivarianzanalyse sind keine Rendering-Typen, die in der Dropdown-Liste Symbologie-Rendering verfügbar sind. Wenn Sie jedoch die Option *Datendefinierte Übersteuerung* auf eine der vorherigen Rendering-Optionen anwenden, können Sie mit QGIS Ihre Punkt- und Liniendaten mit einer solchen Darstellung anzeigen.

Proportionale Symbole erstellen

Um ein proportionales Rendering anzuwenden:

1. Wenden Sie zunächst auf den Layer den *Einzelsymbol-Renderer* an.
2. Legen Sie dann das Symbol fest, das auf die Objekte angewendet werden soll.
3. Wählen Sie das Element auf der obersten Ebene des Symbolbaums aus und verwenden Sie die Schaltfläche *Datendefinierte Übersteuerung* neben der Option *Größe* (für Point Layer) oder *Strichbreite* (für Linienlayer).
4. Wählen Sie ein Feld aus oder geben Sie einen Ausdruck ein, und QGIS wird für jedes Objekt den Ausgabewert auf die Eigenschaft anwenden und die Größe des Symbols in der Kartenansicht proportional anpassen.

   Falls nötig, verwenden Sie aus dem Menü der Datendefinierten Übersteuerung die Option *Assistent…* um eine Transformation (exponentiell, flannery…) auf die Skalierung der Symbolgröße anzuwenden (für weitere Details siehe *Verwendung des Assistenten für die datendefinierte Übersteuerung*).

Sie können die Proportionalsymbole im *Layer* Bedienfeld und in der *Drucklayout-Legende* anzeigen lassen: Klappen Sie die Dropdown-Liste *Erweitert* am unteren Rand des Hauptdialogs auf der Registerkarte *Symbolisierung* aus

und wählen Sie *Datendefinierte Größenlegende…*, um die Legendenelemente zu konfigurieren (siehe *Datendefinierte Legendengröße* für weitere Einzelheiten).

Mehrdimensionale Analyse erzeugen

Eine mehrdimensionale Analyse hilft Ihnen, die Beziehungen zwischen zwei oder mehr Variablen auszuwerten z. B. kann eine als Farbverlauf und die andere als Größe dargestellt werden.

Die einfachste Art, eine multivariate Analyse in QGIS zu erstellen, ist die folgende:

1. Wenden Sie zunächst ein kategorisiertes oder abgestuftes Rendering auf einen Layer an, indem Sie für alle Klassen denselben Symboltyp verwenden.
2. Wenden Sie dann eine proportionale Symbolik auf die Klassen an:
    1. Klicken Sie auf die *Symbol* Schaltfläche oberhalb des Klassifizierungsrahmens, um zum *Symbolauswahl-Dialog* zu gelangen.
    2. Skalieren Sie die Größe oder Breite der Symbolisierung mithilfe der *Datendefinierte Übersteuerung* wie oben beschrieben.

Wie das proportionale Symbol kann die skalierte Symbolik dem Layer-Baum hinzugefügt werden, und zwar über den kategorisierten oder abgestuften Klassensymbolen unter Verwendung der Option *Datendefinierte Größenlegende*. Beide Darstellungen sind auch in der Drucklayoutlegende verfügbar.

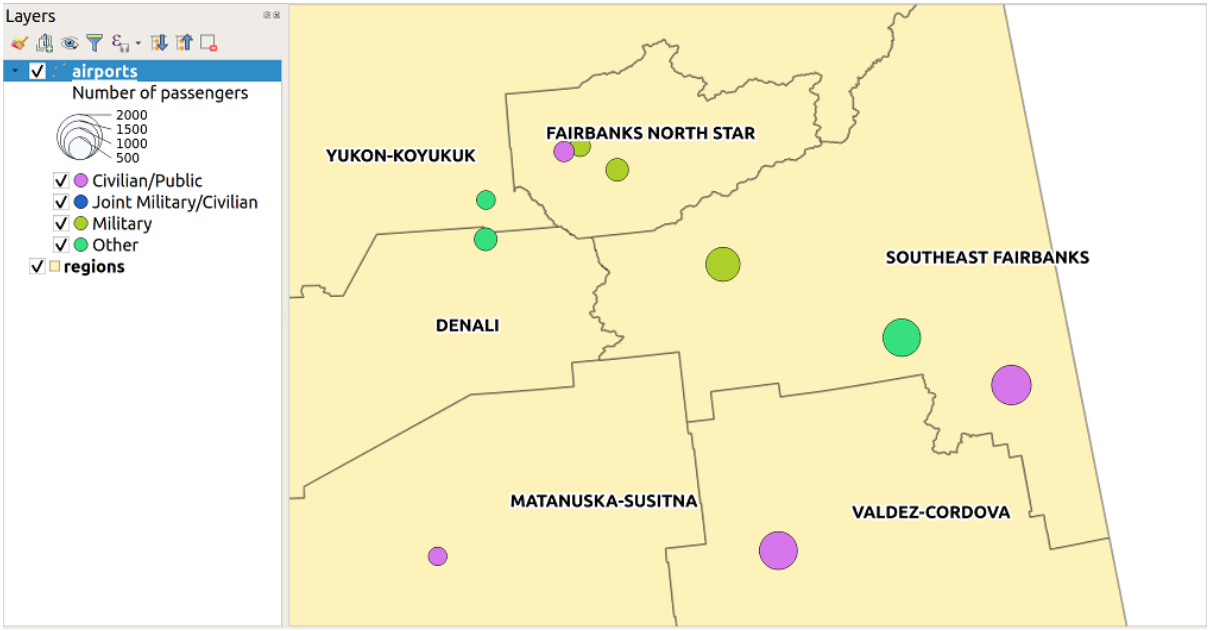


Abb. 12.6: Multivariantes Beispiel mit skalierter Größenlegende

Regelbasierter Renderer

Regeln sind QGIS :ref:`Ausdrücke <vector_expressions> `, die verwendet werden, um Objekte nach ihren Attributen oder Eigenschaften zu unterscheiden, um bestimmte Rendering-Einstellungen auf sie anzuwenden. Regeln können verschachtelt werden, und Objekte gehören zu einer Klasse, wenn sie zu allen oberen Verschachtelungsebenen gehören.

Der Renderer *Regelbasiert* dient also der Darstellung aller Objekte eines Layers unter Verwendung von Symbolen, deren Aspekt die Zuordnung eines ausgewählten Objekts zu einer differenzierten Klasse widerspiegelt.

So erstellen Sie eine Regel:

1. Aktivieren Sie eine bestehende Zeile durch Doppelklick (standardmäßig fügt QGIS automatisch ein Symbol ohne Regel hinzu, wenn das regelbasierte Rendering aktiviert wird) oder klicken Sie auf die Schaltfläche Aktuelle Regel bearbeiten oder Regel hinzufügen.
2. In dem sich öffnenden Dialog *Regel bearbeiten* können Sie eine Beschriftung definieren, die Ihnen hilft, jede Regel zu identifizieren. Diese Beschriftung wird im *Layer* Bedienfeld und auch in der Legende des Drucklayouts angezeigt.
3. Geben Sie manuell einen Ausdruck in das Textfeld neben der Option *Filter* ein oder drücken Sie die Schaltfläche daneben, um den Dialog zur Erstellung von Ausdrucksfolgen zu öffnen.
4. Verwenden Sie die bereitgestellten Funktionen und die Layer-Attribute, um einen *Ausdruck* zu erstellen, um die Objekte zu filtern, die Sie abrufen möchten. Wenn Sie einen Filter definiert haben, können Sie mit der Schaltfläche *Test*, das Ergebnis der Abfrage zu überprüfen.
5. Sie können zusätzlich eine :guilabel:‘Beschreibung , eingeben, um die der Regel zu erläutern.
6. Sie können die Option *Maßstabsbereich* verwenden, um die Maßstäbe festzulegen, innerhalb derer die Regel angewendet werden soll.
7. Konfigurieren Sie schließlich das :ref:-Symbol <symbol-selector>` welches für diese Objekte verwendet werden soll.
8. Und drücken Sie *OK*.

Dem Dialogfeld Layer-Eigenschaften wird eine neue Zeile hinzugefügt, in der die Regel zusammengefasst wird. Sie können so viele Regeln wie nötig erstellen, indem Sie die obigen Schritte wiederholen oder eine bestehende Regel kopieren und einfügen. Ziehen Sie die Regeln per Drag-and-Drop übereinander, um sie zu verschachteln und die oberen Objekte der Regel in Unterklassen zu verfeinern.

Der regelbasierte Renderer kann mit kategorisierten oder abgestuften Renderern kombiniert werden. Wenn Sie eine Regel auswählen, können Sie ihre Objekte in Unterklassen organisieren, indem Sie das Dropdown-Menü *Ausgewählte Regeln verfeinern* verwenden. Die verfeinerten Klassen werden wie die Unterelemente der Regel in einer Baumhierarchie angezeigt, und wie bei der übergeordneten Klasse können Sie die Symbolik und die Regel jeder Klasse festlegen. Die automatische Regelverfeinerung kann auf folgender Grundlage erfolgen:

- **Maßstäbe**: Aus einer Liste von Maßstäben erzeugt diese Option eine Reihe von Klassen, für die die verschiedenen benutzerdefinierten Maßstäbe gelten. Jede neue maßstabsbasierte Klasse kann ihre eigene Symbolik und ihren eigenen Definitionsausdruck haben. Dies kann z. B. eine bequeme Möglichkeit sein, dieselben Objekte mit verschiedenen Symbolen in unterschiedlichen Maßstäben darzustellen oder nur eine Reihe von Objekten in Abhängigkeit vom Maßstab anzuzeigen (z. B. lokale Flughäfen im großen Maßstab gegenüber internationalen Flughäfen im kleinen Maßstab).
- **Kategorien**: wendet einen *kategorisierten Renderer* auf die Objekte an, die unter die ausgewählte Regel fallen.
- oder **Bereiche**: wendet einen *graduierten Renderer* auf die Objekte an, die unter die ausgewählte Regel fallen.

Verfeinerte Klassen erscheinen wie Unterelemente der Regel in einer Baumhierarchie, und wie oben können Sie die Symbolik jeder Klasse festlegen. Die Symbole der verschachtelten Regeln werden übereinander gestapelt, seien Sie also vorsichtig bei der Auswahl der Symbole. Es ist auch möglich, das Häkchen bei *Symbol* im Dialog *Regel bearbeiten* zu entfernen, um die Darstellung eines bestimmten Symbols im Stapel zu vermeiden.

Im Dialog *Regel bearbeiten* können Sie die Option *Sonst* verwenden, um alle Objekte zu erfassen, die keiner der anderen Regeln entsprechen (auf derselben Ebene). Dies kann auch erreicht werden, indem man `ELSE` in die Spalte *Regel* im Hauptfenster des Dialogs :guilabel:‘Regelbasierend“ schreibt.

Wenn Sie mit der rechten Maustaste auf ein ausgewähltes Element klicken, wird ein Kontextmenü angezeigt:

- *Kopieren* und *Einfügen* stellt eine bequeme Möglichkeit dar, neue Regeln auf der Grundlage vorhandener zu erstellen
- *Symbol kopieren* und *Symbol einfügen*, eine bequeme Möglichkeit, die Darstellung des Elements auf andere anzuwenden
- *Farbe ändern…* des ausgewählten Symbols/der ausgewählten Symbole

- *Deckkraft ändern…* des/der ausgewählten Symbols/Symbole
- *Ausgabeeinheit ändern…* des gewählten Symbols/der gewählten Symbole
- *Breite ändern…* des/der ausgewählten Liniensymbole(s)
- *Größe ändern…* des/der ausgewählten Punktsymbole(s)
- *Winkel ändern…* des/der ausgewählten Punktsymbole(s)
- *Aktuelle Regel verfeinern*: öffnet ein Untermenü, das es erlaubt, die aktuelle Regel mit **Maßstäben**, **Kategorien** oder **Bereichen** zu verfeinern. Dasselbe gilt für die Auswahl des *entsprechenden Menüs* am unteren Rand des Dialogs.

Wenn Sie eine Zeile im Dialogfeld des regelbasierten Renderers deaktivieren, werden die Objekte der betreffenden Regel und die verschachtelten Objekte in der Kartenansicht ausgeblendet.

Die erstellten Regeln erscheinen auch in einer Baumhierarchie in der Kartenlegende. Doppelklicken Sie auf einen Eintrag in der Kartenlegende, um das zugewiesene Symbol zu bearbeiten. Klicken Sie mit der rechten Maustaste und Sie erhalten *mehr Optionen*.

Das Beispiel in Abb. 12.7 zeigt den regelbasierten Rendering-Dialog für den Flüsse-Layer des QGIS-Beispieldatensatzes.

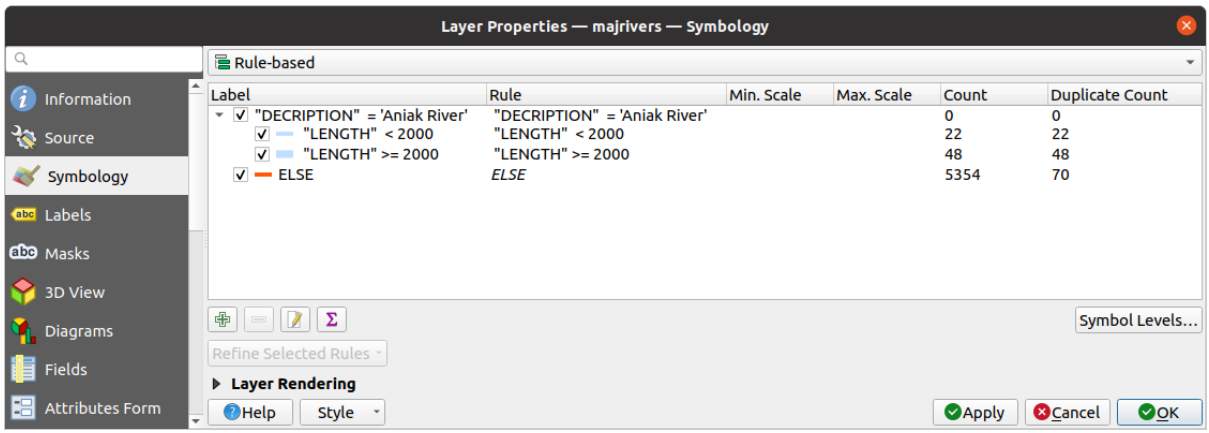


Abb. 12.7: Regelbasierte Symbolisierungsoptionen

Punktversatz-Renderer

Der Renderer *Point Displacement* nimmt die punktförmigen Objekte, die in einer bestimmten Entfernungstoleranz zueinander liegen, und platziert ihre Symbole um ihr jeweiliges Zentrum, wobei verschiedene Platzierungsmethoden angewendet werden. Dies kann eine bequeme Möglichkeit sein, alle Objekte eines Layers zu visualisieren, auch wenn sie den gleichen Standort haben (z. B. Einrichtungen in einem Gebäude).

Um einen Punktversatz-Renderer zu konfigurieren, müssen Sie Folgendes tun:

1. Setzen Sie das *Zentrumssymbol*: wie der virtuelle Punkt in der Mitte aussehen soll
2. Wählen Sie den *Renderer*-Typ: wie Sie Objekte im Layer klassifizieren wollen (Einzelsymbol, kategorisiert, regelbasiert…)
3. Drücken Sie die Schaltfläche *Darstellungseinstellungen…*, um die Symbologie der Objekte entsprechend dem ausgewählten Renderer zu konfigurieren
4. Geben Sie die *Distance*-Toleranz an, bei der nahe Objekte als überlappend betrachtet und dann über denselben virtuellen Punkt verschoben werden. Unterstützt gemeinsame Symboleinheiten.
5. Konfigurieren Sie die *Platzierungsmethode*:

- **Ring**: places all the features on a circle whose radius depends on the number of features to display.
- **Concentric rings**: uses a set of concentric circles to show the features.
- **Grid**: generates a regular grid with a point symbol at each intersection.

6. Verdrängte Symbole werden auf den *Verlagerungslinien* platziert. Während der minimale Abstand der Verlagerungslinien vom Renderer der Punktsymbole abhängt, können Sie dennoch einige ihrer Einstellungen wie *Strichbreite*, *Strichfarbe* und *Größenanpassung* anpassen (z.B. um mehr Abstand zwischen den gerenderten Punkten zu schaffen).
7. Verwenden Sie den Bereich *Beschriftungen*, um eine Punktbeschriftung durchzuführen: die Beschriftungen werden in der Nähe des verschobenen Symbols und nicht an der tatsächlichen Position des Objekts platziert.
    1. Wählen Sie das *Beschriftungsattribut*: ein Feld des Layers, das für die Beschriftung verwendet werden soll
    2. Geben Sie die Eigenschaften und die Größe der *Schriftart* an
    3. Wählen Sie eine *Beschriftungsfarbe*
    4. Legen Sie einen *Beschriftungsabstandsfaktor* fest: für jedes punktförmige Objekt wird die Beschriftung proportional zur Diagonale des Symbols von der Symbolmitte aus verschoben.
    5. Schalten Sie *Maßstabsabhängige Beschriftung benutzen* ein, wenn Sie Beschriftungen nur in Maßstäben anzeigen wollen, die größer sind als der angegebene *Minimaler Kartenmaßstab*.

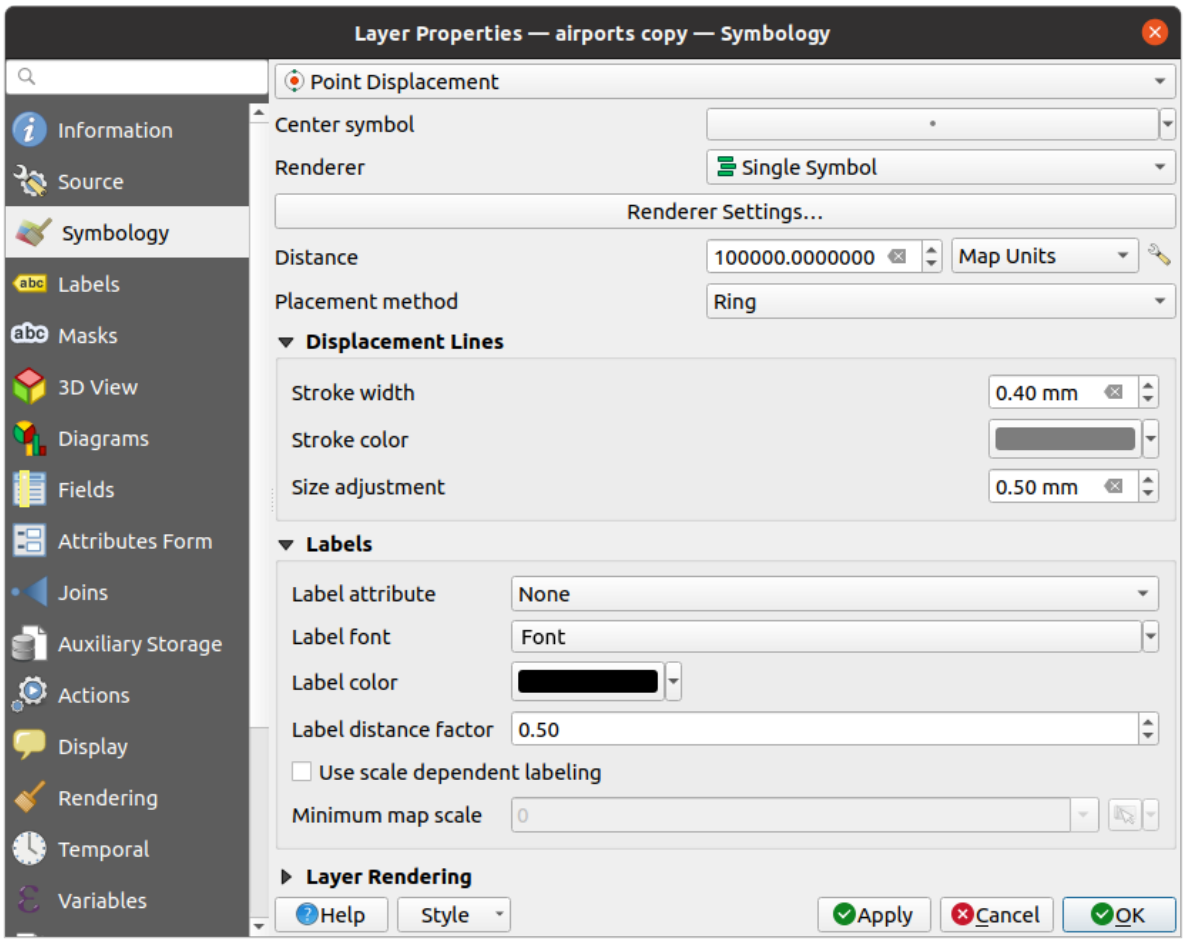


Abb. 12.8: Dialog Punktversatz

Bemerkung

Der Punktversatz-Renderer verändert die Geometrie des Objekts nicht, d. h. die Punkte werden nicht von ihrer Position verschoben. Sie befinden sich immer noch an ihrem ursprünglichen Ort. Die Änderungen sind nur visuell, zum Zweck des Renderns. Verwenden Sie stattdessen den Verarbeitungsalgorithmus *Points displacement*, wenn Sie verschobene Objekte erstellen möchten.

Punkthäufung-Renderer

Im Gegensatz zum Renderer *Punktversatz*, der die Darstellung der nächstgelegenen oder überlagerten Objekte vergrößert, fasst der Renderer *Punkthäufung* nahe beieinander liegende Punkte zu einem einzigen Markersymbol zusammen. Punkte, die innerhalb eines bestimmten Abstands zueinander liegen, werden zu einem einzigen Symbol zusammengeführt. Die Zusammenfassung der Punkte erfolgt auf der Grundlage der nächstgelegenen Gruppe, die gebildet wird, anstatt ihnen einfach die erste Gruppe innerhalb der Suchdistanz zuzuweisen.

Vom Hauptdialog aus können Sie:

1. Legen Sie das Symbol fest, das den Punktcluster im *Häufungssymbol* repräsentieren soll; die Standarddarstellung zeigt die Anzahl der aggregierten Objekte dank der *Variable* `@cluster_size` als einfaches Markierungssymbol mit einer zusätzlichen Schriftmarkierung für die Anzahl.
2. Wählen Sie den *Renderer*-Typ, d.h. wie Sie Objekte im Layer klassifizieren wollen (Einzelsymbol, kategorisiert, regelbasiert…)
3. Drücken Sie die Schaltfläche *Darstellungseinstellungen…*, um die Symbolisierung der Objekte wie gewohnt zu konfigurieren. Beachten Sie, dass diese Symbolik nur bei Objekten sichtbar ist, die nicht geclustert sind, da sonst das *Häufungssymbol* verwendet wird. Wenn alle Objekte eines Clusters zur gleichen Darstellungs-Klasse gehören und daher die gleiche Farbe erhalten, entspricht diese Farbe der Variable `@cluster_color` des Clusters.
4. Geben Sie den maximalen *Abstand* an, der für das Clustering von Objekten berücksichtigt werden soll. Unterstützt werden die üblichen Symboleinheiten.

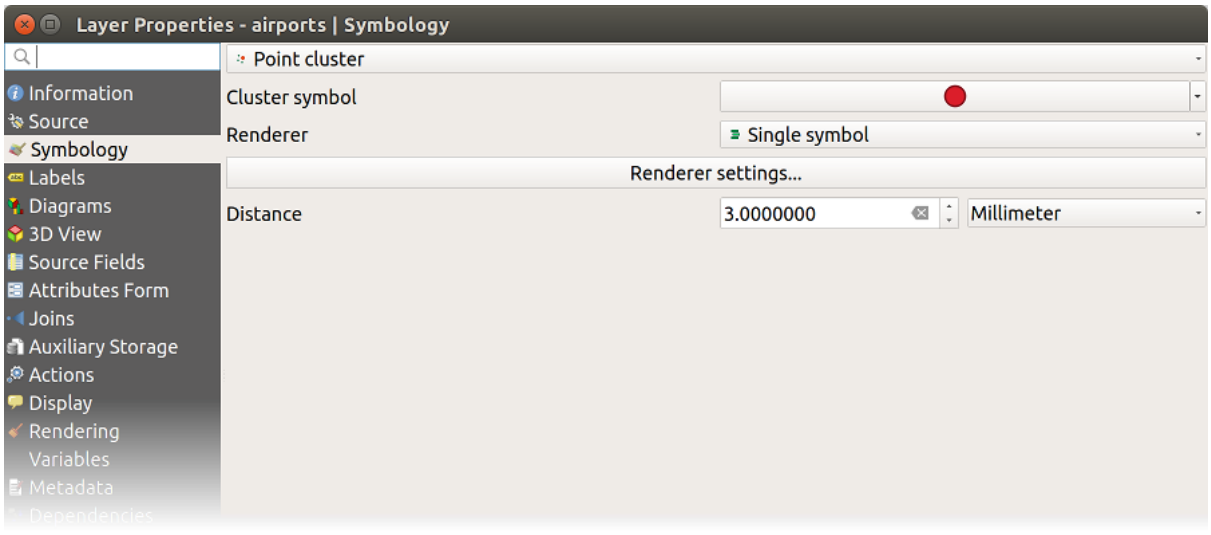


Abb. 12.9: Dialog Punkthäufung

Bemerkung

Der Punkthäufung-Renderer verändert die Geometrie der Objekte nicht, d. h. die Punkte werden nicht von ihrer Position verschoben. Sie befinden sich immer noch an ihrem ursprünglichen Ort. Die Änderungen sind nur visuell,

zum Zweck des Renderns. Verwenden Sie stattdessen den Verarbeitungsalgorithmus *K-means clustering* oder *DBSCAN clustering*, wenn Sie clusterbasierte Objekte erstellen möchten.

Objekte verschmolzen Renderer

Mit dem Renderer *Objekte verschmolzen* können Flächen- und Linienfeatures vor dem Rendern in ein einziges Objekt „aufgelöst“ werden, um sicherzustellen, dass komplexe Symbole oder sich überlappende Objekte durch ein einheitliches und zusammenhängendes kartografisches Symbol dargestellt werden.

Invertierte Polygone Renderer

Der Renderer *Invertierte Polygone* ermöglicht es dem Benutzer, ein Symbol zu definieren, das außerhalb der Polygone des Layers ausgefüllt wird. Auch hier können Sie Unter-Renderer auswählen, nämlich Einzelsymbol, abgestuft, kategorisiert, regelbasierend oder 2.5 D-Renderer.

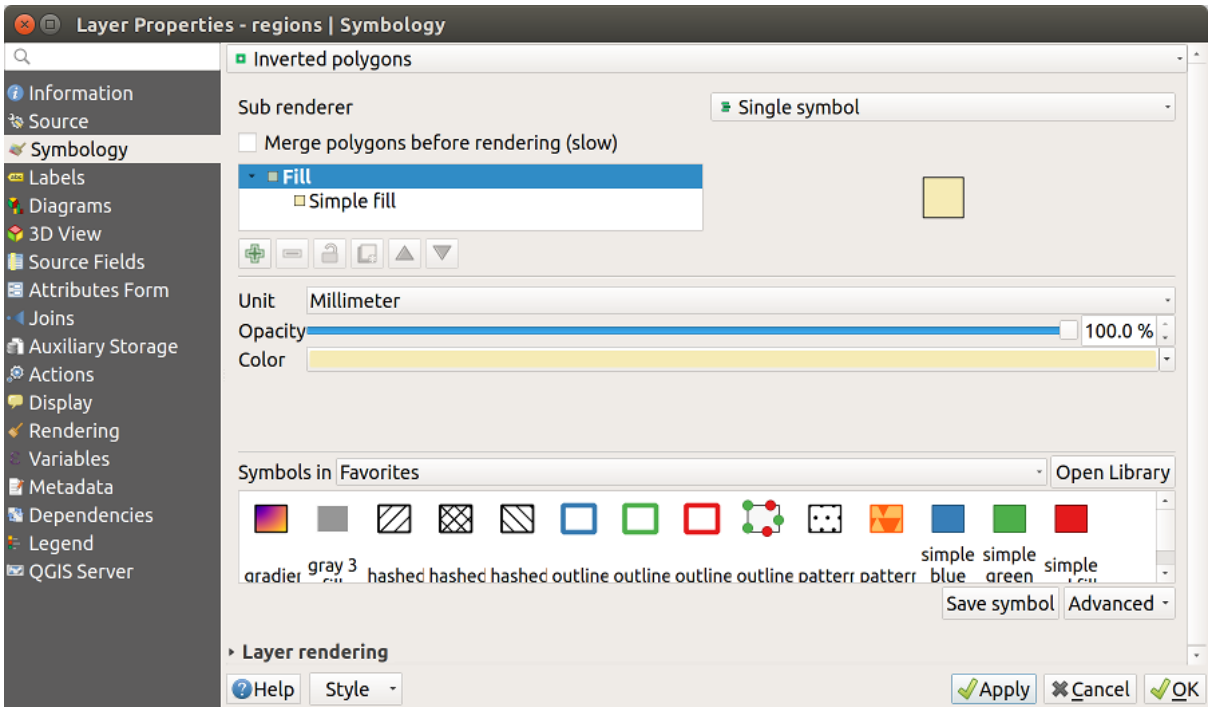


Abb. 12.10: Invertierte Polygone Dialog

Heatmap Renderer

Mit dem *Heatmap*-Renderer können Sie dynamische Live-Heatmaps für (Multi-)Punkt-Layer erstellen. Sie können den Heatmap-*Radius* in Millimetern, Punkten, Pixeln, Karteneinheiten oder Zoll angeben, eine *Farbverlauf* für den Heatmap-Stil auswählen und bearbeiten und mit einem Schieberegler einen Kompromiss zwischen Rendergeschwindigkeit und Qualität wählen. Sie können auch einen *Maximalwert* und *Gewichtungspunkte nach* mithilfe eines Feldes oder eines Ausdrucks definieren.

Verwenden Sie datendefinierte Übersteuerung, um *Radius* und *Maximalwert* basierend auf den Attributen Ihrer Daten dynamisch zu steuern. Beispielsweise könnte der Radius eines Heatmap-Punkts durch sein Bevölkerungsattribut bestimmt werden oder der Maximalwert könnte auf einem zeitlichen Bereich basieren.

Beim Hinzufügen oder Entfernen einer Funktion aktualisiert der Heatmap-Renderer automatisch den Heatmap-Stil. Der *Farbverlauf* wird als Legendenleiste angezeigt, und in den *Legendeneinstellungen* können Sie die *Beschriftungen* für die *Maximal-* und *Minimal*-Werte festlegen. Außerdem können Sie die Ausrichtung und Richtung der Legende im *Layout* ändern.

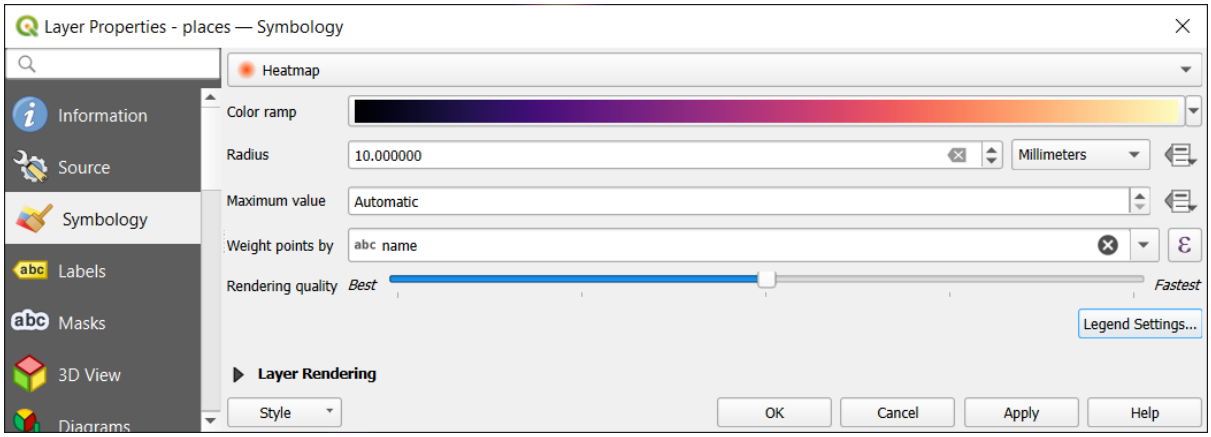


Abb. 12.11: Der Heatmap Dialog

2.5D Renderer

Mit dem Renderer *2.5D* ist es möglich, einen 2.5D-Effekt auf den Objekten Ihres Layers zu erzeugen. Sie beginnen mit der Auswahl eines Wertes für die *Höhe* (in Karteneinheiten). Hierfür können Sie einen festen Wert, eines der Felder Ihres Layers oder einen Ausdruck verwenden. Sie müssen auch einen *Winkel* (in Grad) wählen, um die Position des Betrachters zu bestimmen (0° bedeutet Westen, im Gegenuhrzeigersinn wachsend). Verwenden Sie die erweiterte Konfiguration, um die *Dachfarbe* und *Wandfarbe* einzustellen. Wenn Sie die Sonneneinstrahlung auf die Wände des Objekts simulieren möchten, müssen Sie die Option *Mauern nach Perspektive schattieren* aktivieren. Sie können auch einen Schatten simulieren, indem Sie eine *Farbe* und *Größe* (in Karteneinheiten) festlegen.

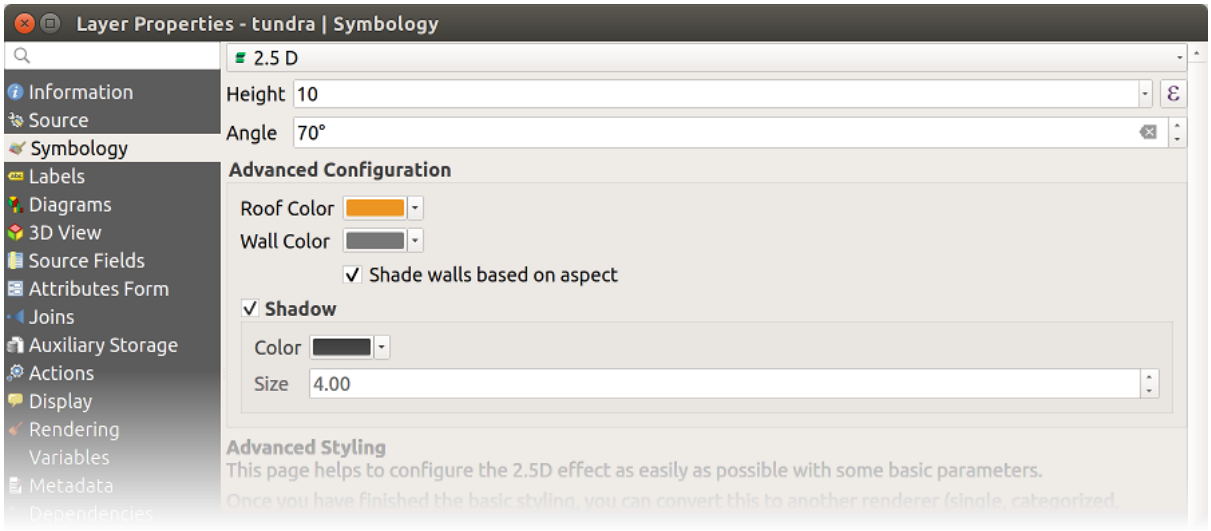


Abb. 12.12: 2.5D Dialog

Tipp

Verwendung des 2.5D-Effekts mit anderen Renderern

Sobald Sie den grundlegenden Stil für den 2,5D-Renderer festgelegt haben, können Sie diesen in einen anderen Renderer (Einzelsymbol, kategorisiert, abgestuft) konvertieren. Die 2,5D-Effekte werden beibehalten und alle anderen Renderer-spezifischen Optionen stehen Ihnen zur Feinabstimmung zur Verfügung (auf diese Weise können Sie z. B. kategorisierte Symbole mit einer 2,5D-Darstellung haben oder Ihren 2,5D-Symbolen ein zusätzliches Styling hinzufügen). Um sicherzustellen, dass der Schatten und das „Gebäude“ selbst nicht mit anderen Objekten in der Nähe kollidieren, müssen Sie eventuell die Symbolebenen aktivieren (*Erweitert* ► *Symbolebenen…*). Die 2,5D-Höhen- und -Winkelwerte werden in den Variablen des Layers gespeichert, sodass Sie sie nachträglich auf der Registerkarte Variablen im Dialogfeld Eigenschaften des Layers bearbeiten können.

Eingebette Symbole Renderer

Der :guilabel:` Eingebettete Symbole` Renderer erlaubt es, die ‚native‘ Symbologie einer bereitgestellten Datenquelle anzuzeigen. Dies ist vor allem bei `KML` und `TAB`-Datensätzen der Fall, die eine vordefinierte Symbologie haben.

Layerdarstellung

Auf der Registerkarte Symbolisierung können Sie auch einige Optionen einstellen, die sich immer auf alle Objekte des Layers auswirken:

- *Deckkraft* : Mit diesem Werkzeug können Sie unter diesem Layer liegenden Objekte in der Kartenansicht sichtbar machen. Verwenden Sie den Schieberegler, um die Sichtbarkeit Ihres Vektorlayers an Ihre Bedürfnisse anzupassen. Sie können den Prozentsatz der Sichtbarkeit auch im Menü neben dem Schieberegler genau festlegen.
- *Mischmodus* auf den Ebenen *Layer* und *Objekt*: Mit diesen Werkzeugen können Sie spezielle Rendering-Effekte erzielen, die Sie bisher vielleicht nur aus Grafikprogrammen kennen. Die Pixel Ihrer überlagernden und unterlagernden Layer werden durch die in *Mischmodi* beschriebenen Einstellungen gemischt.
- Mit dem *Zeicheneffekte* Knopf können Sie *Zeicheneffekte* auf alle Layerobjekte anwenden.
- Mit *Objektzeichenreihenfolge steuern* können Sie die z-Reihenfolge festlegen, in der die Objekte des Layers gerendert werden sollen. Aktivieren Sie das Kontrollkästchen und klicken Sie auf die Schaltfläche daneben. Sie gelangen zum Dialog *Reihenfolge festlegen*, in dem Sie:

  1. Wählen Sie ein Feld oder erstellen Sie einen Ausdruck, der auf die Objekte des Layers angewendet werden soll.
  2. Legen Sie fest, in welcher Reihenfolge die abgerufenen Objekte sortiert werden sollen, d. h. wenn Sie **Aufsteigend** wählen, werden die Objekte mit niedrigerem Wert unter denen mit höherem Wert angezeigt.
  3. Definieren Sie, wann Objekte, die den Wert NULL zurückgeben, gerendert werden sollen: **am Anfang** (unten) oder **am Ende** (oben).
  4. Wiederholen Sie die obigen Schritte so oft, wie Sie Regeln verwenden möchten.

  Die erste Regel wird auf alle Objekte des Layers angewandt, wobei sie nach ihrem zurückgegebenen Wert in der Z-Ebene geordnet werden. Dann wird innerhalb jeder Gruppe von Objekten mit demselben Wert (einschließlich derjenigen mit NULL-Wert) und somit derselben Z-Ebene die nächste Regel angewandt, um sie zu sortieren. Und so weiter…

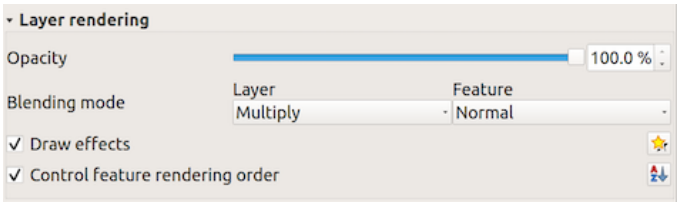


Abb. 12.13: Layerdarstellungsoptionen

Andere Einstellungen

Symbolebenen

Für Renderer, die Symbollayer gestapelt ermöglichen (nur Heatmap nicht) gibt es eine Option, um die Darstellungsreihenfolge der einzelnen Symbolebenen zu steuern.

Bei den meisten Renderern können Sie auf die Option Symbolebenen zugreifen, indem Sie auf die Schaltfläche *Erweitert* unterhalb der Liste der gespeicherten Symbole klicken und *Symbolebenen* wählen. Für den *Regelbasierter Renderer* ist die Option direkt über die Schaltfläche *Symbolebenen…* verfügbar, während sich für den Renderer *Punktversatz-Renderer* die gleiche Schaltfläche im Dialog *Rendering settings* befindet.

Um die Symbolebenen zu aktivieren, wählen Sie *Symbolebenen aktivieren*. Jede Reihe zeigt eine kleine Vorschau des kombinierten Symbols, seiner Beschriftung und die individuellen Symbollayer, unterteilt in verschiedene Spalten mit einer Nummer. Die Nummern zeigen die Darstellungsreihenfolge, in der die Symbollayer gezeichnet werden. Niedrige Werteebenen werden zuerst gezeichnet, liegen ganz unten, während höhere Werte als letztes gezeichnet werden und über den anderen liegen.

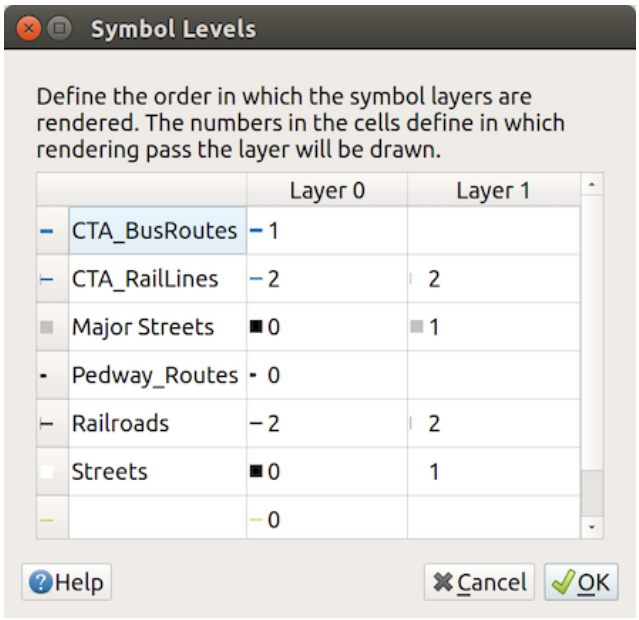


Abb. 12.14: Symbolebenen Dialog

Bemerkung

Wenn Symbolebenen deaktiviert werden, werden alle Symbole entsprechend ihren jeweiligen Objektreihenfolge gezeichnet. Überlappende Symbole werden einfach zu anderen darunter verschleiert. Außerdem „verschmelzen“ ähnliche Symbole nicht mit anderen.

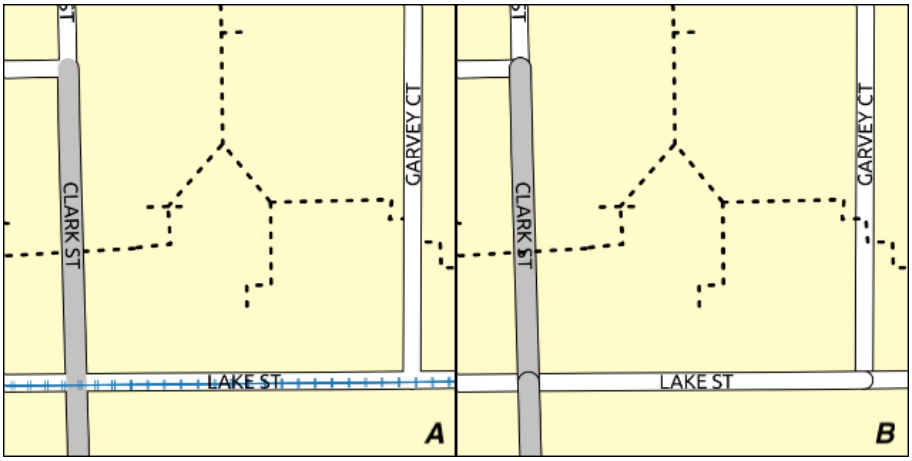


Abb. 12.15: Unterschied von aktivierten (A) und deaktivierten (B) Symbolebenen

Datendefinierte Legendengröße

Wenn ein Layer den *Proportionale Symbole und Multivarianzanalyse* Renderer verwendet oder wenn ein *Diagramm mit skalierter Größe* auf den Layer angewendet ist, können Sie die Anzeige der skalierten Symbole sowohl im *Layer* Bedienfeld als auch in der *Legende im Druck-Layout* erlauben.

Um das Dialogfeld *Datendefinierte Größenlegende* zur Darstellung von Symbolen zu aktivieren, wählen Sie die gleichlautende Option in der Schaltfläche *Erweitert* unterhalb der Liste der gespeicherten Symbole. Für Diagramme ist die Option unter der Registerkarte *Legende* verfügbar. Der Dialog bietet die folgenden Optionen an:

- wählen Sie die Art der Legende: *Legende nicht aktiviert*, *Einzelne Legendenelemente* und *Eingeklappte Legende*. Bei der letztgenannten Option können Sie auswählen, ob die Legendenelemente am **Unten** oder in der **Zentriert** ausgerichtet werden sollen;
- Vorschau des :ref:-Symbols <symbol-selector>` welches für die Legendendarstellung verwenden werden soll;
- fügen Sie den Titel in die Legende ein;
- Ändern Sie die Größe der zu verwendenden Klassen: QGIS stellt Ihnen standardmäßig eine Legende mit fünf Klassen zur Verfügung (basierend auf natürlichen Unterbrechungen), aber Sie können Ihre eigene Klassifizierung anwenden, indem Sie die Option *Manuelle Klassengrößen* verwenden. Verwenden Sie die Schaltflächen und , um die Werte und Bezeichnungen Ihrer eigenen Klassen festzulegen.
- Für eingeklappte Legenden ist es möglich:
  - *Symbole ausrichten* unten oder zentriert
  - konfigurieren Sie die horizontale Führungslinie *Liniensymbol* vom Symbol zum entsprechenden Legendentext.

Eine Vorschau der Legende wird im rechten Bedienfeld des Dialogs angezeigt und aktualisiert, wenn Sie die Parameter einstellen.

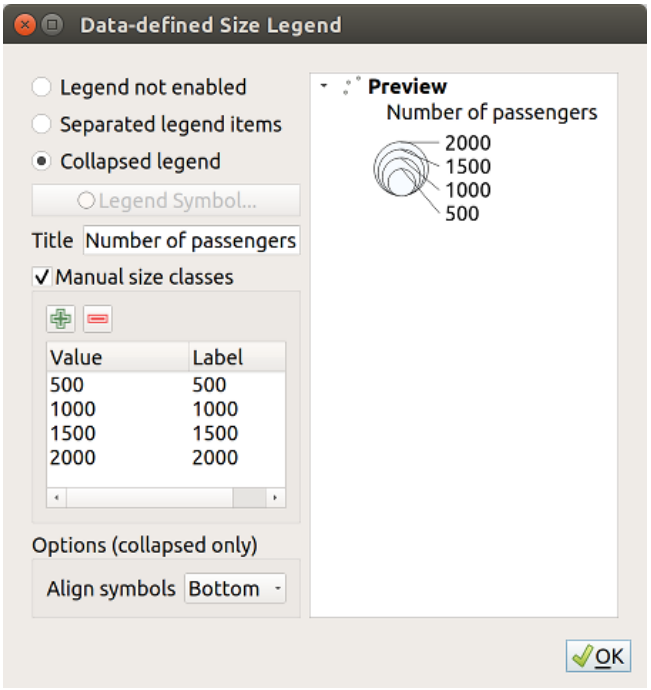


Abb. 12.16: Einstellungen der datendefinierten Größenlegende

Bemerkung

Gegenwärtig kann die datendefinierte Größenlegende für Layer-Symbologie nur auf Punkt-Layer mit einfacher, kategorisierter oder abgestufter Symbologie angewendet werden.

Animationseinstellungen

Um ein beliebiges Symbol zu einem *Animierten Symbol* zu machen, können Sie die *Animationseinstellungen* verwenden. In diesem Dialog können Sie die Animation für das Symbol aktivieren und eine bestimmte Bildrate für das Neuzeichnen des Symbols festlegen.

1. Gehen Sie zunächst auf die oberste Symbolebene und wählen Sie das Menü *Erweitert* in der rechten unteren Ecke des Dialogs
2. Wählen Sie die Option *Animationseinstellungen* aus
3. Aktivieren Sie das Kontrollkästchen *Ist animiert*, um die Animation für das Symbol zu aktivieren
4. Konfigurieren Sie die *Bildrate*, d.h. wie schnell die Animation abgespielt werden soll
5. Sie können nun die Variable `@symbol_frame` in jeder Eigenschaft, die in den Untersymboldaten definiert ist, verwenden, um diese Eigenschaft zu animieren.

Wenn Sie zum Beispiel die Rotation des Symbols auf den Ausdruck `@symbol_frame % 360` setzen, wird das Symbol mit der Zeit rotieren, wobei die Rotationsgeschwindigkeit durch die Bildrate des Symbols bestimmt wird:

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. Gehen Sie zunächst auf die oberste Symbolebene und wählen Sie das Menü *Erweitert* in der rechten unteren Ecke des Dialogs
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)`:

Zeicheneffekte

Damit die Layerdarstellung verbessert und vermieden (oder zumindest reduziert) wird, dass andere Software auf die endgültige Darstellung der Karte umsortiert, bietet QGIS eine weitere leistungsfähige Funktionalität: Die Option *Zeicheneffekte*, die zur Anpassung Zeicheneffekte für die Visualisierung von Vektorlayern hinzufügt.

Die Option ist im Dialog *Layereigenschaften* ► *Symbolisierung* im Bereich *Layerdarstellung* (gilt für den gesamten Layer) oder unter den *Symboleigenschaften* (gilt für entsprechende Objekte) verfügbar. Sie können beide Verwendungen kombinieren.

Zeicheneffekte können aktiviert werden, indem man die Option *Zeicheneffekte* aktiviert und auf die Schaltfläche Effekte einstellen klickt. Das öffnet den Dialog *Effekteigenschaften* (siehe Abb. 12.17). Die folgenden Effekttypen mit eigenen Optionen sind verfügbar:

- **Quelle**: Zeichnet das Symbol mit dem ursprünglichen Stil entsprechend der Konfiguration der Layereigenschaften. Die *Deckkraft* kann angepasst werden, ebenso wie der *Mischmodus* und der *Zeichnungsmodus*. Diese Eigenschaften lassen sich bei allen Effekten konfigurieren.

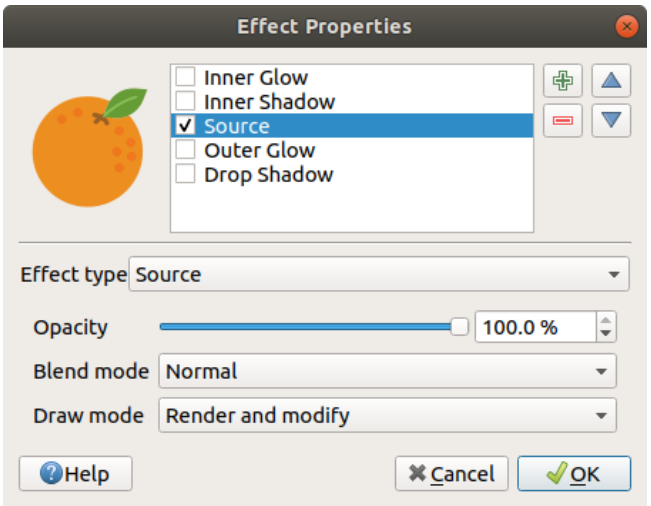


Abb. 12.17: Zeicheneffekte: Dialog Quelle

- **Verwischen**: Fügt dem Layer einen Weichzeichnungseffekt hinzu. Die benutzerdefinierten Optionen, die Sie ändern können, sind der *Verwischtyp* (*Stapelunschärfe (schnell,…)* oder *Gaußscher Unschärfe (Qualität, …)*) und die *Verwischstärke*.

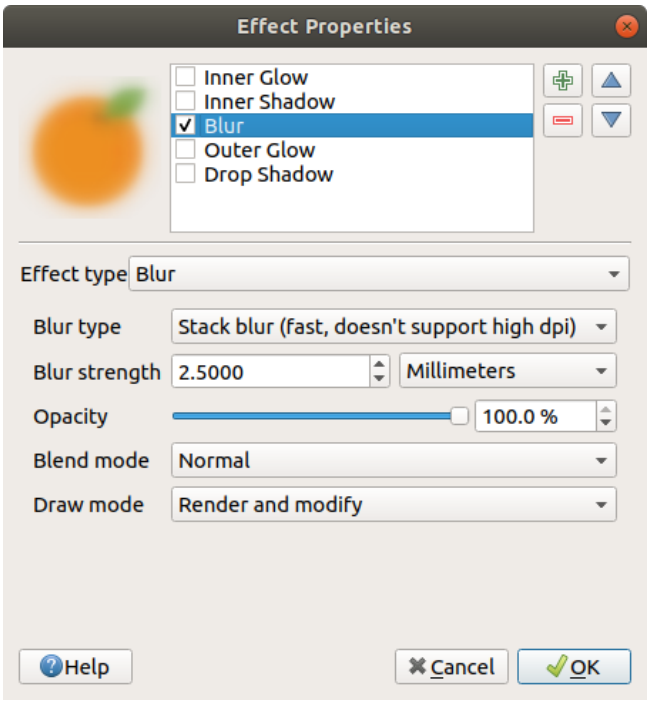


Abb. 12.18: Zeicheneffekte: Dialog verwischen

- **Einfärben**: Dieser Effekt kann verwendet werden, um eine Version des Stils mit einem einzigen Farbton zu erstellen. Die Basis wird immer eine Graustufenversion des Symbols sein und Sie können:
  - Verwenden Sie die Option *Graustufen*, um auszuwählen, wie es erstellt werden soll: Die Optionen sind ‚Nach Helligkeit', ‚Nach Leuchtkraft', ‚Nach Durchschnitt' und ‚Aus'.
  - Wenn *Einfärben* ausgewählt ist, kann man eine andere Farbe mischen und wählen, wie stark sie sein soll.
  - Steuern Sie die *Helligkeit*, *Kontrast* und *Sättigung* des resultierenden Symbols.

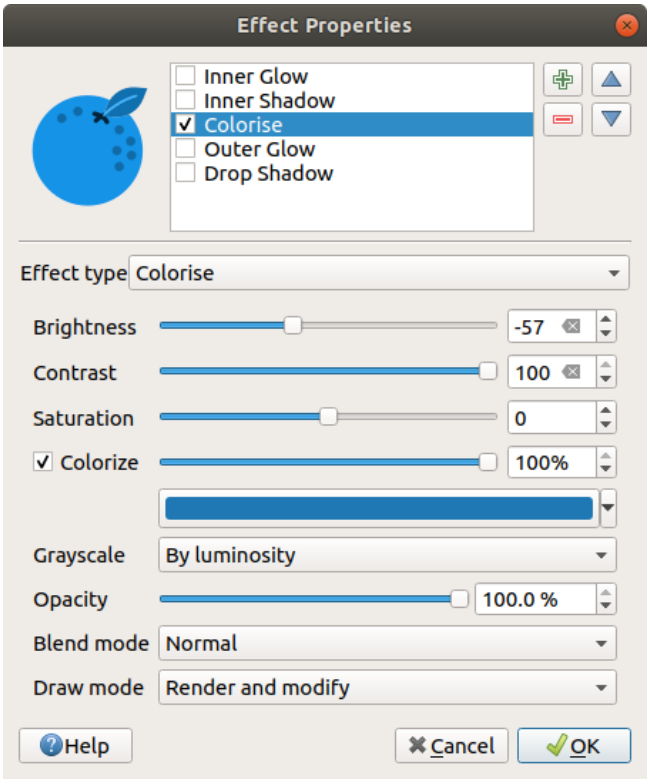


Abb. 12.19: Zeicheneffekte: Dialog einfärben

- **Schattenwurf**: Mit diesem Effekt wird ein Schatten auf das Objekt gelegt, der wie eine zusätzliche Dimension aussieht. Dieser Effekt kann durch Ändern des Winkels und des Abstands von *Versatz* angepasst werden, wodurch bestimmt wird, wohin sich der Schatten verschiebt und wie nah er am Ausgangsobjekt liegt. *Schattenwurf* hat auch die Möglichkeit, den *Verwischungsradius* und die *Farbe* des Effekts zu ändern.

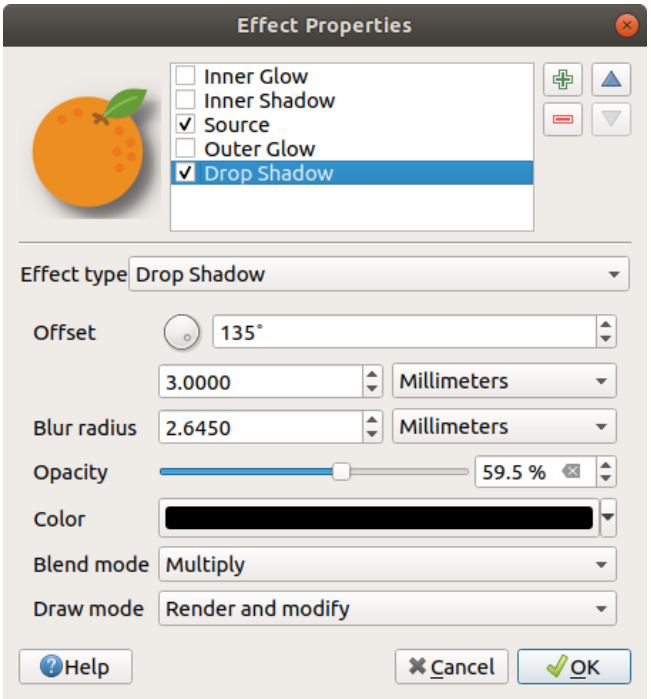


Abb. 12.20: Zeicheneffekte: Dialog Schattenwurf

- **Innerer Schatten**: Dieser Effekt ähnelt dem Effekt *Schattenwurf*, fügt aber den Schatteneffekt an der Innenseite der Kanten des Objekts hinzu. Die verfügbaren Optionen zur Anpassung sind dieselben wie beim Effekt *Schattenwurf*.

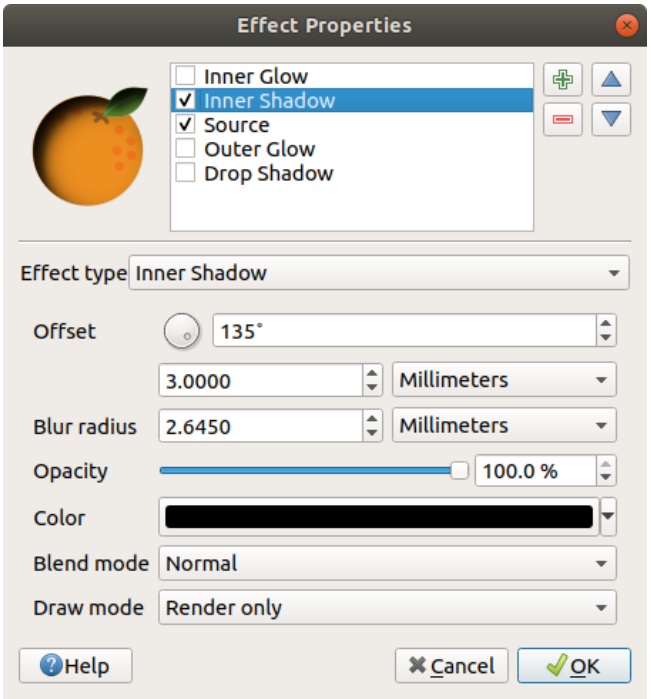


Abb. 12.21: Zeicheneffekte: Dialog Innerer Schatten

- **Inneres Glühen**: Fügt einen Glüheffekt innerhalb des Objekts hinzu. Dieser Effekt kann durch Einstellen der *Spreizen* (Breite) des Glühens oder des *Verwischungsradius* angepasst werden. Letzterer gibt den Abstand vom Rand des Objekts an, in dem die Unschärfe auftreten soll. Zusätzlich gibt es die Möglichkeit, die Farbe des Glühens mit *Einzelfarbe* oder einer *Farbverlauf* einzustellen.

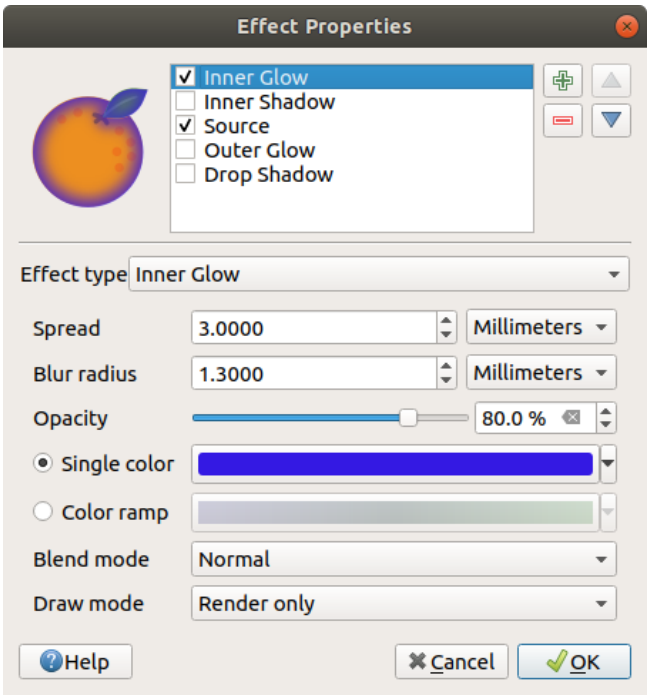


Abb. 12.22: Zeicheneffekte: Dialog Inneres Glühen

- **Außeres Glühen**: Dieser Effekt ähnelt dem Effekt *Inneres Glühen*, fügt aber den Glüheffekt an der Außenseite der Kanten des Objekts hinzu. Die verfügbaren Optionen zur Anpassung sind die gleichen wie beim Effekt *Inneres Glühen*.

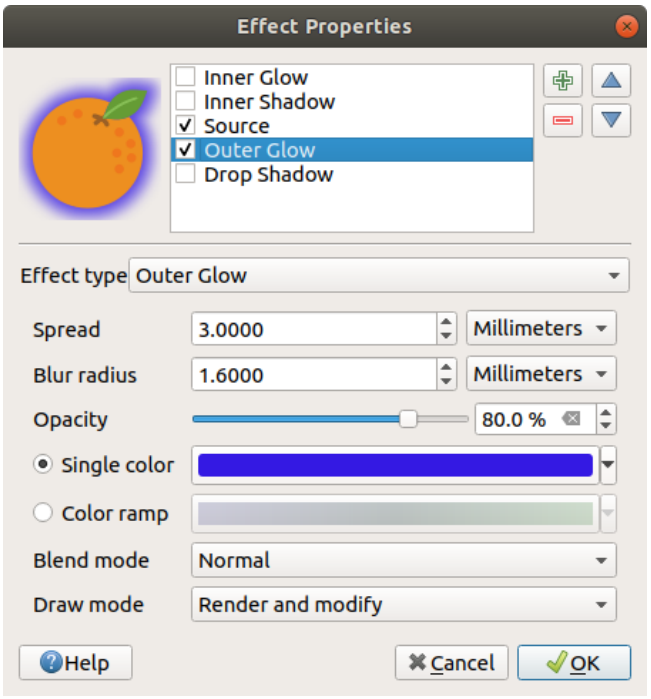


Abb. 12.23: Zeicheneffekte: Dialog Äußeres Glühen

- **Umformen**: Fügt die Möglichkeit hinzu, die Form des Symbols zu transformieren. Die ersten Optionen, die zur Anpassung zur Verfügung stehen, sind *Horizontal reflektieren* und *Vertikal reflektieren*, die tatsächlich eine Spiegelung an der horizontalen und/oder vertikalen Achse erzeugen. Die anderen Optionen sind:
  - *Scherung X,Y*: Neigt das Objekt entlang der X- und/oder Y-Achse.
  - *Skalierung X,Y*: Vergrößert oder verkleinert das Objekt entlang der X- und/oder Y-Achse um den angegebenen Prozentsatz.
  - *Drehung*: Dreht das Objekt um seinen Mittelpunkt.
  - und *Verschieben X,Y* ändert die Position des Elements anhand eines auf der X- und/oder Y-Achse angegebenen Abstands.

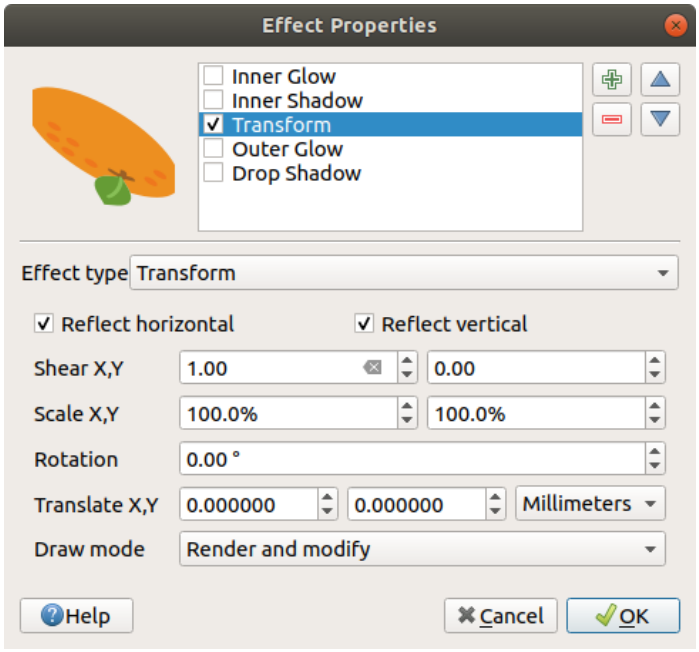


Abb. 12.24: Zeicheneffekte: Dialog transformieren

Es können ein oder mehrere Effekttypen gleichzeitig verwendet werden. Sie (de)aktivieren einen Effekt über das entsprechende Kontrollkästchen in der Effektliste. Sie können den ausgewählten Effekttyp mithilfe der Option *Effekttyp* ändern. Sie können die Effekte mit den Schaltflächen Verschiebe aufwärts und Verschiebe abwärts neu anordnen und mit den Schaltflächen Neuen Effekt hinzufügen und Effekt entfernen Effekte hinzufügen/entfernen.

Die Optionen *Deckkraft* und *Mischmodus* funktionieren ähnlich wie die in *Layerdarstellung* beschriebenen und können in allen Zeicheneffekten außer dem Umformen-Effekt verwendet werden.

Es gibt auch eine Option *Zeichenmodus* für jeden Effekt, und Sie können wählen, ob das Symbol gerendert und/oder verändert werden soll, indem Sie einige Regeln beachten:

- Die Effekte werden von oben nach unten gerendert.
- Der Modus *Nur Zeichnen* bedeutet, dass der Effekt sichtbar sein wird.
- Der Modus *Nur ändern* bedeutet, dass der Effekt nicht sichtbar ist, aber die Änderungen, die er anwendet, werden an den nächsten Effekt (den unmittelbar darunter liegenden) weitergegeben.
- Der Modus *Zeichnen und ändern* macht den Effekt sichtbar und gibt alle Änderungen an den nächsten Effekt weiter. Befindet sich der Effekt an der Spitze der Effektliste oder ist der unmittelbar darüber liegende Effekt nicht im Änderungsmodus, wird das ursprüngliche Quellensymbol aus den Layer-Eigenschaften verwendet (ähnlich wie bei Quelle).

12.1.4 Beschriftungen Eigenschaften

Die Eigenschaft *Beschriftungen* bietet Ihnen alle notwendigen und angemessenen Möglichkeiten, um intelligente Beschriftungen auf Vektor-Layern zu konfigurieren. Dieser Dialog kann auch über das Bedienfeld *Layergestaltung* oder über die Schaltfläche Layerbeschriftungseinstellungen in der **Beschriftungswerkzeugleiste** aufgerufen werden.

Am oberen Rand des Dialogfelds finden Sie:

- Eine Combobox zur Auswahl der gewünschten Beschriftungsmethode für den aktiven Layer
- Die Schaltfläche Projektbeschriftungsregeln einstellen: Hilft Ihnen, die Interaktionen zwischen Beschriftungen und Funktionen über den Layer im Projekt hinweg zu steuern. Weitere Details finden Sie unter *Projektbeschriftungsregeln einstellen*.

- Die Schaltfläche Automatische Platzierungseinstellungen (gilt für alle Layer): Konfigurieren Sie allgemeine Eigenschaften für die Platzierung von Beschriftungen und die Konfliktlösung. Weitere Details finden Sie unter *Einstellung des automatischen Platzierungssystems*.

Der erste Schritt besteht darin, die Beschriftungsmethode aus der Dropdown-Liste auszuwählen. Verfügbare Optionen sind:

- *Keine Beschriftung*: der Standardwert, der keine Beschriftungen auf dem Layer anzeigt
- *Einzelne Beschriftungen*: Zeigt Beschriftungen auf der Karte mit einem einzelnen Attribut oder einem Ausdruck an
- *Regelbasierte Beschriftung*
- and *Blockierend*: ermöglicht es, einen Layer nur als Hindernis für die Beschriftungen anderer Layer zu verwenden, ohne eigene Beschriftungen zu erzeugen.

Die nächsten Schritte gehen davon aus, dass Sie die Option *Einzelne Beschriftungen* wählen, wodurch sich der folgende Dialog öffnet.

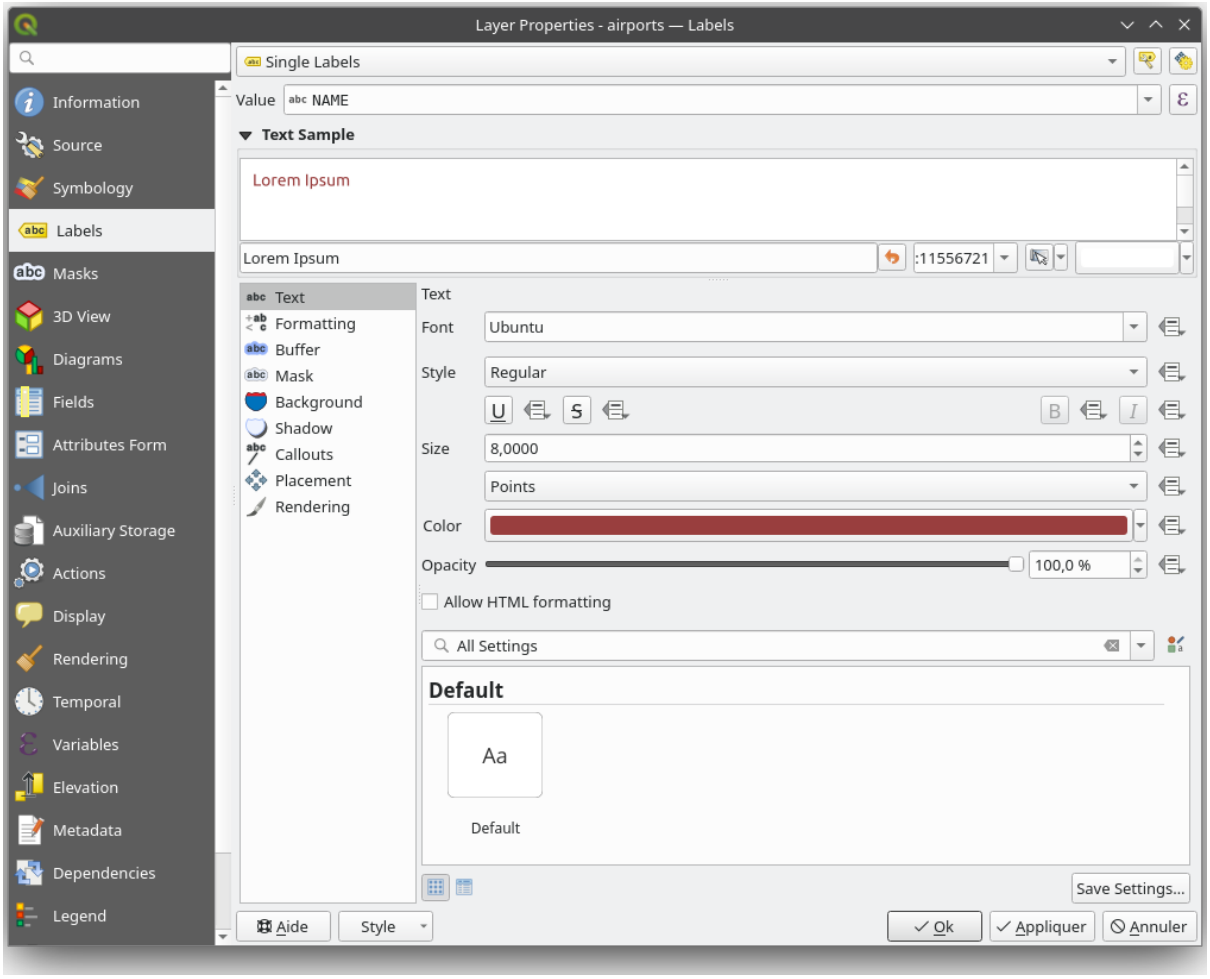


Abb. 12.25: Layer-Beschriftungseinstellungen - Einzelne Beschriftungen

Am oberen Rand des Dialogfelds ist eine Dropdown-Liste *Wert* vorhanden. Sie können eine Attributspalte auswählen, die für die Beschriftung verwendet werden soll. Standardmäßig wird der *Anzeigename* verwendet. Klicken Sie auf **|ausdruck|**, wenn Sie Beschriftungen auf der Grundlage von Ausdrücken definieren möchten - siehe *Ausdrucksbasierte Beschriftungen definieren*.

Bemerkung

Beschriftungen mit ihrer Formatierung können als Einträge in den Legenden angezeigt werden, wenn sie in der Registerkarte *Legende* aktiviert sind.

Im Folgenden werden unter verschiedenen Registerkarten Optionen zur Anpassung der Etiketten angezeigt:

- *Text*
- *Formatierung*
- *Puffer*
- *Maske*
- *Hintergrund*
- *Schatten*
- *Führungslinien*
- *Platzierung*
- *Darstellung*

Eine Beschreibung, wie die einzelnen Eigenschaften einzustellen sind, findet sich unter *Einrichten einer Beschriftung*.

Projektbeschriftungsregeln einstellen

Durch Klicken auf die Schaltfläche Projektbeschriftungsregeln einstellen neben dem Dropdown-Menü für die Beschriftungsmethode können Sie Regeln festlegen, die steuern, wie Beschriftungen aus einem Layer mit Beschriftungen oder Elementen aus einem anderen Layer zusammenspielen.

1. Drücken Sie die Schaltfläche Regel hinzufügen und wählen Sie im Dropdown-Menü einen der Regeltypen aus:
   - *Objektüberlappende Beschriftungen vermeiden*: Verhindert, dass Beschriftungen so platziert werden, dass sie Elemente aus einer anderen Ebene überlappen.
   - *Beschriftungen an Objekten heranziehen*: Verhindert, dass Beschriftungen zu weit von Objekten eines anderen Layers entfernt platziert werden. Der maximale Abstand kann in einer Einheit Ihrer Wahl festgelegt werden.
   - *Beschriftungen von Objekten fortschieben*: Verhindert, dass Beschriftungen zu nahe an Objekten aus einem anderen Layer platziert werden. Der Mindestabstand kann in einer Einheit Ihrer Wahl festgelegt werden, ebenso wie die *Priorität der Regel* (Regeln mit der höchsten Priorität sind bei Konflikten bei der Beschriftungsplatzierung wichtiger).
   - *Beschriftungen von anderen Beschriftungen fortschieben*: Verhindert, dass Beschriftungen zu nahe an Beschriftungen aus einer anderen Ebene platziert werden.

   Achtung

   The last three options are only available on QGIS installed with GEOS >= 3.10 (see *Help* ► *About* menu).

2. Füllen Sie die Eigenschaften nach Belieben aus; Sie können der Regel einen aussagekräftigeren Namen geben.
3. Drücke Sie *OK*.
4. Fügen Sie so viele Regeln hinzu, wie nötig.

5. Drücken Sie bei Bedarf Regel bearbeiten, um die ausgewählte Regel zu ändern, oder Regel entfernen, um sie aus dem Projekt zu löschen.

Die festgelegten Regeln sind in jedem Layer im Projekt unter der Registerkarte *Beschriftungen* in den Layereigenschaften durch Klicken auf die Schaltfläche Projektbeschriftungsregeln einstellen verfügbar. Sie können jede Regel vorübergehend aktivieren oder deaktivieren, indem Sie das Kontrollkästchen neben dem Namen aktivieren bzw. deaktivieren. Bewegen Sie den Mauszeiger über eine Regel, um eine Vorschau ihrer Details anzuzeigen.

Einstellung des automatischen Platzierungssystems

Sie können die Einstellungen für die automatische Platzierung verwenden, um ein automatisches Verhalten der Beschriftungen auf Projektebene zu konfigurieren. Klicken Sie in der rechten oberen Ecke der Registerkarte *Beschriftungen* auf die Schaltfläche Automatische Platzierungseinstellungen (gilt für alle Layer), um einen Dialog mit den folgenden Optionen zu öffnen:

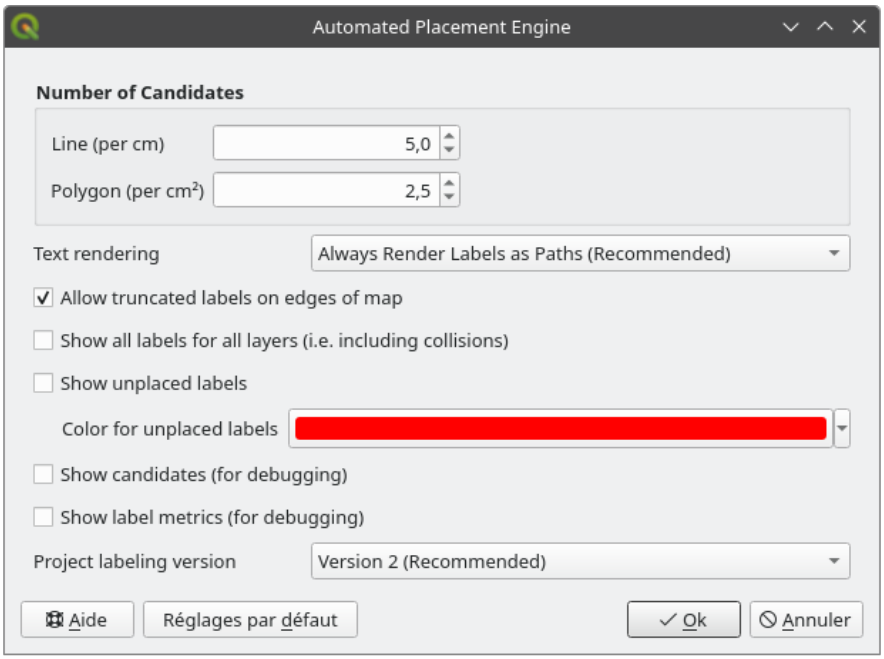


Abb. 12.26: Der Dialog für automatische Platzierungen

- *Kandidatenanzahl*: berechnet und weist Linien- und Polygonobjekten die Anzahl der möglichen Beschriftungen zu, die auf ihrer Größe basieren. Je länger oder breiter ein Objekt ist, desto mehr Kandidaten hat es, und seine Beschriftungen können mit geringerem Kollisionsrisiko besser platziert werden.
- *Textdarstellung*: Legt den Standardwert für die Darstellung von Beschriftungs-Widgets fest, wenn *die Kartenansicht* oder *ein Druck-Layout* als PDF oder SVG exportiert wird. Wenn *Beschriftungen immer als Text darstellen* ausgewählt ist, können Beschriftungen in externen Anwendungen (z. B. Inkscape) wie normaler Text bearbeitet werden. ABER dies hat den Nebeneffekt, dass die Darstellungsqualität beeinträchtigt wird und es zu Problemen bei der Darstellung kommt, wenn bestimmte Texteinstellungen wie Puffer vorhanden sind. Aus diesem Grund wird die Option *Beschriftungen immer als Pfad darstellen* empfohlen, die Labels als Konturen exportiert, aber vollständige Kompatibilität mit allen verfügbaren Formatierungsoptionen garantiert. Mit der Option *Anzeige der Schriftungen als Text bevorzugen* werden Labels als Textobjekte gerendert, es sei denn, dies führt zu Rendering-Artefakten oder einer schlechten Rendering-Qualität (abhängig von den Textformateinstellungen).

Bemerkung

Beim Rendern von Beschriftungen als Text auf einem vektorbasierten Format (z. B. PDF oder SVG) muss darauf geachtet werden, dass die erforderlichen Schriftarten beim Öffnen der erstellten Dateien für die Benutzer verfügbar sind, da sonst stattdessen Standard-Fallback-Schriftarten zur Anzeige der Ausgabe verwendet werden. (Allerdings KÖNNEN PDF-Exporte je nach Plattform des Benutzers einige Schriftarten automatisch einbetten, sofern dies möglich ist).

- *Abgeschnittene Beschriftungen am Kartenrand erlauben*: steuert, ob Beschriftungen, die teilweise außerhalb der Kartenausdehnung liegen, dargestellt werden sollen. Wenn diese Option aktiviert ist, werden diese Beschriftungen angezeigt (wenn es keine Möglichkeit gibt, sie vollständig innerhalb des sichtbaren Bereichs zu platzieren). Wenn sie nicht markiert ist, werden teilweise sichtbare Beschriftungen übersprungen. Beachten Sie, dass diese Einstellung keine Auswirkungen auf die Darstellung der Beschriftungen in einem *Druck-Layout* hat.
- *Alle Beschriftungen aller Layer anzeigen (d.h. einschließlich Kollisionen)*. Beachten Sie, dass diese Option auch pro Layer gesetzt werden kann (siehe *Darstellungs-Reiter*)
- :guilabel:` Nicht plazierbare Beschriftungen anzeigen`: ermöglicht es, festzustellen, ob wichtige Beschriftungen in den Maps fehlen (z.B. aufgrund von Überlappungen oder anderen Einschränkungen). Sie werden in einer einstellbaren Farbe angezeigt.
- *Kandidaten anzeigen (zur Fehlersuche)*: steuert, ob Kästchen auf der Karte gezeichnet werden sollen, die alle für die Platzierung der Beschriftung generierten Kandidaten anzeigen. Wie die Bezeichnung schon sagt, ist dies nur zur Fehlersuche und zum Testen der Auswirkungen verschiedener Beschriftungseinstellungen nützlich. Dies könnte für eine bessere manuelle Platzierung mit Werkzeugen aus der *Etiketten-Symbolleiste* nützlich sein.
- *Beschriftungseigenschaften anzeigen (zur Fehlersuche)*: Zeigt die Textgrenzen der Beschriftung in Rot und die Grundlinien in Blau an.
- *Projektbeschriftungsversion*: QGIS unterstützt zwei verschiedene Versionen der automatischen Platzierung von Labels:
  - *Version 1*: das alte System (verwendet von QGIS-Versionen 3.10 und früher, und beim Öffnen von Projekten, die in diesen Versionen erstellt wurden, in QGIS 3.12 oder später). In Version 1 werden Beschriftungs- und Hindernisprioritäten nur als „grobe Richtwerte“ behandelt, und es ist möglich, dass in dieser Version eine Beschriftung mit niedriger Priorität über ein Hindernis mit hoher Priorität gelegt wird. Dementsprechend kann es schwierig sein, die gewünschten Beschriftungsergebnisse zu erzielen, wenn diese Version verwendet wird, und sie wird daher nur für die Kompatibilität mit älteren Projekten empfohlen.
  - *Version 2 (empfohlen)*: dies ist das Standardsystem in neuen Projekten, die in QGIS 3.12 oder später erstellt wurden. In Version 2 wurde die Logik, die vorgibt, wann Beschriftungen *Hindernisse* überlappen dürfen, überarbeitet. Die neue Logik verbietet es, dass sich Labels mit Hindernissen überlappen, die eine höhere Gewichtung als die Priorität des Labels haben. Als Ergebnis führt diese Version zu viel vorhersehbareren und leichter verständlichen Beschriftungsergebnissen.

Regelbasierte Beschriftung

Bei der regelbasierten Beschriftung können mehrere Beschriftungskonfigurationen definiert und selektiv auf der Grundlage von Ausdrucksfiltern und Skalenbereichen angewendet werden, wie bei der *Regelbasierten Darstellung* von Symbolen.

So erstellen Sie eine Regel:

1. Wählen Sie die Option **Regelbasierte Beschriftung** in der Haupt-Dropdown-Liste auf der Registerkarte *Beschriftungen*
2. Klicken Sie auf die Schaltfläche Regel hinzufügen am unteren Rand des Dialogs.
3. Füllen Sie den neuen Dialog mit:

- *Beschreibung*: ein Text, der zur Identifizierung der Regel auf der Registerkarte *Beschriftungen* und als *Beschriftungseintrag* in der Druck-Layoutlegende verwendet wird.
- *Filter*: ein Ausdruck zur Auswahl der Objekte, auf die die Beschriftungseinstellungen angewendet werden sollen
- Wenn bereits Regeln festgelegt sind, kann die Option *Else* verwendet werden, um alle Objekte auszuwählen, die keinem Filter der Regeln in derselben Gruppe entsprechen.

4. Sie können einen *Maßstabsbereich* festlegen, in dem die Etikettenregel angewendet werden soll.
5. Die im Bereich *Beschriftungen* verfügbaren Optionen sind die üblichen *Beschriftungseinstellungen*. Konfigurieren Sie sie und drücken Sie *OK*.

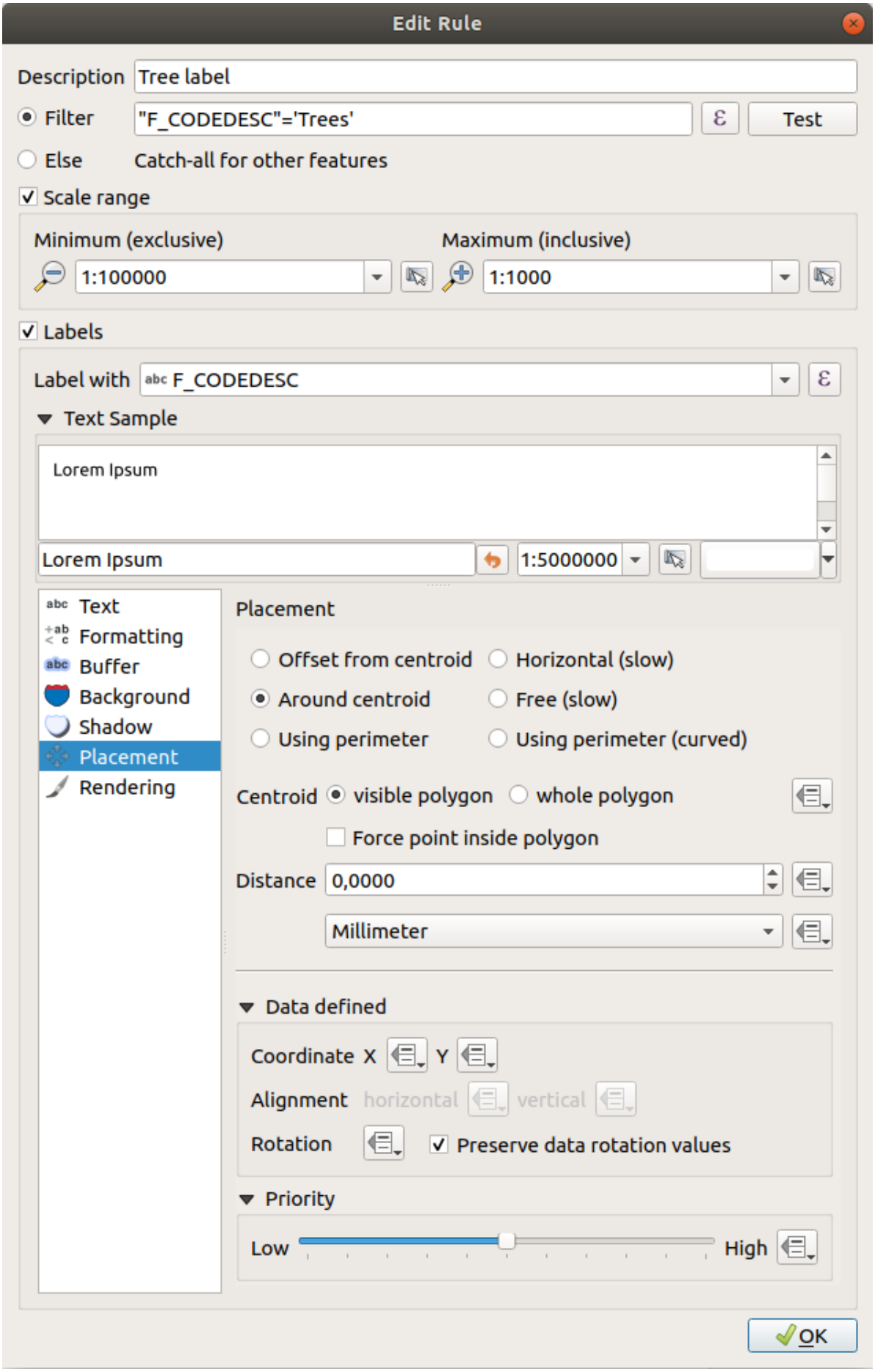


Abb. 12.27: Regeleigenschaften

Eine Zusammenfassung der vorhandenen Regeln wird im Hauptdialog angezeigt (siehe Abb. 12.28). Sie können mehrere Regeln hinzufügen, neu anordnen oder per Drag-and-Drop verschmelzen. Sie können sie auch mit der Schaltfläche entfernen oder mit der Schaltfläche oder einem Doppelklick bearbeiten.

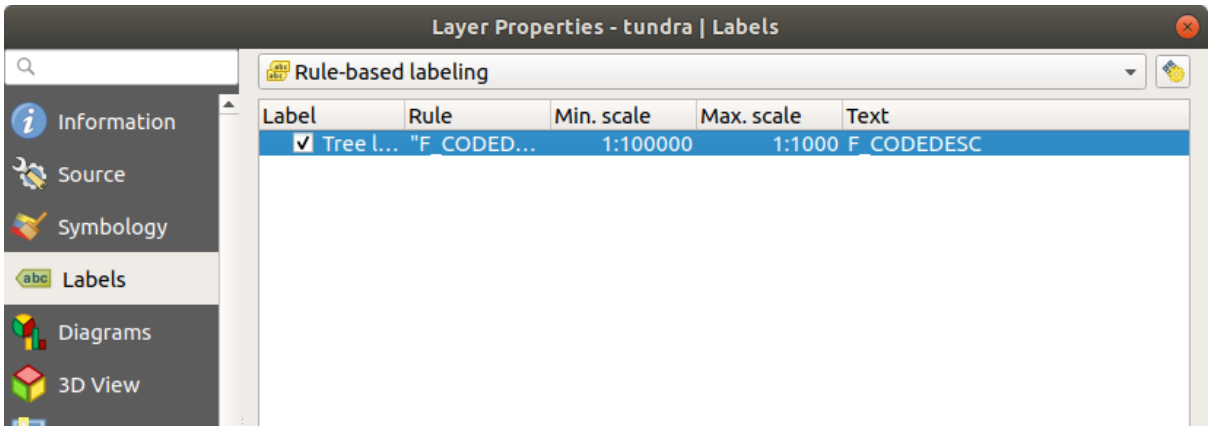


Abb. 12.28: Regelbasierte Beschriftung Bedienfelder

Ausdrucksbasierte Beschriftungen definieren

Unabhängig davon, ob Sie einen einzelnen oder einen regelbasierten Beschriftungstyp wählen, erlaubt QGIS die Verwendung von Ausdrücken zur Beschriftung von Objekten.

Angenommen, Sie verwenden die Methode *Einzelne Beschriftung*, dann klicken Sie auf die Schaltfläche **|Ausdruck|** neben der Dropdown-Liste *Wert* in der Registerkarte *Beschriftungen* des Layereigenschaften-Dialogs.

In Abb. 12.29 sehen Sie einen Beispielausdruck, um den Layer Alaska-Bäume mit Baumart und Fläche zu beschriften, basierend auf dem Feld ‚VEGDESC', etwas beschreibendem Text und der Funktion `$area` in Kombination mit `format_number()`, um den Zahlenwert zu verdeutlichen.

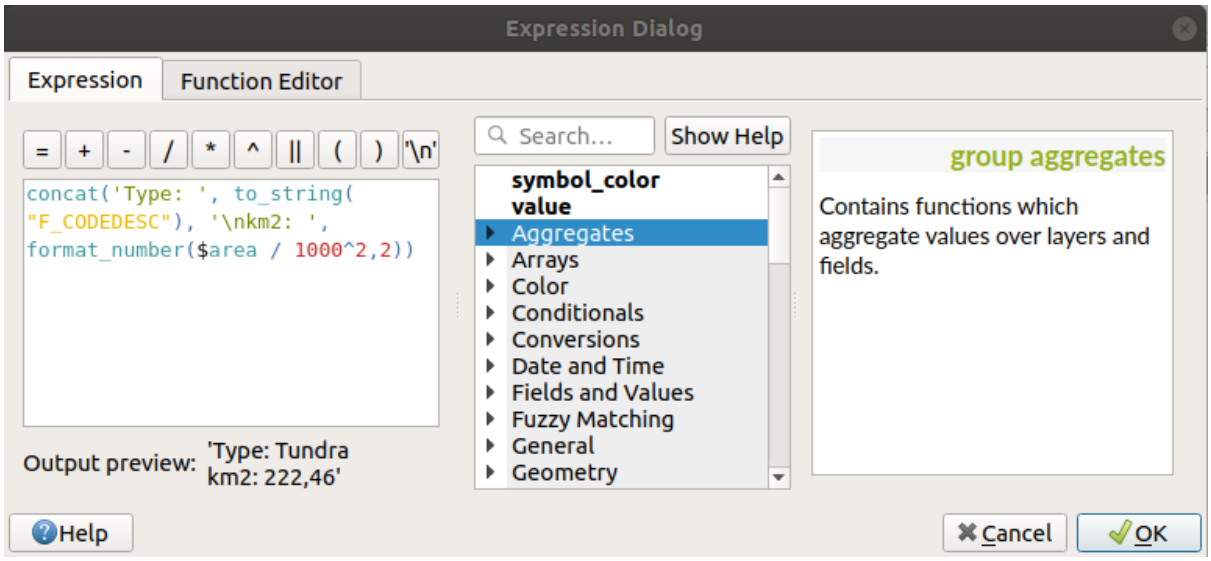


Abb. 12.29: Ausdrücke für das Beschriften verwenden

Die ausdrucksbasierte Beschriftung ist einfach zu handhaben. Alles, worauf Sie achten müssen, ist folgendes:

- Möglicherweise müssen Sie alle Elemente (Strings, Felder und Funktionen) mit einer String-Verkettungsfunktion wie `concat`, `+` oder `||` kombinieren. Seien Sie sich bewusst, dass in manchen

Situationen (wenn Null oder numerische Werte involviert sind) nicht alle dieser Werkzeuge Ihren Bedürfnissen entsprechen werden.

- Zeichenketten werden in ‚einfache Anführungszeichen' geschrieben.
- Felder werden in „doppelten Anführungszeichen" oder ohne Anführungszeichen geschrieben.

Schauen wir uns einige Beispiele an:

1. Etikett basierend auf zwei Feldern ‚name' und ‚place' mit einem Komma als Trennzeichen:

```
"name" || ', ' || "place"
```

Liefert:

```
John Smith, Paris
```

2. Beschriftung basierend auf zwei Feldern ‚name' und ‚place' mit anderen Texten:

```
'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")
```

Liefert:

```
My name is John Smith and I live in Paris
```

3. Beschriftung auf der Grundlage von zwei Feldern ‚name' und ‚place' mit anderen Texten, die verschiedene Verkettungsfunktionen kombinieren:

```
concat('My name is ', name, ' and I live in ' || place)
```

Liefert:

```
My name is John Smith and I live in Paris
```

Oder, wenn das Feld ‚place' NULL ist, wird zurückgegeben:

```
My name is John Smith
```

4. Mehrzeilige Beschriftung basierend auf zwei Feldern ‚name' und ‚place' mit einem beschreibenden Text:

```
concat('My name is ', "name", '\n' , 'I live in ' , "place")
```

Liefert:

```
My name is John Smith
I live in Paris
```

5. Etikett, das auf einem Feld und der Funktion $area basiert, um den Namen des Ortes und seine gerundete Flächengröße in einer umgerechneten Einheit anzuzeigen:

```
'The area of ' || "place" || ' has a size of '
|| round($area/10000) || ' ha'
```

Liefert:

```
The area of Paris has a size of 10500 ha
```

6. Erstellen Sie eine CASE ELSE-Bedingung. Wenn der Bevölkerungswert im Feld *population* <= 50000 ist, handelt es sich um einen Ort, andernfalls um eine Stadt:

```
concat('This place is a ',
CASE WHEN "population" <= 50000 THEN 'town' ELSE 'city' END)
```

Liefert:

```
This place is a town
```

7. Anzeigename für die Städte und keine Beschriftung für die anderen Objekte (für den Kontext „city“, siehe Beispiel oben):

```
CASE WHEN "population" > 50000 THEN "NAME" END
```

Liefert:

```
Paris
```

Wie Sie im Ausdruckeditor sehen können stehen Ihnen hunderte von Funktionen zur Verfügung um einfache und sehr komplexe Ausdrücke zumBeschriften Ihrer Daten in QGIS zu erstellen. Siehe das *Ausdrücke* Kapitel für weitere Informationen und ein Beispiel zu Ausdrücken.

Datendefinierte Übersteuerung für das Beschriften

Mit der Funktion Datendefinierte Übersteuerung werden die Einstellungen für die Beschriftung durch Einträge in der Attributtabelle oder darauf basierende Ausdrücke überschrieben. Mit diesem Objekt lassen sich Werte für die meisten der oben beschriebenen Beschriftungsoptionen setzen.

Nehmen wir zum Beispiel den QGIS-Beispieldatensatz Alaska und beschriften den Layer `airports` mit dem Namen, basierend auf ihrem militärischen `USE`, d.h. ob der Flughafen für:

- Militär, dann in grauer Farbe, Größe 8 anzeigen;
- andere, dann in blauer Farbe, Größe 10 anzeigen.

Um dies zu tun, müssen Sie die Beschriftung im Feld `NAME` des Layers aktivieren (siehe *Einrichten einer Beschriftung*):

1. Aktivieren Sie die Registerkarte *Text*.
2. Klicken Sie auf das Symbol neben der Eigenschaft *Größe*.
3. Wählen Sie *Bearbeiten…* und geben Sie ein:

```
CASE
  WHEN "USE" like '%Military%' THEN 8 -- because compatible values are
↪'Military'
                                    -- and 'Joint Military/Civilian'
  ELSE 10
END
```

4. Drücken Sie *OK*, um zu bestätigen. Der Dialog wird geschlossen und die Schaltfläche wird zu , was bedeutet, dass eine Regel ausgeführt wird.
5. Klicken Sie dann auf die Schaltfläche neben der Eigenschaft Farbe, geben Sie den folgenden Ausdruck ein und bestätigen Sie:

```
CASE
  WHEN "USE" like '%Military%' THEN '150, 150, 150'
  ELSE '0, 0, 255'
END
```

Ebenso können Sie jede andere Eigenschaft des Etiketts so anpassen, wie Sie es wünschen. Weitere Details zur Beschreibung und Handhabung des Widgets Datendefinierte Übersteuerung finden Sie im Abschnitt *Einrichten von datendefinierter Übersteuerung*.

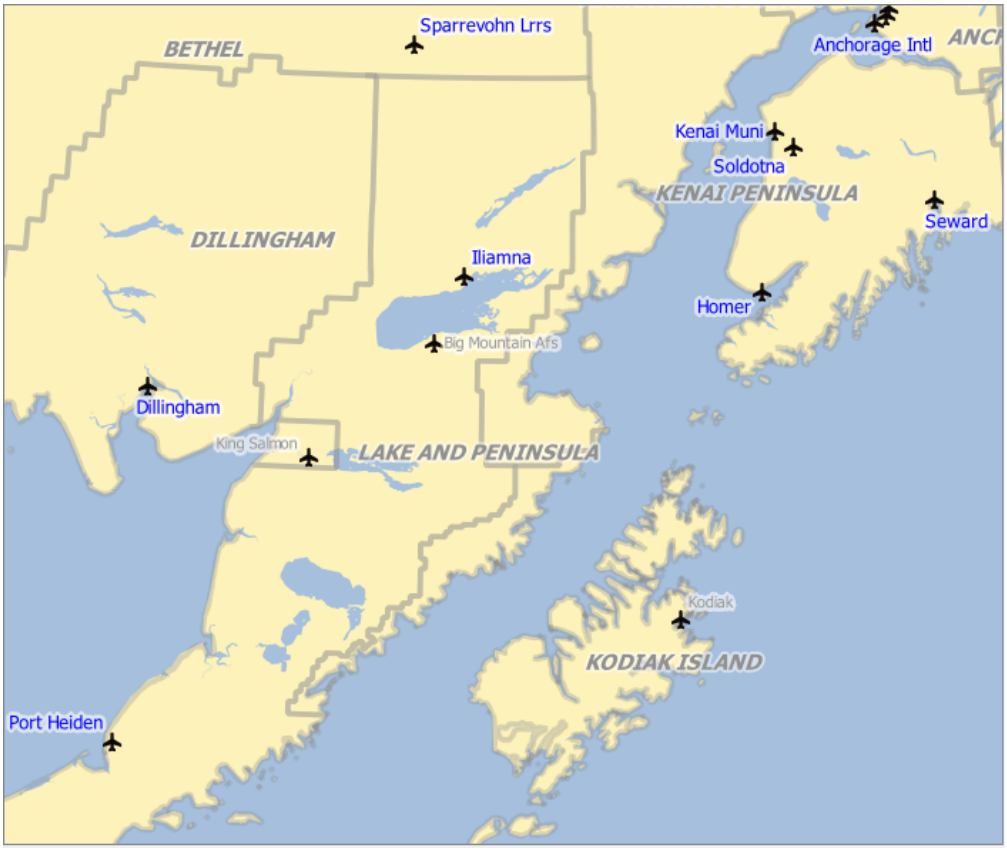


Abb. 12.30: Die Beschriftungen der Flughäfen werden auf der Grundlage ihrer Attribute formatiert

Tipp

Verwenden Sie die datendefinierte Übersteuerung, um jedes Teil von mehrteiligen Objekten zu beschriften.

Es gibt eine Option, um die Beschriftung für mehrteilige Objekte unabhängig von Ihren Beschriftungseigenschaften festzulegen. Wählen Sie den Menüpunkt *Darstellung*, `Objektoptionen`, gehen Sie auf die Schaltfläche Datendefinierte Übersteuerung neben der Checkbox *Alle Teile mehrteiliger Objekte beschriften* und definieren Sie die Labels wie in *Einrichten von datendefinierter Übersteuerung* beschrieben.

Die Beschriftungswerkzeugleiste

Die *Beschriftungswerkzeugleiste* bietet einige Möglichkeiten zur Manipulation von *Beschriftungen* (einschließlich deren *Führungslinien*) und *Diagrammen*. Wenn Sie Beschriftungen oder Diagramme manuell angepassen, dann werden diese Informationen im *Hilfsspeicher* abgelegt. Dazu wird bei der ersten Verwendung eines Werkzeugs automatisch eine Verknüpfung mit dem Hilfsspeicher hergestellt. Der Layer muss dazu über ein eindeutiges Attribut verfügen (z.B. ID). Weitere Informationen siehe *Hilfsspeicher*.

Abb. 12.31: Die Beschriftungswerkzeugleiste

- Gepinnte Beschriftungen, Diagramme und Callouts hervorheben : Wenn der Vektorlayer auf bearbeitbar gestellt ist, sind die

Hervorhebung grün, sonst blau.

- Schaltet die Anzeige von nicht platzierten Beschriftungen um : Ermöglicht es, festzustellen, ob wichtige Beschriftungen in den Karten fehlen (z.B. aufgrund von Überlappungen oder anderen Einschränkungen). Sie werden mit einer anpassbaren Farbe angezeigt (siehe *Einstellung des automatischen Platzierungssystems*) (ACHTUNG: Objekte, die über die Symbolisierung nicht dargestellt werden, erhalten auch keine Beschriftung!).

- Beschriftungen und Diagramme anpinnen/lösen : Wenn Sie eine Beschriftung oder ein Diagramm anklicken oder über einen Bereich ziehen, werden die entsprechenden Elemente angeheftet. Wenn Sie dabei die `Umschalt`-Taste gedrückt halten, wird die Anheftung der Elemente aufgehoben, wenn Sie die `Strg`-Taste gedrückt halten, können Sie den Anheftungsstatus umzuschalten (ACHTUNG: es kann vorkommen, dass dieses Werkzeug nicht korrekt funktioniert, wenn noch keine Verknüpfung zum *Hilfsspeicher* hergestellt wurde. Verwenden Sie dann zunächst ein anderes Werkzeug aus der Beschriftungswerkzeugleiste, um die Verknüpfung herzustellen).

- Beschriftungen und Diagramme anzeigen/ausblenden : Wenn Sie auf die Elemente klicken oder einen Bereich mit `Umschalt` anklicken und ziehen, werden sie ausgeblendet. Wenn eine Beschriftung oder ein Diagramm ausgeblendet ist, müssen Sie nur auf das entsprechende Vektor-Objekt klicken, um Beschriftung der Diagramm wieder sichtbar zu machen. Wenn Sie einen Bereich ziehen, werden alle Elemente in diesem Bereich wiederhergestellt.

- Beschriftung, Diagramm oder Callout verschieben : Wenn Sie ein Element anklicken, bleibt es am Mauszeiger haften, Sie können es an die gewünschte Stelle verschieben und mit einem erneuten Klick dort platzieren. Die neuen Koordinaten werden automatisch in *Hilfsfeldern* gespeichert. Wenn Sie das Element mit diesem Werkzeug auswählen und die Taste `Entfernen` drücken, wird der gespeicherte Positionswert gelöscht.

- Eine Beschriftung drehen : Klicken Sie, um ein Element auszuwählen, und klicken Sie erneut, um die gewünschte Drehung anzuwenden. Der neue Winkel wird ebenfalls in einem Hilfsfeld gespeichert. Wenn Sie ein Etikett mit diesem Werkzeug auswählen und die Taste `Entfernen` drücken, wird der Drehwert dieses Etiketts gelöscht.

- Beschriftungseigenschaften ändern : Hiermit können Sie einen Dialog öffnen, um die Eigenschaften der angeklickten Beschriftung zu ändern; dies kann die Beschriftung selbst, ihre Koordinaten, Winkel, Schriftart, Größe, mehrzeilige Ausrichtung … sein, sofern diese Eigenschaft einem Feld zugeordnet wurde (ACHTUNG: s. Warnung unten). Hier können Sie auch die Option *Jeden Teil eines Objekts beschriften* auswählen.

Warnung

Beschriftungswerkzeuge können die Werte in der Attributtabelle überschreiben!

Die Verwendung der *Beschriftungswerkzeugleiste* zur Anpassung der Beschriftung kann unter Umständen vorhandene Werte in der Attributtabelle überschreiben. Dies ist z. B. der Fall, wenn sie das Werkzeug „Beschriftungseigenschaften ändern“ verwenden, um den Text der Beschriftung anzupassen. Achten Sie daher darauf, dass Sie nicht versehentlich Daten ersetzen, die Sie später benötigen!

Bemerkung

Der Mechanismus *Hilfsspeicher* kann verwendet werden, um die Beschriftung (Position usw.) anzupassen, ohne die zugrunde liegende Datenquelle zu verändern.

Anpassen der Beschriftungen in der Kartenansicht

In Kombination mit der *Beschriftungswerkzeugleiste* hilft Ihnen die Datendefinierte Übersteuerung bei der Manipulation von Beschriftungen in der Kartenansicht (Verschieben, Bearbeiten, Drehen). Im Folgenden wird ein Beispiel beschrieben, bei dem die Funktion Beschriftung, Diagramm oder Callout verschieben verwendet wird (siehe Abb. 12.32).

1. Importieren Sie `lakes.shp` aus dem QGIS Beispieldatensatz.
2. Doppelklicken Sie den Layer um die Layereigenschaften zu öffnen. Klicken Sie auf *Beschriftungen* und *Platzierung*. Wählen Sie *Abstand vom Punkt*.
3. Suchen Sie nach den Einträgen *Datendefiniert*. Klicken Sie auf das Symbol , um den Feldtyp für das Feld *Koordinate* zu definieren. Wählen Sie `xlabel` für X und `ylabel` für Y. Die Icons sind nun gelb hervorgehoben.

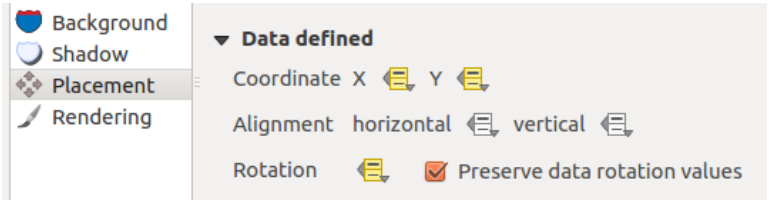


Abb. 12.32: Das Beschriften von Polygonlayern mit datendefinierter Übersteuerung

4. Zoomen Sie auf einen See.
5. Mit der Schaltfläche Bearbeitungsstatus umschalten den Layer editierbar machen.
6. Klicken Sie in der Beschriftungswerkzeugleiste auf das Symbol . Nun können Sie das Label manuell an eine andere Position verschieben (siehe Abb. 12.33). Die neue Position des Labels wird in den Spalten `xlabel` und `ylabel` der Attributtabelle gespeichert.
7. Es ist auch möglich, eine Linie hinzuzufügen, die jeden See mit seinem verschobenen Etikett verbindet, über:
   - die *Callout Eigenschaften* der Beschriftung
   - oder den *Geometriegenerator in einem Symbol* mit dem folgenden Ausdruck:

```
make_line( centroid( $geometry ), make_point( "xlabel", "ylabel" ) )
```

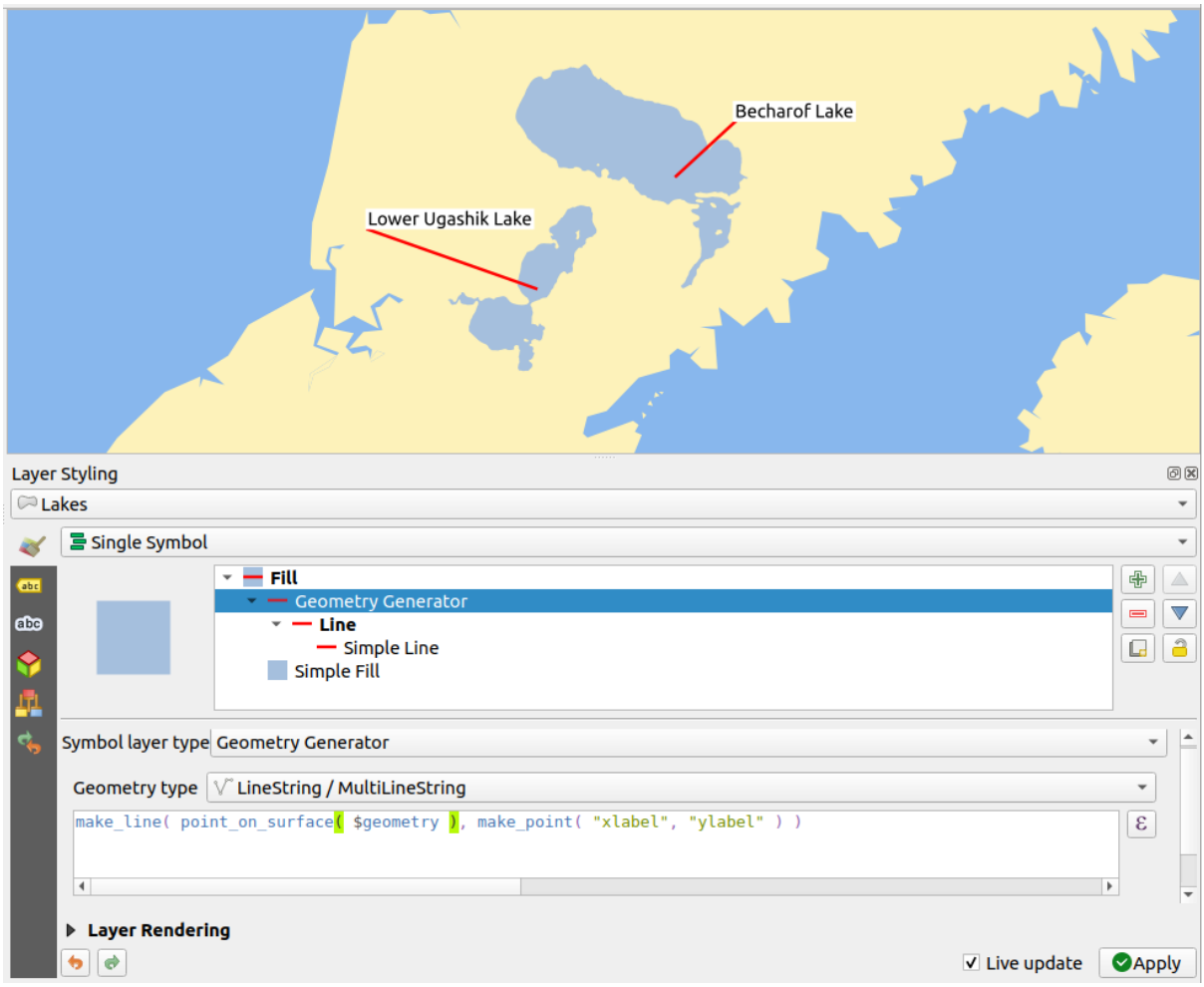


Abb. 12.33: Verschobene Beschriftungen

Bemerkung

Der Mechanismus *Hilfsspeicher* kann mit daten-definierten Eigenschaften verwendet werden, ohne dass eine editierbare Datenquelle vorhanden ist.

12.1.5 Diagrammeigenschaften

Über die Registerkarte *Diagramme* können Sie einem Vektorlayer eine grafische Überlagerung hinzufügen (siehe Abb. 12.34). Dieser Dialog kann auch über das Bedienfeld *Layergestaltung* oder über die Schaltfläche Layerdiagrammoptionen der **Beschriftungen**-Werkzeugleiste aufgerufen werden.

Die aktuelle Kernimplementation von Diagrammen bietet Unterstützung von:

- *Keine Diagramme*: der Standardwert, bei dem kein Diagramm über den Objekten angezeigt wird;
- *Kuchendiagramm*, eine kreisförmige statistische Grafik, die zur Veranschaulichung numerischer Proportionen in Stücke unterteilt ist. Die Bogenlänge jeder Scheibe ist proportional zu der Menge, die sie darstellt;
- abc *Textdiagramm*, ein horizontal geteilter Kreis, der statistische Werte enthält;

- |Histogramm| *Histogramm*, Balken in unterschiedlichen Farben für jedes Attribut, die nebeneinander angeordnet sind;
- *Balkenstapel*, stapelt Balken unterschiedlicher Farben für jedes Attribut vertikal oder horizontal übereinander;
- *Gestapeltes Diagramm*, stapelt Diagramme gleichen oder unterschiedlichen Typs nebeneinander, vertikal oder horizontal. Weitere Details unter *Gestapelte Diagramme*.

In der oberen rechten Ecke der Registerkarte *Diagramme* bietet die Schaltfläche

die Möglichkeit, die Platzierung von Diagrammen *Beschriftungen* auf der Kartenansicht zu steuern.

> **Tipp**
>
> **Schnelles Umschalten zwischen verschiedenen Diagrammtypen**
>
> Da die Einstellungen für die verschiedenen Diagrammtypen fast identisch sind, können Sie bei der Erstellung Ihres Diagramms einfach den Diagrammtyp ändern und prüfen, welcher Diagrammtyp für Ihre Daten am besten geeignet ist, ohne dass Verluste entstehen.

Für jeden Diagrammtyp sind die Eigenschaften in mehrere Registerkarten unterteilt:

- *Attribute*
- *Darstellung*
- *Größe*
- *Platzierung*
- *Optionen*
- *Legende*

Attribute

Attribute`definiert, welche Variablen in dem Diagramm gezeigt werden. Nutzen Sie den :sup:`Objekt hinzufügen Knopf um das gewünschte Feld in das ‚Zugeordnete Attribut' Bedienfeld zu schieben. Erzeugte Attribute mit den *Ausdrücke* können auch genutzt werden.

Sie können jede Zeile durch Klicken und Ziehen nach oben oder unten verschieben und so die Anzeige der Attribute sortieren. Sie können auch die Beschriftung in der Spalte „Legende" oder die Farbe des Attributs ändern, indem Sie auf das Element doppelklicken.

Diese Bezeichnung ist der Standardtext, der in der Legende des Drucklayouts oder des Layer-Baums angezeigt wird.

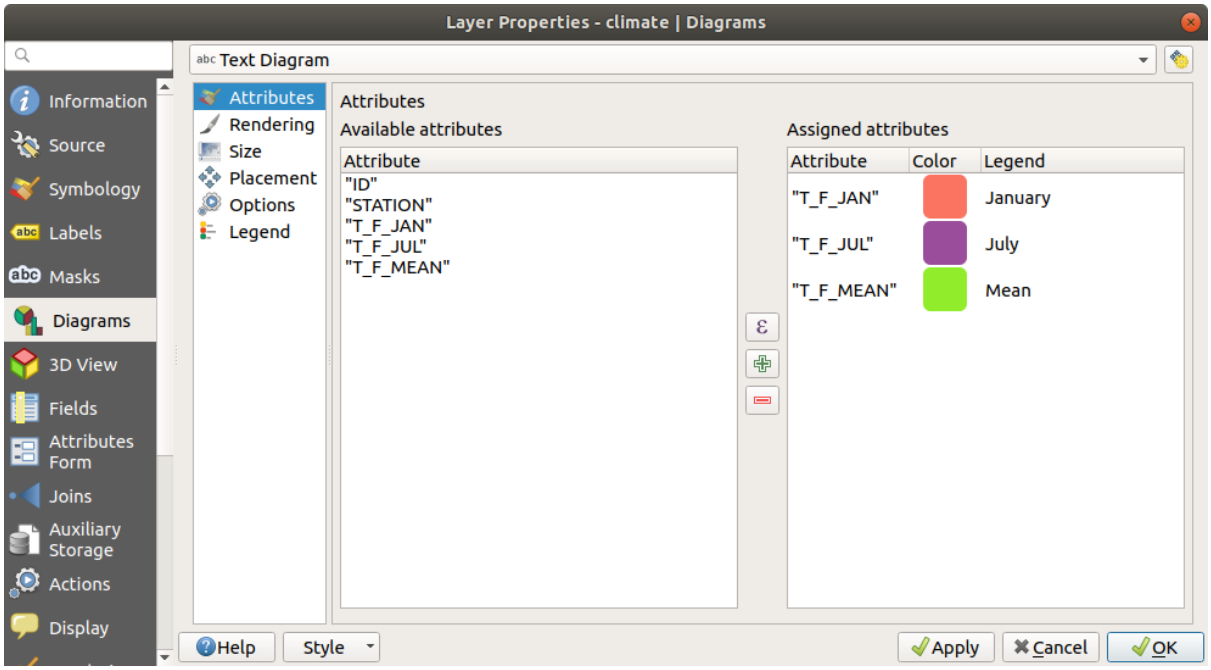


Abb. 12.34: Diagrammeigenschaften - Attribute-Reiter

Darstellung

Darstellung definiert, wie das Diagramm aussieht. Es bietet allgemeine Einstellungen, die nicht mit den statistischen Werten interferieren, wie z.B.:

- die Deckkraft, die Breite und die Farbe des Umrisses der Grafik;
- abhängig von der Art des Diagramms:
  - für Histogramm und gestapelte Balken die Breite des Balkens und den Abstand zwischen den Balken. Bei gestapelten Balken können Sie den Abstand auf `0` setzen. Außerdem kann das *Achsenliniensymbol* in der Kartenansicht sichtbar gemacht und mit *Liniensymboleigenschaften* angepasst werden.
  - für Textdiagramme, die Hintergrundfarbe des Kreises und die für Texte verwendete *Schriftart*;
  - bei Kreisdiagrammen den *Startwinkel* der ersten Scheibe und ihre *Richtung* (im Uhrzeigersinn oder nicht).
- die Verwendung von :ref:Zeicheneffekten <draw_effects> auf die Grafik.

Auf dieser Registerkarte können Sie auch die Sichtbarkeit des Diagramms mit verschiedenen Optionen verwalten und feineinstellen:

- *Diagramm-Z-Index*: Steuert, wie Diagramme übereinander und über Beschriftungen gezeichnet werden. Ein Diagramm mit einem hohen Index wird über anderen Diagrammen und Beschriftungen gezeichnet.
- *Alle Diagramme anzeigen*: zeigt alle Diagramme an, auch wenn sie sich überschneiden;
- *Diagramm anzeigen*: erlaubt es, nur bestimmte Diagramme darzustellen;
- *Immer anzeigen*: wählt bestimmte Diagramme aus, die immer angezeigt werden, auch wenn sie andere Diagramme oder Kartenbeschriftungen überlappen;
- Einstellen der *Maßstabsabhängigen Sichtbarkeit*;

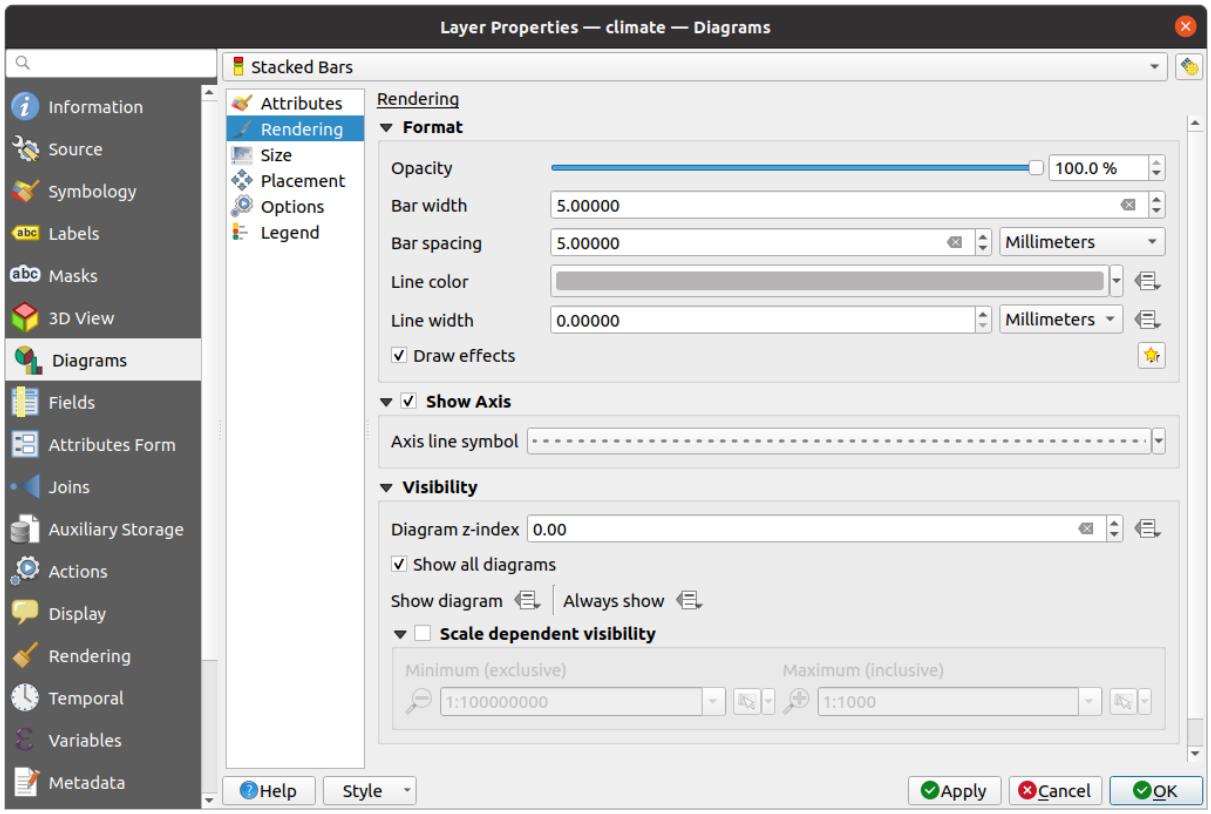


Abb. 12.35: Diagrammeigenschaften - Darstellung-Reiter

Größe

Größe ist die wichtigste Registerkarte, auf der Sie festlegen, wie die ausgewählten Statistiken abgebildet werden. Die Diagrammgröße *Einheit* kann ‚Millimeter', ‚Punkte', ‚Pixel', ‚Karteneinheiten' oder ‚Zoll' sein. Sie können verwenden:

- *Feste Größe*, eine eindeutige Größe zur Darstellung der Grafik aller Objekte (nicht verfügbar für Histogramme)
- oder *Skalierte Größe*, auf der Grundlage eines Ausdrucks mit Layer-Attributen:

  1. In *Attribut* ein Feld auswählen oder einen Ausdruck erstellen
  2. Drücken Sie *Suchen*, um den *Maximalwert* des Attributs zu ermitteln oder geben Sie einen eigenen Wert in das Widget ein.
  3. Für Histogramm und gestapelte Balken geben Sie einen *Balkenlänge*-Wert ein, der den *Maximalwert* der Attribute darstellt. Für jedes Objekt wird die Balkenlänge dann linear skaliert, um diese Übereinstimmung zu erhalten.
  4. Für Kreisdiagramme und Textdiagramme geben Sie einen *Größe*-Wert ein, der den *Maximalwert* der Attribute darstellt. Für jedes Objekt wird dann die Kreisfläche oder der Durchmesser linear skaliert, um diese Übereinstimmung zu erhalten (von `0`). Ein *Minimalgröße* kann jedoch für kleine Diagramme gesetzt werden.

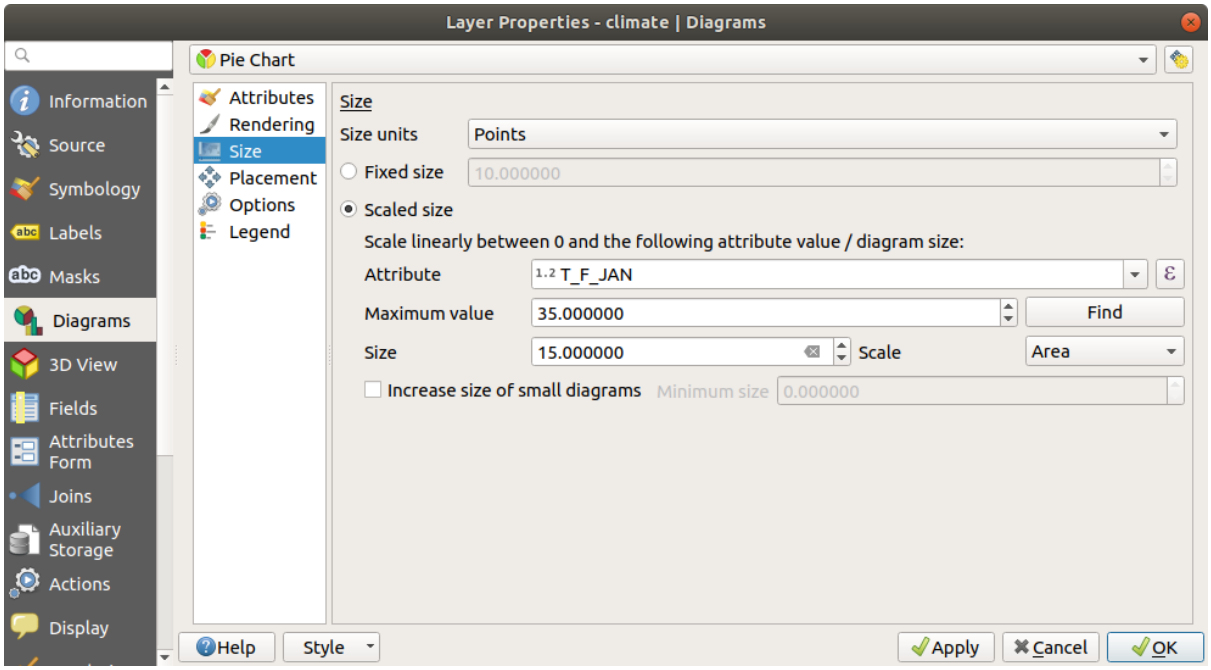


Abb. 12.36: Diagrammeigenschaften - Größe-Reiter

Platzierung

Platzierung definiert die Position des Diagramms. Je nach Art der Layer-Geometrie bietet es verschiedene Optionen für die Platzierung (mehr Details unter *Platzierung*):

- *Um Punkt* oder *Über Punkt* für Punktgeometrie. Die erste Variable erfordert einen Radius, dem sie folgen muss.
- *Um Linie* oder *Über Linie* für Liniengeometrie. Wie bei Punktobjekten erfordert die erste Variable einen Abstand, der eingehalten werden muss, und Sie können die Platzierung des Diagramms relativ zum Objekt festlegen (‚über', ‚auf' und/oder ‚unter' der Linie). Es ist möglich, mehrere Optionen auf einmal zu wählen. In diesem Fall sucht QGIS nach der optimalen Position für das Diagramm. Denken Sie daran, dass Sie auch die Linienausrichtung für die Position des Diagramms verwenden können.
- *Um Zentroid*, *Über Zentroid*, *Nach Umfang* und *In Polygon* sind die Optionen für Polygonobjekte, jeweils in einer bestimmten *Entfernung*.

Der Bereich *Koordinaten* ermöglicht die direkte Steuerung der Diagrammplatzierung für jedes einzelne Objekt, indem Attribute oder ein Ausdruck verwendet werden, um die Koordinaten *X* und *Y* zu setzen. Die Informationen können auch mit dem Werkzeug *Beschriftungen und Diagramme verschieben* gefüllt werden.

Im Abschnitt *Priorität* können Sie die Platzierungspriorität jedes Diagramms festlegen, d. h. wenn mehrere Diagramme oder Beschriftungen für denselben Ort in Frage kommen, wird das Element mit der höheren Priorität angezeigt und die anderen können weggelassen werden.

Möglichst keine Objekte durch Diagramme und Beschriftungen verdecken definiert Objekte, die als *Hindernisse* verwendet werden sollen, d. h. QGIS versucht, keine Diagramme oder Beschriftungen über diesen Features zu platzieren. Der Prioritätsrang wird dann verwendet, um zu bewerten, ob ein Diagramm aufgrund eines höher gewichteten Hindernis-Features weggelassen werden kann.

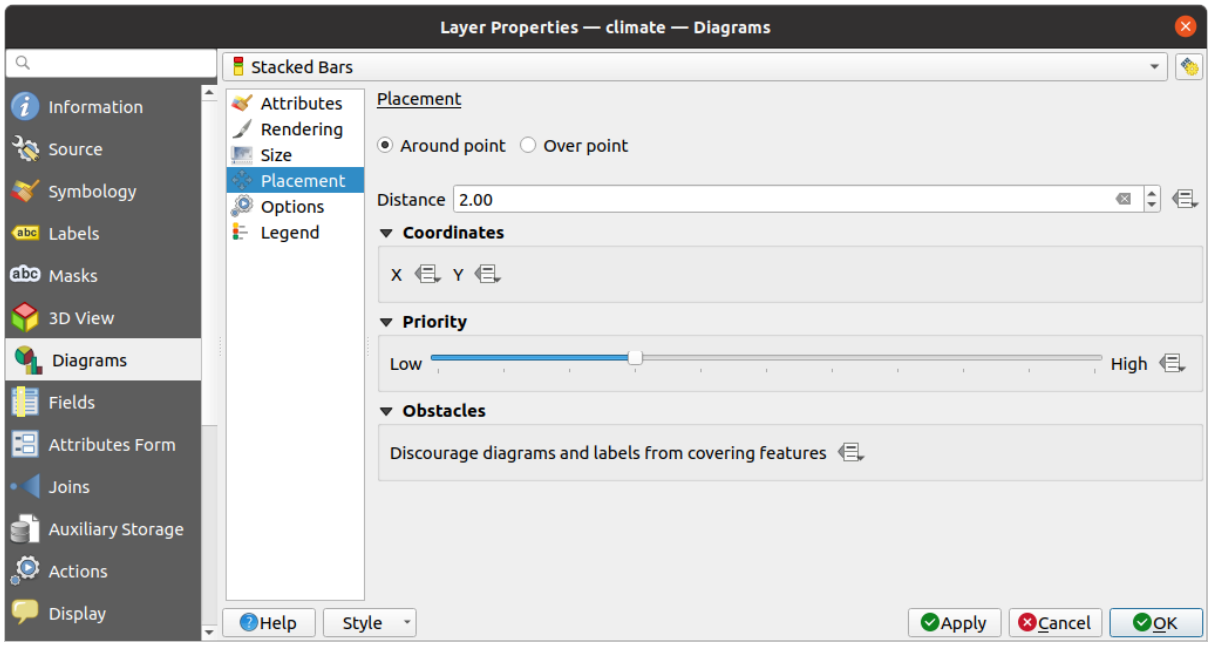


Abb. 12.37: Diagrammeigenschaften - Reiter Platzierung

Optionen

Der Reiter *Optionen* enthält Einstellungen für Histogramme und gestapelte Balken. Sie können wählen, ob die *Balkenorientierung Auf*, *Ab*, *Rechts* oder *Links*, für horizontale und vertikale Diagramme sein soll.

Legende

Auf dem Reiter *Legende* können Sie wählen, ob Elemente des Diagramms im *Layer-Bedienfeld* und in der *Legende eines Drucklayouts* neben der Layer-Symbolik angezeigt werden sollen:

- Aktivieren Sie *Legendeneinträge für Diagrammattribute anzeigen*, um in den Legenden die Eigenschaften `Farbe` und `Legende` anzuzeigen, die zuvor auf der Registerkarte *Attribute* zugewiesen wurden;
- und wenn eine *Skalierte Größe* für die Diagramme verwendet wird, drücken Sie die Schaltfläche *Legendeneinträge für Diagrammgröße…*, um den Diagrammsymbolaspekt in den Legenden zu konfigurieren. Dies öffnet den Dialog *datendefiniert Legendengröße*, dessen Optionen in *Datendefinierte Legendengröße* beschrieben sind.

Wenn diese Option aktiviert ist, werden die Elemente der Diagrammlegende (Attribute mit Farbe und Diagrammgröße) auch in der Legende des Drucklayouts angezeigt, und zwar neben der Layer-Symbolik.

Gestapelte Diagramme

Gestapelte Diagramme ermöglichen es Benutzern, komplexe Diagramme wie Bevölkerungspyramiden zu erstellen, bei denen zwei Unterdiagramme, nämlich Histogramme, nebeneinander angeordnet und horizontal angezeigt werden.

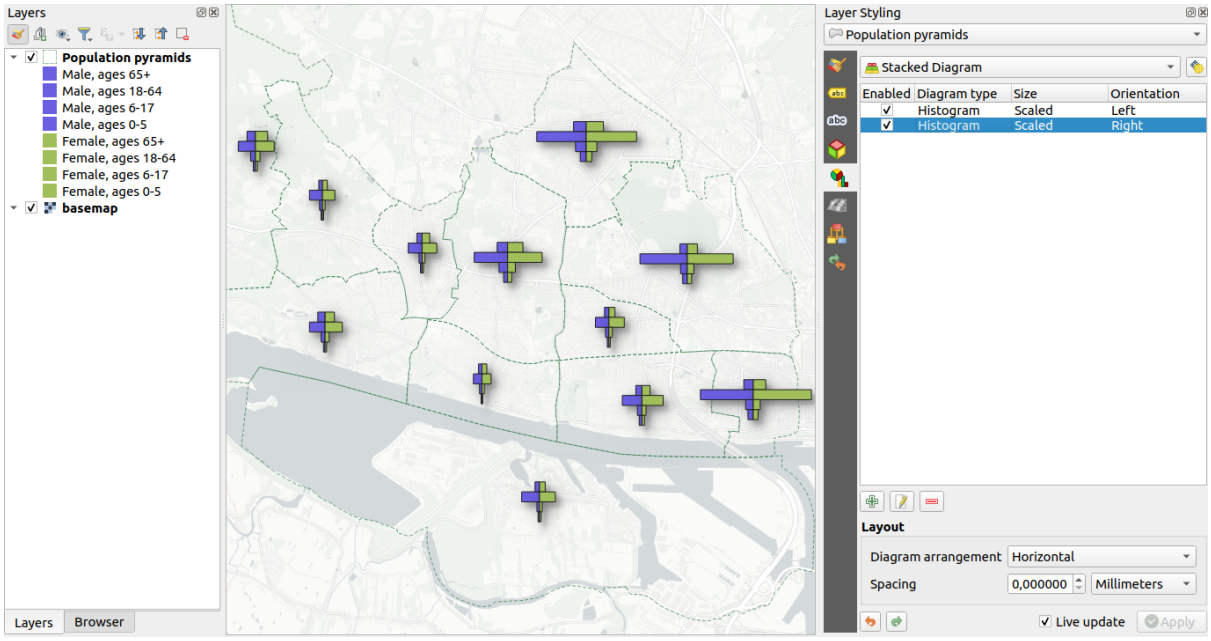


Abb. 12.38: Für jeden Layer erstellte Bevölkerungspyramiden

Multitemporale Diagramme können auch als gestapelte Diagramme erstellt werden. Die Anzahl der Unterdiagramme sowie der Abstand zwischen ihnen können konfiguriert werden.

Darüber hinaus können Unterdiagramme unterschiedliche Typen haben (z. B. ein Kreisdiagramm neben einem Histogramm) und über eigene unabhängige Einstellungen verfügen, wie z. B. *Attribute*, *Darstellung*, *Größe*, *Optionen* und *Legende*.

Die Einstellungen unter *Platzierung* in einem gestapelten Diagramm sowie einige Sichtbarkeitseinstellungen (zu finden unter der Registerkarte *Darstellung*) werden durch die Platzierungs- und Sichtbarkeitseinstellungen des ersten Unterdiagramms im Stapel bestimmt.

Schließlich wird die Reihenfolge der Unterdiagramme durch die Reihenfolge der Elemente in der Liste des gestapelten Diagramms vorgegeben. Das erste Unterdiagramm erscheint in einem horizontal gestapelten Diagramm links oder in einem vertikalen Diagramm oben.

12.1.6 Eigenschaften Masken

Die Registerkarte *Masken* hilft Ihnen bei der Konfiguration der Überlagerung von Symbolen des aktuellen Layers mit anderen Symbolebenen oder Beschriftungen eines beliebigen Layers. Dies soll die Lesbarkeit von Symbolen und Beschriftungen verbessern, deren Farben nahe beieinander liegen und bei Überlagerungen schwer zu entziffern sind; es fügt eine benutzerdefinierte und transparente Maske um die Elemente hinzu, um Teile der Symbolebenen des aktuellen Layers zu „verstecken“.

Um Masken auf den aktiven Layer anzuwenden, müssen Sie zunächst im Projekt entweder *Masken im Symbol-Layer* oder *Masken in den Beschriftungseinstellungen* aktivieren. Aktivieren Sie dann auf der Registerkarte *Masken*:

- *Maskierter Symbollayer*: listet in einer Baumstruktur alle Symbolebenen des aktuellen Layers auf. Dort können Sie das Element der Symbolebene auswählen, das Sie transparent „ausschneiden“ möchten, wenn es die ausgewählten Maskenquellen überlappt
- *Maskenquelle*: listet alle im Projekt definierten Maskenetiketten und Layer für Maskensymbole auf. Wählen Sie die Elemente aus, die die Maske über den ausgewählten maskierten Symbol Layern erzeugen würden.

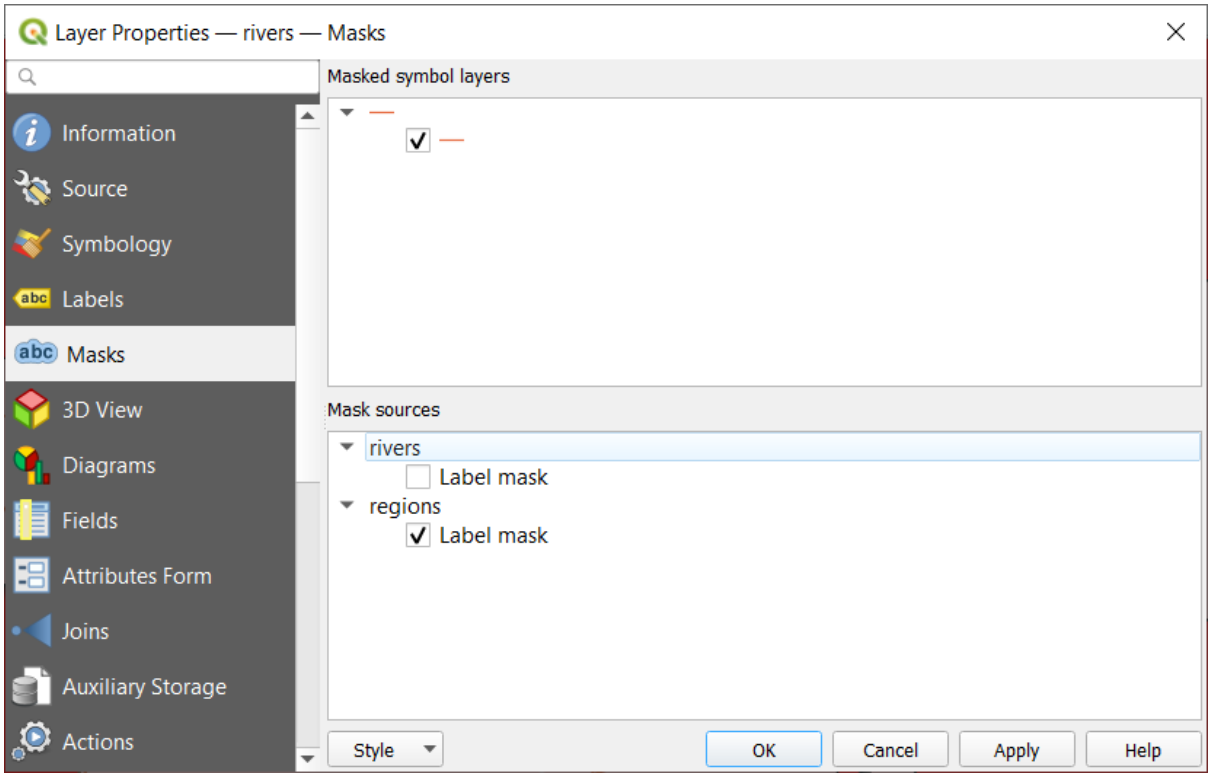


Abb. 12.39: Diagrammeigenschaften - Masken-Reiter

12.1.7 3D-Ansicht Eigenschaften

Die Registerkarte *3D-Ansicht* bietet Einstellungen für Vektorlayer, die der *3D Kartenansicht* dargestellt werden sollen.

Um einen Layer in 3D anzuzeigen, wählen Sie in der Combobox oben auf der Registerkarte entweder

- *Einzel-Symbol*: Objekte werden mit einem gemeinsamen 3D-Symbol gerendert, dessen Eigenschaften datendefiniert für jeden Layer-Geometrietyp.
- *Regelbasierend*: es können mehrere Symbolkonfigurationen definiert und selektiv auf der Grundlage von Ausdrucksfiltern und Skalenbereichen angewendet werden. Weitere Details zur Vorgehensweise unter *Regelbasiertes Rendering*.

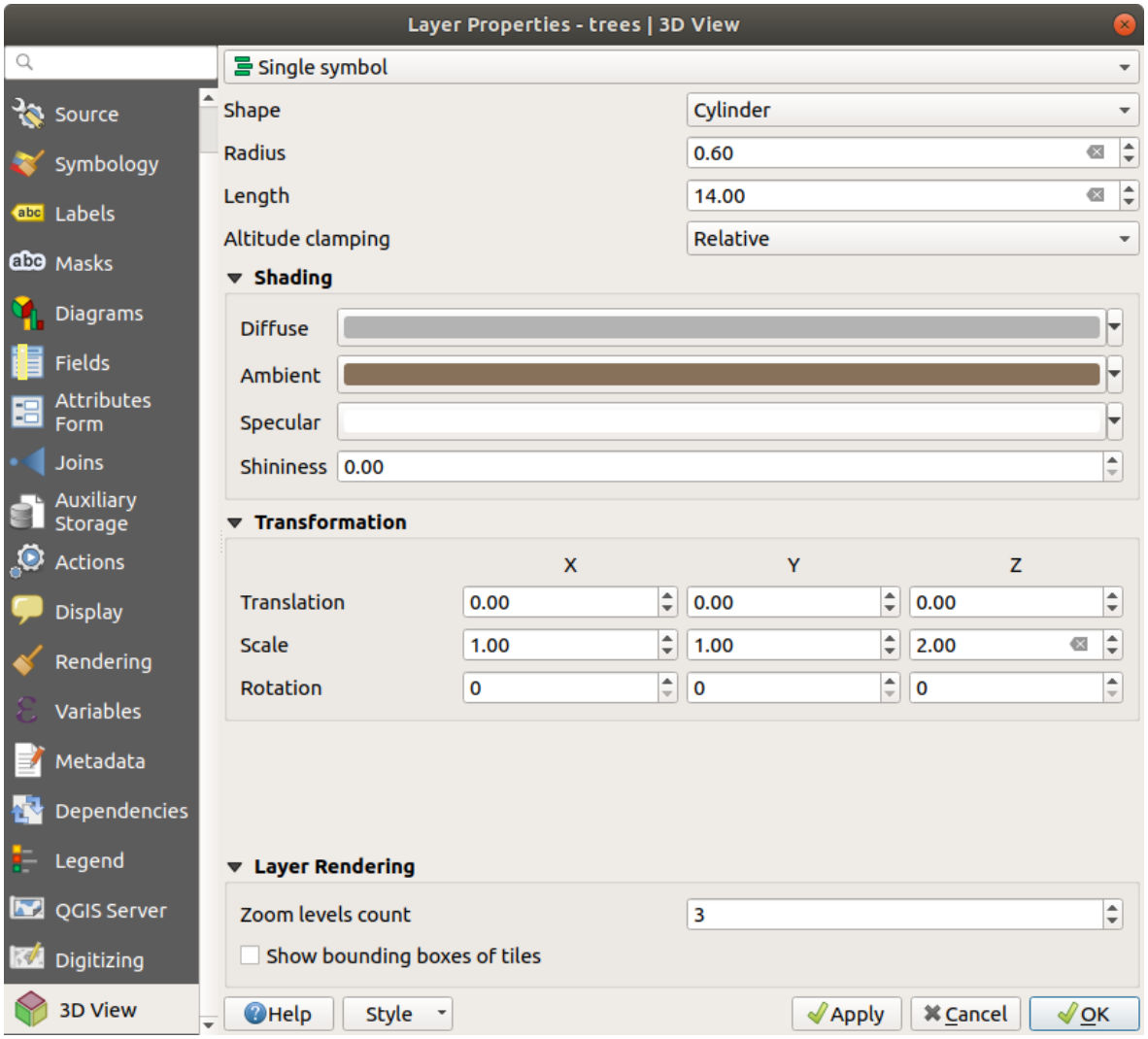


Abb. 12.40: 3D-Eigenschaften eines Punkt-Layers

Achtung

Ziehen Sie für Symbolhöhen- und Geländeeinstellungen die *Höhe* Registerkarte vor

Die Höhen- und höhenbezogenen Eigenschaften von Objekten (*Höhenbegrenzung*, *Höhenbindung*, *Extrudierung* oder *Höhe*) in der Registerkarte *3D-Ansicht* erben ihre Standardwerte von den *Höheneigenschaften* des Layers und sollten vorzugsweise in der Registerkarte *Höhe* eingestellt werden.

Um die Leistung zu verbessern, werden die Daten aus den Vector Layern im Hintergrund unter Verwendung von Multithreading geladen und in Kacheln gerendert, deren Größe im Bereich *Layerdarstellung*, unten auf der Registerkarte, gesteuert werden kann:

- *Zoomstufenanzahl*: bestimmt, wie tief der Quadtree sein wird. Eine Zoomstufe bedeutet zum Beispiel, dass der gesamte Layer aus einer einzigen Kachel besteht. Drei Zoomstufen bedeuten, dass es 16 Kacheln auf der Blattebene gibt (jede weitere Zoomstufe multipliziert diese Zahl mit 4). Die Voreinstellung ist `3` und das Maximum ist `8`.
- *Boundingboxes der Kacheln anzeigen*: besonders nützlich, wenn es Probleme mit Kacheln gibt, die nicht angezeigt werden, wenn sie sollten.

12.1.8 Eigenschaften Felder

Auf der Registerkarte *Felder* finden Sie Informationen zu Feldern, die mit dem Layer zusammenhängen, und können sie organisieren.

Der Layer kann mit der Schaltfläche Bearbeitungsmodus umschalten *editierbar* gemacht werden. In diesem Moment können Sie seine Struktur mit den Schaltflächen Neues Feld und Feld löschen ändern.

Wenn ein Neues Feld hinzugefügt wird, ist die Option *Kommentar* nur für Datenquellen verfügbar, die die Bearbeitung von Kommentaren erlauben (siehe *Datenbankeinträge* für weitere Details). Sie können auch Aliase im Dialog *Feld hinzufügen* für unterstützte OGR-Formate (GeoPackage und ESRI File Geodatabase) setzen.

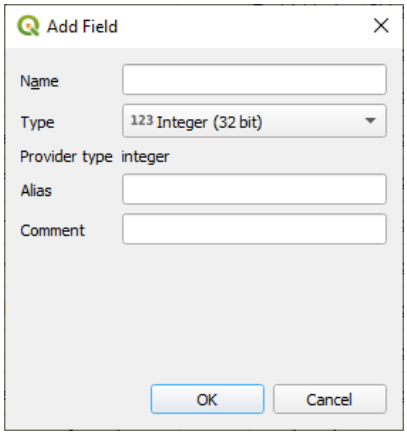


Abb. 12.41: Dialog Feld hinzufügen

Sie können Felder auch umbenennen, indem Sie auf ihren Namen doppelklicken. Dies wird nur für Datenanbieter wie PostgreSQL, Oracle, Memory Layer und einige GDAL Layer je nach GDAL Version unterstützt.

Wenn in der zugrunde liegenden Datenquelle oder in den *Formulareigenschaften* festgelegt, wird auch der Alias des Feldes angezeigt. Ein Alias ist ein von Menschen lesbarer Feldname, den Sie im Objektformular oder in der Attributtabelle verwenden können. Aliasnamen werden in der Projektdatei gespeichert.

Neben den im Datensatz enthaltenen Feldern, einschließlich *virtueller Felder* und *Zusatzspeicher*, werden auf der Registerkarte *Felder* auch Felder aus allen *verknüpften Layern* aufgeführt. Je nach Herkunft des Feldes wird eine andere Hintergrundfarbe verwendet.

Für jedes aufgelistete Feld listet das Dialogfeld auch schreibgeschützte Eigenschaften wie *Typ*, *Typname*, *Länge* und *Genauigkeit*` auf.

Je nach Datenanbieter können Sie einem Feld einen Kommentar zuordnen, z. B. bei seiner Erstellung. Diese Information wird abgerufen und in der Spalte *Kommentar* angezeigt und wird später angezeigt, wenn der Mauszeiger über die Feldbeschriftung in einem Objektformular bewegt wird.

In der Spalte *Konfiguration* können Sie festlegen, wie sich das Feld unter bestimmten Umständen verhalten soll:

- `Nicht durchsuchbar`: aktivieren Sie diese Option, wenn Sie nicht möchten, dass dieses Feld von der *Suchleiste* abgefragt wird.
- `Nicht via WMS offenlegen`: Aktivieren Sie diese Option, wenn Sie dieses Feld nicht anzeigen möchten, wenn der Layer als WMS vom QGIS-Server bereitgestellt wird
- `Nicht via WFS offenlegen`: Aktivieren Sie diese Option, wenn Sie dieses Feld nicht anzeigen möchten, wenn der Layer als WFS vom QGIS-Server bereitgestellt wird

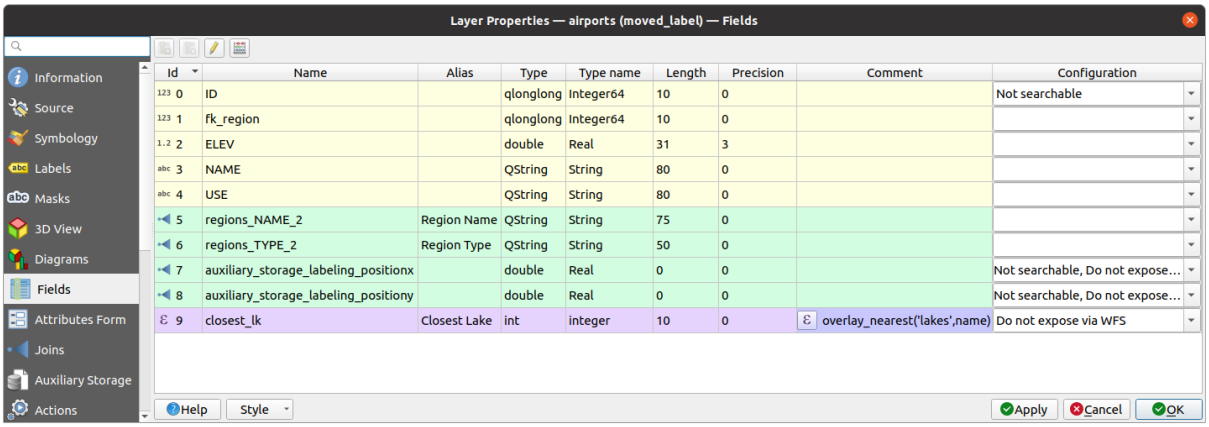


Abb. 12.42: Layereigenschaften - Felder Reiter

12.1.9 Attributformular

Der Reiter *Attributformular* hilft Ihnen dabei, das Formular einzurichten, das beim Erstellen neuer Features, beim Bearbeiten oder Abfragen bestehender Features angezeigt wird. Dies wirkt sich sowohl auf die *Tabellen- als auch auf die Formularansicht* aus. Sie können Folgendes definieren:

- das Aussehen und Verhalten (Beschriftung, Sichtbarkeit, Widget, Einschränkungen…) jedes Feldes im aktuellen Layer, den *verknüpften Layern* und den *in Beziehung gesetzten Layern*;
- die Platzierung von aktivierten *Aktionen*;
- die Struktur des Formulars (benutzerdefiniert oder automatisch generiert);
- zusätzliche HTML-, QML- und Text-Widgets zur Verbesserung des Formulars;
- zusätzliche Logik in Python für die Interaktion mit dem Formular oder den Feldelementen.

Oben rechts im Dialog können Sie einstellen, ob das Formular bei der Erstellung neuer Features standardmäßig geöffnet wird. Dies kann pro Ebene oder global mit der Option *Attributformular-Popup nach Feature-Erstellung unterdrücken* im Menü *Einstellungen* ► *Optionen* ► *Digitalisieren* konfiguriert werden.

Anpassen eines Formulars für Ihre Daten

Standardmäßig zeigt QGIS ein einfaches Formular mit vordefinierten Widgets (in der Regel Spinboxen und Textboxen — jedes Feld wird in einer eigenen Zeile durch seine Beschriftung neben dem Widget dargestellt) an, wenn Sie auf ein Objekt mit dem Werkzeug Objekte abfragen klicken oder die Attributtabelle in den *Formularansicht*-Modus schalten. Wenn *Verknüpfungen* auf dem Layer gesetzt sind, werden Felder aus den referenzierenden Layern in einem eingebetteten Rahmen am unteren Rand des Formulars angezeigt, der der gleichen Grundstruktur folgt.

Diese Darstellung ist das Ergebnis des Standardwertes `Automatisch erzeugen`. Für diesen Eigenschaftsreiter stehen insgesamt drei verschiedene Werte zur Verfügung:

- `Automatisch erzeugen`: behält die Grundstruktur „eine Zeile - ein Feld“ für das Formular bei, ermöglicht aber die Anpassung jedes entsprechenden Widgets.
- `Mit Drag-and-drop zusammenstellen`: Neben der Anpassung von Widgets kann die Formularstruktur komplexer gestaltet werden, z. B. mit Widgets, die in Gruppen und Registerkarten eingebettet sind.
- `UI-Datei verwenden`: erlaubt es, eine Qt-Designer-Datei, also eine potentiell komplexere und voll ausgestattete Vorlage, als Objekt-Form zu verwenden.

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.

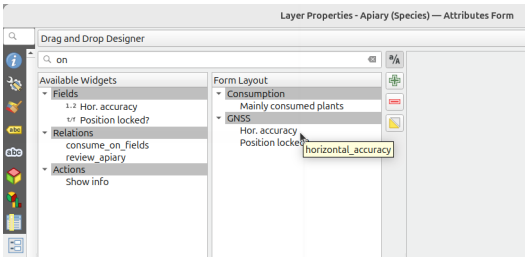


Abb. 12.43: Search fields using aliases and names in Attributes Form page

Das automatisch erzeugte Formular

Wenn die Option „Automatisch generieren“ aktiviert ist, werden im Bereich „Verfügbare Elemente“ Listen mit Feldern (aus dem aktuellen Layer, den Verknüpfungs- und Beziehungs-Layern) und den aktivierten Aktionen angezeigt, die im Formular angezeigt werden würden. Wählen Sie ein Feld aus, um dessen Darstellung und Verhalten im rechten Bereich zu konfigurieren:

- *Individuelle Beschriftungen und automatische Kontrollen* für das Feld hinzufügen;
- *bestimmtes Funktionen* setzten.

In diesem Bearbeitungsmodus können nur die Eigenschaften der Aktionen angezeigt werden. Außerdem werden sie über ein Dropdown-Menü „Aktionen“ im Funktionsformular und in anderen Kontexten nach der Eigenschaft „In Attributtabelle anzeigen“ angezeigt.

Der Drag-and-drop-Designer

Mit dem Drag-and-Drop-Designer können Sie ein Formular mit mehreren Containern (Registerkarten oder Gruppen) erstellen, um die Attributfelder oder andere Widgets darzustellen, die nicht direkt mit einem bestimmten Feld verknüpft sind (wie die HTML/QML-Widgets oder die für den Layer definierten *Aktionen*), wie z. B. in Abb. 12.44 gezeigt.

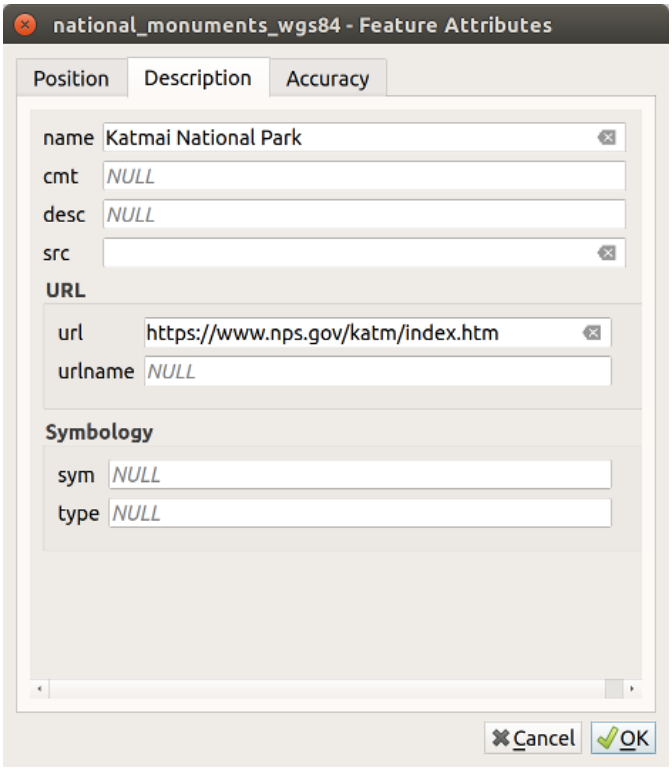


Abb. 12.44: Das Ergebnis eines integrierten Formulars mit Reitern und benannten Gruppen

1. Wählen Sie aus der Combobox die Option `Mit Drag and Drop zusammenstellen`. Dies aktiviert das Bedienfeld *Formularlayout* neben dem Bedienfeld *Verfügbare Elemente*, das mit vorhandenen Feldern gefüllt ist. Das ausgewählte Feld zeigt seine *Eigenschaften* in einem dritten Bedienfeld an.
2. Markieren Sie die Felder, die Sie nicht in Ihrem *Formularlayout* Bedienfeld verwenden möchten, und drücken Sie die Schaltfläche , um sie zu entfernen. Sie können die Auswahl auch mit der Schaltfläche Auswahl umkehren umschalten.
3. Ziehen Sie Felder aus dem ersten Bedienfeld in das Bedienfeld *Formularlayout*, um sie erneut hinzuzufügen. Dasselbe Feld kann mehrfach hinzugefügt werden.
4. Durch Ziehen und Ablegen von Feldern im Bedienfeld *Formularlayout* können Sie deren Position ändern.
5. Fügen Sie Behältnisse hinzu, um Felder, die zur gleichen Kategorie gehören, zu verknüpfen und das Formular besser zu strukturieren.

    1. Der erste Schritt ist die Verwendung des Symbols Fügt einen neuen Reiter oder eine Gruppe zum Formularlayout hinzu. Darin werden Felder und andere Gruppen angezeigt.
    2. Legen Sie dann die Eigenschaften des Containers fest, d. h.:

        - *Beschriftung*: der Titel, der für das Behältnis verwendet werden soll
        - *Behältertyp*: Das kann ein *Reiter*, ein *Gruppenrahmen* (ein zusammenklappbarer Eintrag innerhalb eines Tabs oder einer anderen Gruppe) oder eine *Zeile* (ein Containertyp, mit dem Sie Ihre Widgets in einer horizontalen Reihe anordnen können, wobei die Anzahl der Spalten automatisch auf der Grundlage der Anzahl der Widgets bestimmt wird) sein,
        - *Within*: Mit dieser Option können Sie ein bestehendes Behältnis auswählen, in den das neue Behältnis (*Gruppenrahmen* oder *Zeile*) eingebettet werden soll.
        - *Spaltenanzahl* über die die eingebetteten Felder verteilt werden sollen

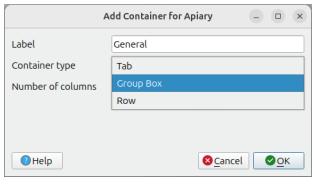


Abb. 12.45: Dialog zum Erstellen von Containern auf dem Attributformular-Reiter

Diese und andere Eigenschaften können später aktualisiert werden, indem das Element im Bereich *Formularlayout* ausgewählt wird; Sie können dann:

- die Beschriftung des Behältnisses ein- oder ausblenden
- das Behältnis umbenennen
- die Anzahl der Spalten festlegen
- geben Sie einen Ausdruck ein, um die Sichtbarkeit des Containers zu steuern. Der Ausdruck wird jedes Mal neu ausgewertet, wenn sich die Werte im Formular ändern, und die Registerkarte oder das Gruppenfeld entsprechend angezeigt/ausgeblendet
- *Als Gruppenfeld anzeigen*: wandelt einen Tabulator in ein Gruppenfeld der obersten Ebene um und umgekehrt
- im Falle eines Gruppenrahmens, legen Sie fest, ob er beim Öffnen des Formulars als *Eingeklappt* für alle Objekte angezeigt werden soll, oder nur für diejenigen, die einem Ausdruck entsprechen (*Bearbeitungselement eingeklappt nach Ausdruck*).
- den *Stil* des Behälters konfigurieren, mit benutzerdefinierten *Hintergrundfarbe*, Etikettenfarbe und Schrifteigenschaften

3. Sie können beliebig viele Container erstellen und einbetten, indem Sie erneut auf das Symbol Fügt einen Reiter oder eine Gruppe zum Formularlayout hinzu drücken.

6. Im nächsten Schritt werden die entsprechenden Felder durch einfaches Ziehen und Ablegen den einzelnen Containern zugewiesen. Auf die gleiche Weise können auch Gruppenrahmen und Registerkarten verschoben werden.
7. *Anpassen des Widgets* der verwendeten Felder
8. Falls der Layer in eine *Eins-zu-viele- oder viele-zu-viele-Beziehung* eingebunden ist, ziehen Sie den Namen der Beziehung aus dem Bedienfeld *Verfügbare Elemente* auf das Bedienfeld *Formularlayout*. Das zugehörige Layer-Attribut-Formular wird an der gewählten Stelle in das Formular des aktuellen Layers eingebettet. Wie bei den anderen Elementen wählen Sie das Beziehungsetikett, um einige Eigenschaften zu konfigurieren:
   - das Beziehungsetikett ein- oder ausblenden
   - den Link-Button anzeigen
   - den Unlink-Button anzeigen
9. Falls der Layer eine oder mehrere *Aktionen* für den Bereich `Layer` oder `Feature` aktiviert hat, werden diese unter *Aktionen* aufgelistet und können wie bei den anderen Feldern per Drag and Drop eingefügt werden. Die zugehörige Aktion wird an der gewählten Stelle im Formular des aktuellen Layers eingebettet.
10. Weitere Anpassungen des Formulars durch Hinzufügen eines oder mehrerer Elemente aus *Andere Bedienelemente* (siehe *Andere Bedienelemente*)
11. Anwenden des Dialogs für die Eigenschaften des Layers
12. Öffnen Sie ein Formular für Objektattribute (z. B. mit dem Werkzeug Objekte abfragen) und das neue Formular sollte angezeigt werden.

Andere Bedienelemente

Der Drag-and-Drop-Designer bietet eine Reihe von Werkzeugen, die nicht mit einem bestimmten Feld verbunden sind. Sie können verwendet werden, um das Aussehen des Formulars zu verbessern oder um dynamisch berechnete Werte anzuzeigen.

- *HTML Widget*: bettet eine HTML-Seite ein, die HTML-Quelle kann das Ergebnis von dynamisch berechneten Ausdrücken enthalten.

  HTML-Bedienelement können zum Beispiel verwendet werden, um Bilder anzuzeigen, die als BLOB in einem Feld gespeichert sind (nennen wir es `Foto`):

  1. Fügen Sie im `Drag-and-drop Designer`-Modus ein *HTML Widget* zu Ihrem *Formularlayout* hinzu.
  2. Doppelklicken Sie auf das *HTML-Bedienelement*, um es zu konfigurieren.
  3. Ändern Sie den standardmäßigen *Title* oder blenden Sie es aus.
  4. Drücken Sie die Taste und geben Sie den folgenden QGIS-Ausdruck ein:

     ```
     '<img src= "' || to_base64("photo") || '">'
     ```

     Stellen Sie sicher, dass Sie *photo* durch Ihren eigenen BLOB-Feldnamen ersetzen. Der obige Ausdruck erzeugt eine Zeichenkette mit HTML-Bild-Tag, in der die BLOB-Datei kodiert ist.
  5. Übernehmen Sie den Dialog und drücken Sie dann die Schaltfläche .
  6. QGIS wendet automatisch HTML-Formatierung und Funktionen an, um Ihren Ausdruck auszuwerten, was zu folgendem Code führt:

     ```
     <script>document.write(expression.evaluate("'<img src=\"data:image/png;
     ↪base64,' || to_base64(\"photo\") || '\">'"));</script>
     ```

     Auf der rechten Seite wird eine Vorschau Ihres Bildes angezeigt.

- *QML Widget*: Bettet ein Qt QML Dokument ein und zeigt grafische Elemente in Form von Attributen an. Neben dem benutzerdefinierten *Titel* des hinzugefügten Widgets und der Entscheidung, ob es angezeigt werden soll oder nicht, können Sie aus vordefinierten *QML-Kode*-Elementen auswählen:
  - *Freier Text…*: Ermöglicht es Ihnen, einen neuen Text zu verfassen oder einen bestehenden Code einzufügen.
  - *Rechteck*: stellt einen minimalen Code zum Anzeigen eines Rechtecks bereit
  - *Balkendiagramm*: bietet einen minimalen Code zum Anzeigen eines Balkendiagramms
  - *Kuchendiagramm*: stellt einen minimalen Code für die Anzeige eines Kuchendiagramms bereit

  Sie können den Code mit QML-Syntax erweitern, Layer-Felder oder QGIS-Ausdrücke verwenden, die dynamisch berechnet werden.

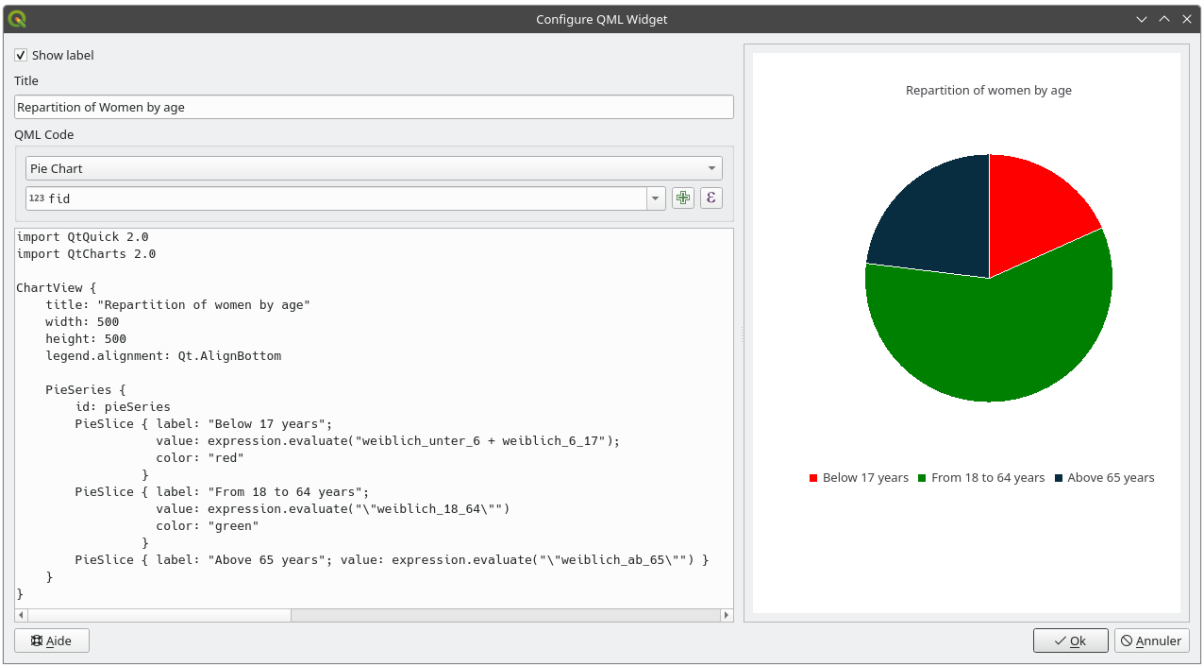


Abb. 12.46: Ein QML-Diagramm so einstellen, dass es in Attributformuar angezeigt wird

- *Textbedienungselement*: zeigt ein Textelement an, das einfache HTML-Auszeichnungen unterstützt und das Ergebnis von dynamisch berechneten Ausdrücken enthalten kann.
- *Abstandshalterbearbeitungselement*: fügt ein leeres transparentes Rechteck ein, das den vertikalen Abstand zwischen zwei Widgets vergrößert.

Tipp

Dynamische Inhalte anzeigen

Die oben erwähnten Elemente (außer dem *Abstandshalterbearbeitungselement*) unterstützen Ausdrücke, die verwendet werden können, um Inhalte anzuzeigen, die sich dynamisch ändern, wenn sich ein anderes Feld im Formular ändert. Dies kann erreicht werden, indem man die Funktion `current_value('field_name')` im Ausdruck verwendet, um den Wert eines anderen Feldes zu untersuchen.

Verwendung einer benutzerdefinierten UI-Datei

Die Option `UI-Datei verwenden` erlaubt es, komplexe Dialoge, die mit Qt-Designer erstellt wurden, zu verwenden. Die Verwendung einer UI-Datei erlaubt eine große Freiheit bei der Erstellung eines Dialogs. Beachten Sie, dass Sie den grafischen Objekten (Textbox, Combobox…) den gleichen Namen geben müssen, um sie mit den Feldern des Layers zu verbinden.

Verwenden Sie das Feld *UI zur Bearbeitung*, um den Pfad zu der zu verwendenden Datei zu definieren.

UI-Dateien können auch auf einem entfernten Server gehostet werden. In diesem Fall geben Sie die URL des Formulars anstelle des Dateipfads in *UI zur Bearbeitung* an.

Einige Beispiele finden Sie in der Lektion Erstellen eines neuen Formulars der QGIS-training-manual-index-reference. Weitergehende Informationen finden Sie unter https://woostuff.wordpress.com/2011/09/05/qgis-tips-custom-feature-forms-with-python-logic/.

Erweitern Sie Ihr Formular mit benutzerdefinierten Funktionen

QGIS-Formulare können eine Python-Funktion enthalten, die aufgerufen wird, wenn der Dialog geöffnet wird. Verwenden Sie diese Funktion, um Ihren Dialogen zusätzliche Logik hinzuzufügen. Der Formularcode kann auf drei verschiedene Arten angegeben werden:

- `Aus der Umgebung laden`: verwendet eine Funktion, zum Beispiel in `startup.py` oder von einem installierten Plugin
- `Aus einer externen Datei laden`: Sie können eine Python-Datei aus Ihrem Dateisystem auswählen oder eine URL für eine entfernte Datei eingeben.
- `Code in diesem Dialogfeld bereitstellen`: es wird ein Python-Editor angezeigt, in den Sie die zu verwendende Funktion direkt eingeben können.

In allen Fällen müssen Sie den Namen der Funktion angeben, die aufgerufen werden soll (`open` im Beispiel unten).

Ein Beispiel ist (im Modul MyForms.py):

```
def open(dialog,layer,feature):
    geom = feature.geometry()
    control = dialog.findChild(QWidget,"My line edit")
```

Referenz in Python Init Funktion wie folgt: `open`

Einstellung der Feldfunktionsweisen

Der Hauptteil der Registerkarte *Attributformular* hilft Ihnen dabei festzulegen, wie das Ausfüllen oder Anzeigen der Werte des Feldes in der Attributtabelle oder im Feature-Formular ausgeführt wird: Sie können festlegen, wie der Benutzer mit jedem Feld interagiert und welche Werte oder Wertebereiche jeweils hinzugefügt werden dürfen.

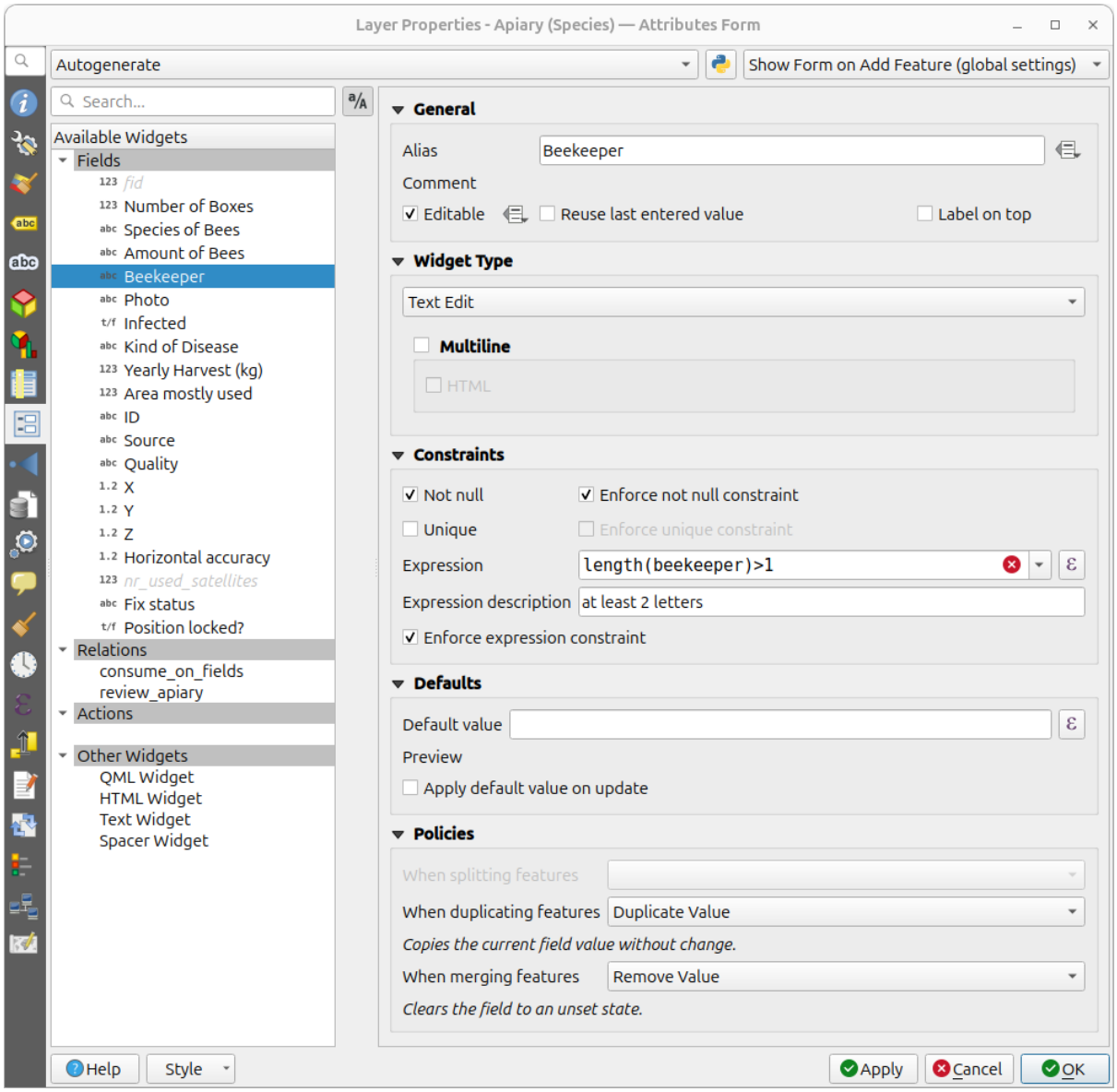


Abb. 12.47: Dialog um ein Bearbeitungselement für eine Attributspalte auszuwählen

Tipp

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.

Allgemeine Einstellungen

Unabhängig von der Art des Widgets, das auf das Feld angewendet wird, gibt es einige allgemeine Eigenschaften, die Sie einstellen können, um zu steuern, ob und wie ein Feld bearbeitet werden kann.

Bedienelementanzeige

Dieser Bereich ist nur für den Designermodus *Drag and drop* verfügbar und hilft Ihnen, das Aussehen des dem Feld zugewiesenen Elements zu konfigurieren:

- *Beschriftung anzeigen*: gibt an, ob der Feldname im Formular angezeigt werden soll
- *Beschriftungsfarbe übersteuern*: wendet eine bestimmte Farbe auf die Beschriftung des Feldes an
- *Beschriftungsschriftart übersteuern*: wendet bestimmte Schrifteigenschaften (fett, kursiv, unterstrichen, durchgestrichen oder Schrift) auf die Feldbezeichnung an
- *Size*: Ermöglicht die Steuerung der relativen Größenänderung von Elementen bei der Größenänderung eines Formularattributs.
  - *Horizontale Dehnung*: setzt einen höheren horizontalen Wert für Elemente, die mehr horizontalen Platz benötigen.
  - *Vertikale Dehnung*: Legt fest, wie sich die Widgets vertikal vergrößern, wenn die Größe des Formulars geändert wird.

Allgemein

- *Alias*: einen menschenlesbaren Namen, der für Felder verwendet werden soll. Der Alias wird im Objektformular, in der Attributtabelle oder im Bedienfeld *Identifikationsergebnisse* angezeigt. Er kann auch als Feldnamenersatz im *Ausdruckseditor* verwendet werden, was das Verständnis und die Überprüfung von Ausdrücken erleichtert. Aliase werden in der Projektdatei gespeichert.
- *Kommentar: zeigt den Kommentar des Feldes an, wie er auf der Registerkarte :guilabel:``Felder* angezeigt wird, in einem schreibgeschützten Zustand. Diese Information wird als Tooltip angezeigt, wenn der Mauszeiger über die Feldbeschriftung in einem Objektformular bewegt wird.
- ☑ *Editable*: Deaktivieren Sie diese Option, um das Feld schreibgeschützt (nicht manuell änderbar) zu machen, auch wenn sich der Layer im Bearbeitungsmodus befindet. Beachten Sie, dass das Markieren dieser Einstellung keine Bearbeitungsbeschränkung des Anbieters außer Kraft setzt. Diese Option kann dank der Schaltfläche Datendefinierte Übersteuerung durch eine daten-definierte Eigenschaft gesteuert werden.
- ☑ *Wiederverwenden des zuletzt eingegebenen Wertes*: merkt sich den zuletzt in dieses Feld eingegebenen Wert und verwendet ihn als Standard für das nächste Objekt, das im Layer bearbeitet wird.
- ☑ *Beschriftung darüber*: platziert den Feldnamen über oder neben dem Element im Objektformular.

Restriktionen

Sie können den Wert, der in das Feld eingefügt werden soll, einschränken. Diese Einschränkung kann sein:

- ☑ *Nicht Null*: verlangt, dass der Benutzer einen Wert angibt;
- ☑ *Eindeutig*: garantiert, dass der eingefügte Wert im gesamten Feld eindeutig ist;
- basierend auf einem benutzerdefinierten *Ausdruck*: z.B. `not regexp_match(col0,'[^A-Za-z]')` stellt sicher, dass der Wert des Feldes *col0* nur Buchstaben des Alphabets enthält. Eine kurze Beschreibung kann hinzugefügt werden, damit Sie sich die Einschränkung besser merken können.

Jedes Mal, wenn ein Wert in einem Feld hinzugefügt oder bearbeitet wird, wird er an die bestehenden Beschränkungen übermittelt:

- wenn es alle Anforderungen erfüllt, wird ein grünes Häkchen neben dem Feld im Formular angezeigt;
- Wenn es nicht alle Anforderungen erfüllt, wird das Feld gelb oder orange gefärbt und ein entsprechendes Kreuz wird neben dem Widget angezeigt. Sie können mit dem Mauszeiger über das Kreuz fahren, um sich daran zu erinnern, welche Einschränkungen auf das Feld angewendet werden, und den Wert korregieren:
  - Ein gelbes Kreuz erscheint, wenn es sich bei der nicht erfüllten Bedingung um eine nicht erzwungene Bedingung handelt (weiche Bedingung) und es Sie nicht daran hindert, die Änderungen mit den „falschen“ Werten zu speichern;
  - Ein orangefarbenes Kreuz kann nicht ignoriert werden und erlaubt es Ihnen nicht, Ihre Änderungen zu speichern, solange sie nicht den Beschränkungen entsprechen. Es erscheint, wenn die Option *Beschränkung erzwingen* aktiviert ist (`harte Einschränkung`).

Vorgaben

- *Vorgabewert*: füllt bei neuen Objekten das Feld automatisch mit einem vordefinierten Wert oder einem *ausdrucksbasierten Wert* aus. Zum Beispiel, können Sie:
  - verwenden Sie `$x`, `$length`, `$area`, um automatisch ein Feld mit der X-Koordinate, der Länge, der Fläche oder anderen geometrischen Informationen des Objekts bei seiner Erstellung zu füllen;
  - ein Feld für jedes neue Objekt um 1 zu erhöhen, indem man `maximum("field")+1` verwendet;
  - den Zeitpunkt der Erstellung des Objekts mit `now()` speichern;
  - Verwenden Sie *Variable* in Ausdrücken, was es einfacher macht, z.B. den Operator-Namen (`@user_full_name`), den Pfad der Projektdatei (`@project_path`), oder anderes einzufügen.

  Eine Vorschau des resultierenden Standardwerts wird am unteren Rand des Widgets angezeigt.

  Bemerkung

  Die Option `Standardwert` kennt die Werte in den anderen Feldern des zu erstellenden Objekts nicht, so dass es nicht möglich ist, einen Ausdruck zu verwenden, der diese Werte kombiniert, d.h. die Verwendung eines Ausdrucks wie `concat( field1, field2)` kann nicht funktionieren.

- *Vorgabewert bei Aktualisierung anwenden*: Wenn das Objektattribut oder die Geometrie geändert wird, wird der Standardwert neu berechnet. Dies könnte praktisch sein, um Werte zu speichern, wie z. B. den letzten Benutzer, der Daten geändert hat, das letzte Mal, als sie geändert wurden…

Richtlinien

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*:

- *Doppelter Wert*: Behält den bestehenden Wert des Feldes für das neue Objekt bei.
- *Vorgabewert verwenden*: Setzt das Feld zurück, indem es seinen Vorgabewert 1 neu berechnet. Wenn keine Vorgabewertklausel vorhanden ist, wird der vorhandene Wert für das neue Objekt beibehalten.
- *Wert entfernen*: Setzt das Feld auf einen ungesetzten Zustand zurück.
- *Geometrieverhältnis verwenden*: Berechnet den Feldwert für alle geteilten Teile neu, indem der vorhandene Wert mit dem Verhältnis der Längen oder Flächen der geteilten Teile multipliziert wird.

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.

Funktionsauswahl

Basierend auf dem Feldtyp bestimmt QGIS automatisch einen Standard-Bedienelementtyp und weist ihn zu. Sie können das Element aber auch durch ein anderes ersetzen, das mit dem Feldtyp kompatibel ist. Die verfügbaren Elemente sind:

- **Binärelement (BLOB)**: ist nur für binäre Felder verfügbar und zeigt standardmäßig eine Beschriftung mit der Größe der eingebetteten Daten an, sofern sie nicht leer ist. Eine Dropdown-Schaltfläche neben der Beschriftung ermöglicht Folgendes:
  - *Eingebettete Datei*, Ersetzen oder Auffüllen des Feldes
  - *Inhalt löschen*, Entfernen aller Daten im Feld
  - *Inhalt in Datei speichern*, Export der Daten als Datei auf Festplatte

  Es ist auch möglich, eine Vorschau der eingebetteten Binärdatei im Feld zu sehen, wenn sie in einem Drag-and-Drop-Formular z.B. mit einem :ref:`QML- oder HTML-Widget <other_widgets> ` kombiniert wird.
- **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.
- **Klassifikation**: Nur verfügbar, wenn eine *kategorisierte Symbologie* auf den Layer angewendet wird, zeigt ein Kombinationsfeld mit den Werten der Klassen an.
- **Farbe**: Zeigt ein *Farb-Widget* an, mit dem eine Farbe ausgewählt werden kann; der Farbwert wird als html-Notation in der Attributtabelle gespeichert.
- **Datum/Zeit**: Stellt ein Linienfeld dar, das ein Kalender-Widget zum Öffnen eines Datums, einer Zeit oder beidem, darstellt. Der Spaltentyp muss Text sein. Sie können ein benutzerdefiniertes Format auswählen, einen Pop-up-Kalender, etc.
- **Aufzählung**: Öffnet ein Kombinationsfeld mit vordefinierten Werten, die aus der Datenbank abgerufen werden. Dies wird derzeit nur vom PostgreSQL-Anbieter für Felder vom Typ `Enum` unterstützt.
- **Anhang**: Verwendet einen Dialog „Datei öffnen", um den Dateipfad in einem relativen oder absoluten Modus zu speichern. Es kann verwendet werden, um den Dokumentpfad als Hyperlink anzuzeigen oder das Dokument in einem speziellen Widget im Formular darzustellen. Unterstützte Dokumenttypen sind Bilder, Webseiten, Audio- und Videodateien, wobei die unterstützten Dateiformate vom Betriebssystem abhängen. Sie können auch ein *externes Speichersystem* konfigurieren, um Ressourcen abzurufen/zu speichern.

> **Tipp**
>
> **Relativer Pfad im Anhänge-Elementtyp**
>
> Befindet sich der mit dem Dateibrowser ausgewählte Pfad im gleichen Verzeichnis wie die `.qgs`-Projektdatei oder darunter, werden Pfade in relative Pfade umgewandelt. Dies erhöht die Portabilität eines `.qgs`-Projekts mit angehängten Multimedia-Informationen.

- **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
- **Versteckt**: Ein verstecktes Attribut ist unsichtbar. Der Anwender kann den Inhalt nicht sehen.
- **Schlüssel/Wert**: Zeigt eine zweispaltige Tabelle an, um Sätze von Schlüssel/Wert-Paaren in einem einzigen Feld zu speichern. Dies wird derzeit vom PostgreSQL-Anbieter für Felder vom Typ `hstore` unterstützt.
- **JSON-Ansicht**: Zeigt JSON-Daten in einer syntaktisch hervorgehobenen Textbearbeitung oder in der Baumansicht an. Dieses Widget ist derzeit nur lesbar. Es stehen mehrere Optionen zur Verfügung, um die Art der Anzeige der Daten zu ändern. Die Option ‚Standardansicht' gibt an, ob das Widget im Text- oder Baummodus angezeigt werden soll. Die Option „JSON formatieren" enthält drei Optionen, die sich nur auf die Strukturansicht beziehen:
  - Eingerückt: Anzeige der Daten in einer für den Menschen lesbaren Form mit Zeilenumbrüchen und vier Leerzeichen als Einrückung.
  - Kompakt: Daten in einer einzeiligen, optimierten Zeichenkette ohne Zeilenumbrüche und Leerzeichen anzeigen.
  - Deaktiviert: Daten so anzeigen, wie sie vom Anbieter kommen.
- **Liste**: Zeigt eine einspaltige Tabelle an, um verschiedene Werte innerhalb eines einzelnen Feldes hinzuzufügen. Dies wird derzeit vom PostgreSQL-Anbieter für Felder vom Typ `Array` unterstützt.
- **Bereich**: Erlaubt Ihnen numerische Werte eines bestimmten Wertebereichs festzulegen. Das Bearbeitungselement kann entweder ein Schieber oder ein Drehfeld sein.

> **Bemerkung**
>
> Einige Layer, wie GeoPackage oder ESRI File Geodatabase, mit vordefinierten **Bereichs-Felddomänen** werden von QGIS automatisch erkannt und mit einem **Bereich**-Widget für die entsprechenden Felder versehen. Das Widget wird mit den in der Domäne angegebenen Minimal- und Maximalwerten vorab ausgefüllt.

- **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.
- **Texteditor** (Standard): Öffnet ein Textbearbeitungsfeld, in dem einfacher Text oder mehrere Zeilen verwendet werden können. Wenn Sie mehrere Zeilen wählen, können Sie auch HTML-Inhalte auswählen.
- **Eindeutige Werte**: Sie können einen der bereits in der Attributtabelle verwendeten Werte auswählen. Wenn „Bearbeitbar" aktiviert ist, wird eine Eingabezeile mit Autovervollständigungsunterstützung angezeigt, andernfalls wird ein Kombinationsfeld verwendet.
- **UUID-Generator**: Erzeugt ein schreibgeschütztes UUID-Feld (Universally Unique Identifiers), falls leer.

- **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.

  > **Bemerkung**
  >
  > Einige Layer, wie GeoPackage oder ESRI File Geodatabase, mit vordefinierten **Schlüssel-Felddomänen** werden von QGIS automatisch erkannt und erhalten ein **Werteabbildungs-Widget** für die entsprechenden Felder.

- **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.

  Wenn ein Layer, der in PostgreSQL, GeoPackage oder SpatiaLite gespeichert ist, so konfiguriert ist, dass er ein Wertebezieungs-Element verwendet, aber der benötigte Layer noch nicht im Projekt geladen ist, sucht QGIS automatisch nach dem Layer in derselben Datenbank/Verbindung.

12.1.10 Verknüpfungen

Mit der Registerkarte *Joins* können Sie Objekte des aktuellen Layers mit Objekten eines anderen geladenen Vektorlayers (oder einer Tabelle) verknüpfen. Die Verknüpfung basiert auf einem Attribut, das von den Layern in einer Eins-zu-Eins-Beziehung gemeinsam genutzt wird. Für weitere Details zu Verknüpfungen lesen Sie bitte *Verknüpfung von Objekten zwischen zwei Layern*.

12.1.11 Hilfsspeicher

Der übliche Weg zur Anpassung von Styling und Beschriftung ist die Verwendung von datendefinierten Eigenschaften, wie in *Einrichten von datendefinierter Übersteuerung* beschrieben. Dies ist jedoch möglicherweise nicht möglich, wenn die zugrunde liegenden Daten nur gelesen werden können. Außerdem kann die Konfiguration dieser datendefinierten Eigenschaften sehr zeitaufwendig oder nicht wünschenswert sein! Wenn Sie z. B. die Kartenwerkzeuge mit *Die Beschriftungswerkzeugleiste* vollständig nutzen wollen, müssen Sie mehr als 20 Felder in Ihrer ursprünglichen Datenquelle hinzufügen und konfigurieren (X- und Y-Position, Drehwinkel, Schriftart, Farbe usw.).

Der Mechanismus des Hilfsspeichers bietet die Lösung für diese Einschränkungen und umständlichen Konfigurationen. Hilfsfelder sind ein Umweg, um diese datendefinierten Eigenschaften (Beschriftungen, Diagramme, Symbole…) dank editierbarer Joins automatisch in einer SQLite-Datenbank zu verwalten und zu speichern. So können Sie Eigenschaften für Layer speichern, die nicht editierbar sind.

Im Dialogfeld für die Eigenschaften von Vektorebenen gibt es eine Registerkarte zur Verwaltung des Zusatzspeichers:

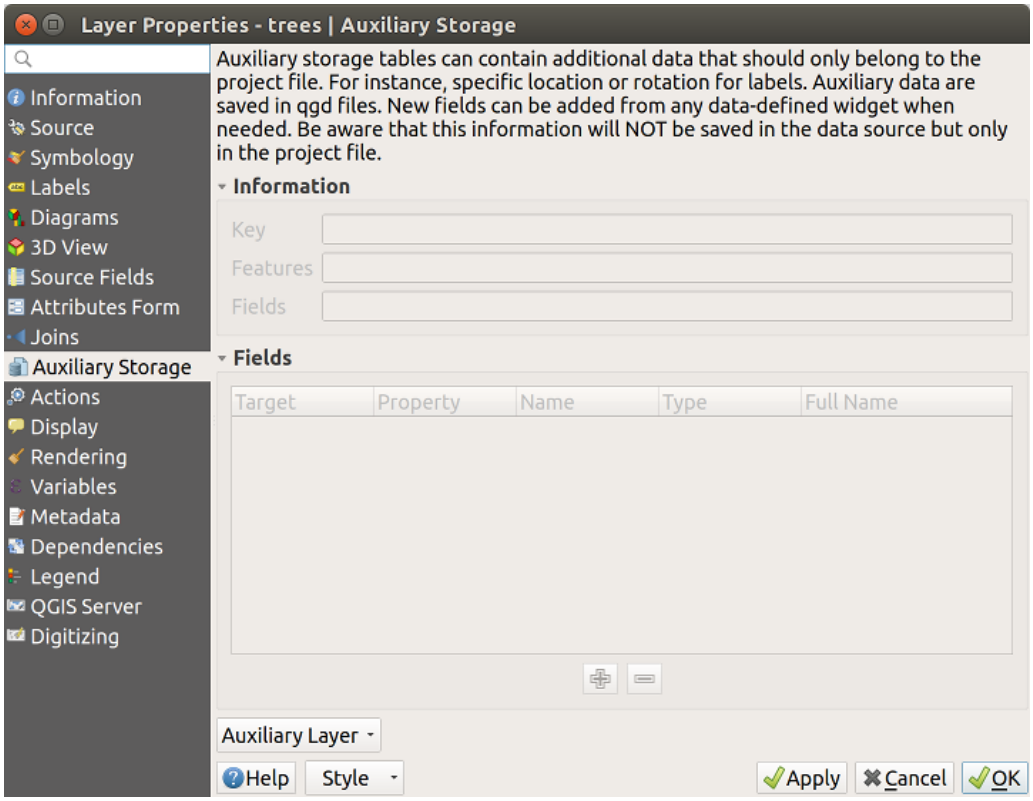


Abb. 12.48: Reiter Hilfsspeicher

Beschriftung

In Anbetracht der Tatsache, dass die Datenquelle dank datendefinierter Eigenschaften angepasst werden kann, ohne editierbar zu sein, sind die in *Die Beschriftungswerkzeugleiste* beschriebenen Beschriftungswerkzeuge immer verfügbar, sobald die Beschriftung aktiviert ist.

Eigentlich benötigt das Hilfsspeichersystem einen Layer, um diese Eigenschaften in einer SQLite-Datenbank zu speichern (siehe *Hilfsspeicher-Datenbank*). Der Erstellungsprozess wird ausgeführt, wenn Sie das erste Mal auf die Karte klicken, während ein Kartenbeschriftungswerkzeug aktiviert ist. Dann wird ein Fenster angezeigt, in dem Sie den Primärschlüssel für die Verknüpfung auswählen können (um sicherzustellen, dass die Objekte eindeutig identifiziert werden):

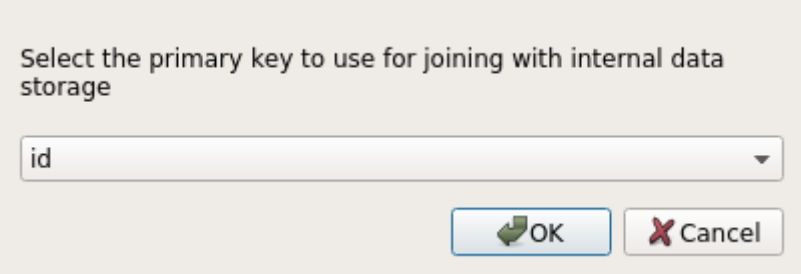


Abb. 12.49: Dialog zur Erstellung eines Hilfslayers

Sobald ein zusätzlicher Layer für die aktuelle Datenquelle konfiguriert ist, können Sie seine Informationen auf der Registerkarte abrufen:

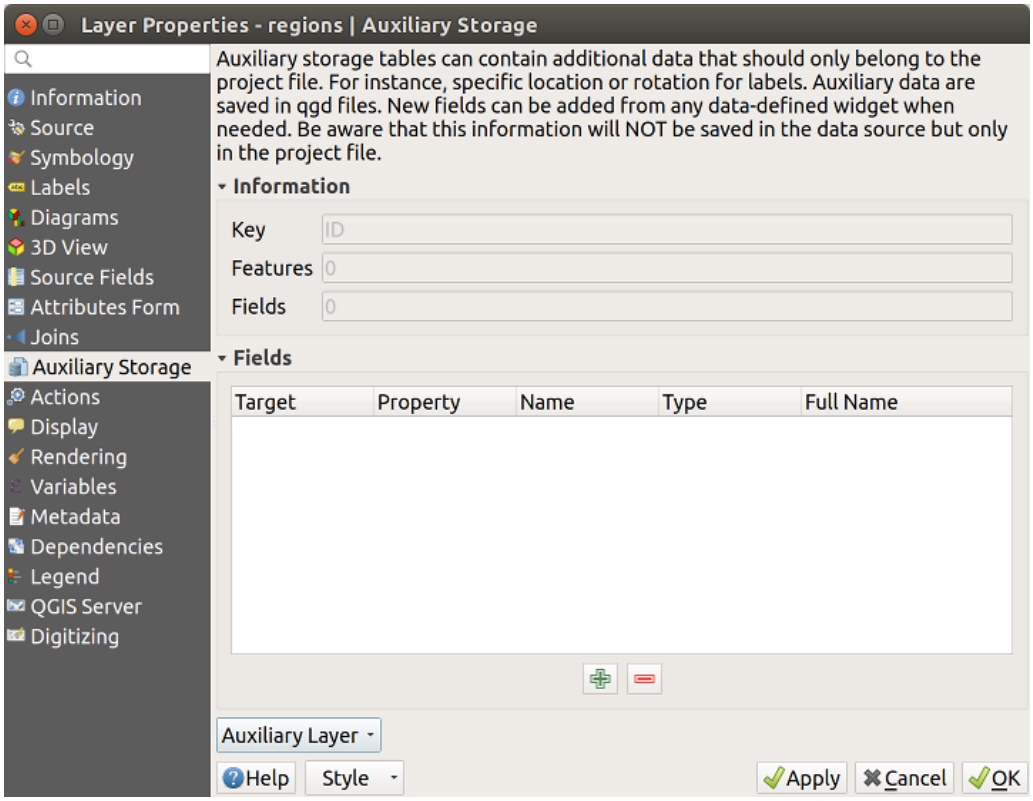


Abb. 12.50: Hilfslayer-Schlüssel

Der Hilfslayer hat nun diese Eigenschaften:

- Der Primärschlüssel ist `ID`,
- es gibt `0` Objekte, die ein Hilfsfeld verwenden,
- es gibt `0` Hilfsfelder.

Jetzt, da der Hilfslayer erstellt ist, können Sie die Beschriftungen der Layer bearbeiten. Klicken Sie auf eine Beschriftung, während das Kartenwerkzeug Beschriftungseigenschaften ändern aktiviert ist, dann können Sie Stil-Eigenschaften wie Größen, Farben usw. aktualisieren. Die entsprechenden datendefinierten Eigenschaften werden erstellt und können abgerufen werden:

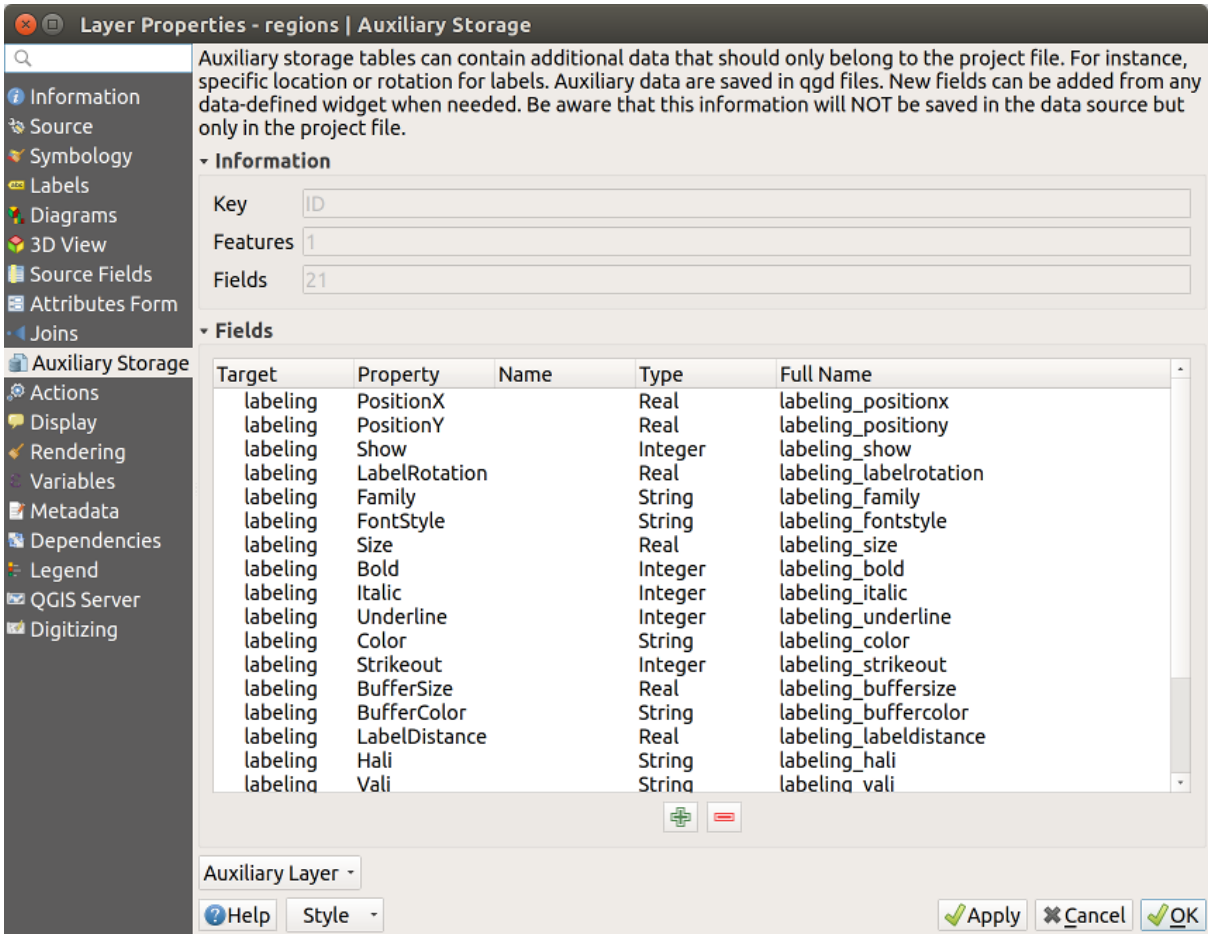


Abb. 12.51: Hilfsfelder

Wie Sie in der obigen Abbildung sehen können, werden `21` Felder automatisch erstellt und für die Beschriftung konfiguriert. Der Hilfsfeldtyp `FontStyle` ist zum Beispiel ein `String` und wird in der zugrundeliegenden SQLite-Datenbank `labeling_fontstyle` genannt. Es gibt auch `1` Objekt, das derzeit diese Hilfsfelder verwendet.

Beachten Sie, dass das Symbol auf der Registerkarte *Beschriftungen* angezeigt wird, was darauf hindeutet, dass die datendefinierten Überschreibungsoptionen korrekt eingestellt sind:

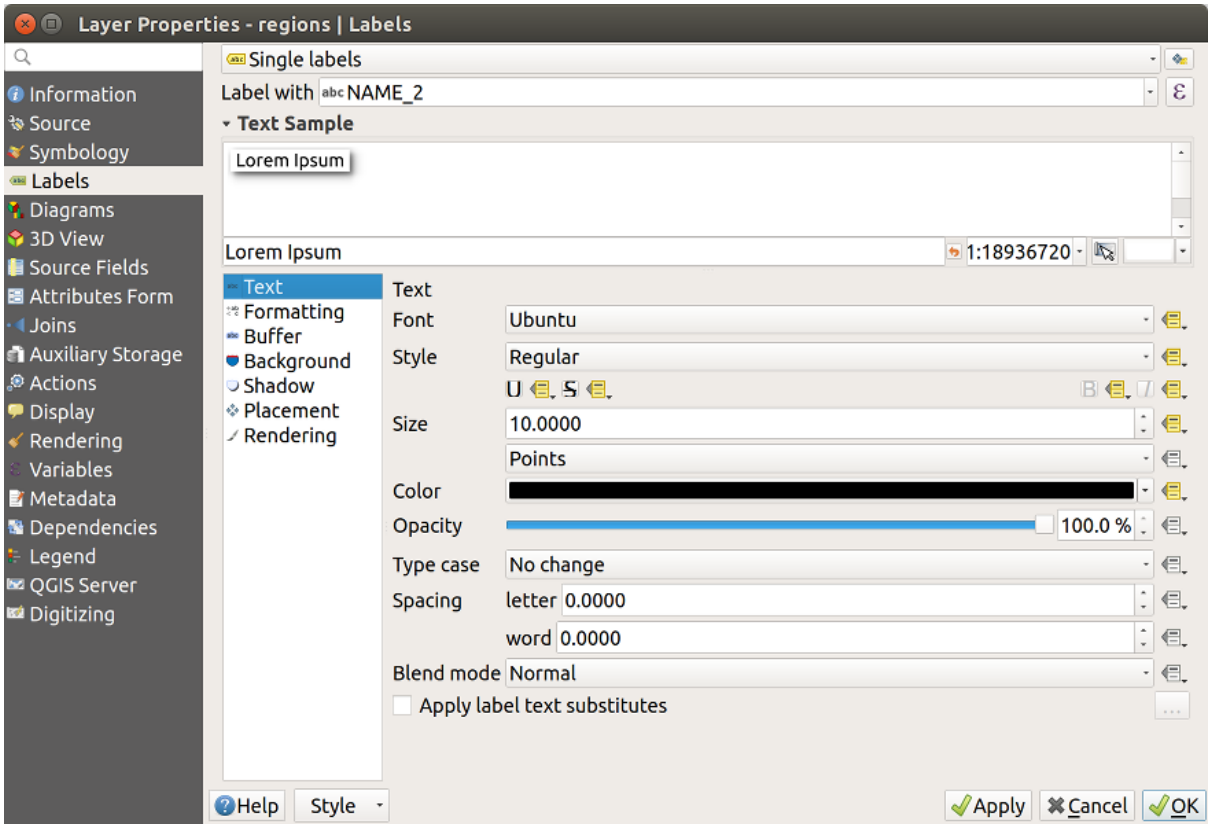


Abb. 12.52: Datendefinierte Eigenschaften werden automatisch erstellt

Andernfalls gibt es eine weitere Möglichkeit, ein Hilfsfeld für eine bestimmte Eigenschaft zu erstellen, und zwar über die Schaltfläche Datendefinierte Übersteuerung. Wenn Sie auf die Schaltfläche *Daten im Projekt speichern* klicken, wird automatisch ein Hilfsfeld für das Feld *Opazität* erstellt. Wenn Sie auf diese Schaltfläche klicken und der Layer noch nicht erstellt wurde, wird zunächst ein Fenster (Abb. 12.49) angezeigt, um den Primärschlüssel für die Verknüpfung auszuwählen.

Symbolisierung

Wie die oben beschriebene Methode zur Anpassung von Beschriftungen können auch Hilfsfelder zur Gestaltung von Symbolen und Diagrammen verwendet werden. Klicken Sie dazu auf Datendefinierte Übersteuerung und wählen Sie *Daten im Projekt speichern* für eine bestimmte Eigenschaft. Zum Beispiel das Feld *Füllfarbe*:

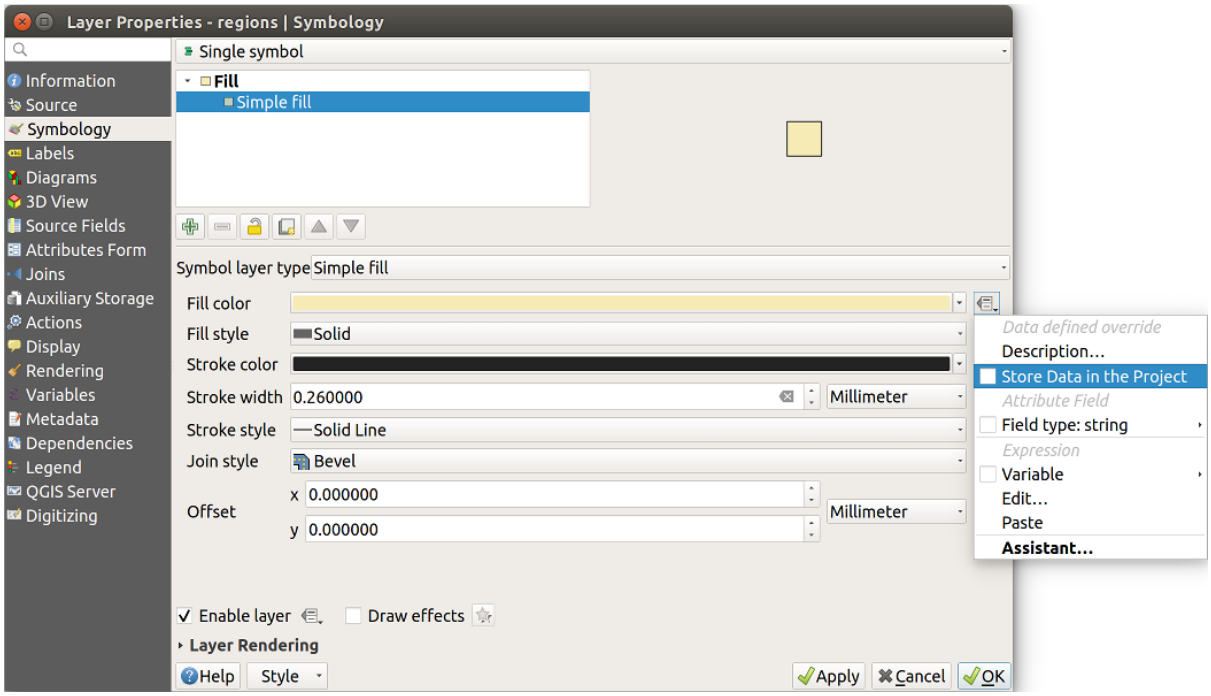


Abb. 12.53: Menü der datendefinierten Eigenschaft für die Symbolisierung

Für jedes Symbol gibt es unterschiedliche Attribute (z.B. Füllstil, Füllfarbe, Strichfarbe, etc…), so dass jedes Hilfsfeld, das ein Attribut darstellt, einen eindeutigen Namen benötigt, um Konflikte zu vermeiden. Nach Auswahl von *Daten im Projekt speichern* öffnet sich ein Fenster, das den *Typ* des Feldes anzeigt und Sie auffordert, einen eindeutigen Namen für das Hilfsfeld einzugeben. Wenn Sie zum Beispiel ein *Füllfarbe*-Hilfsfeld erstellen, öffnet sich das folgende Fenster:

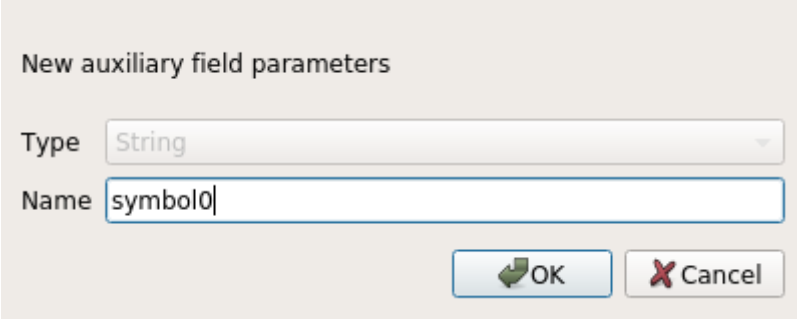


Abb. 12.54: Name des Hilfsfeldes für ein Symbol

Nach der Erstellung kann das Hilfsfeld auf der Registerkarte Hilfsspeicher abgerufen werden:

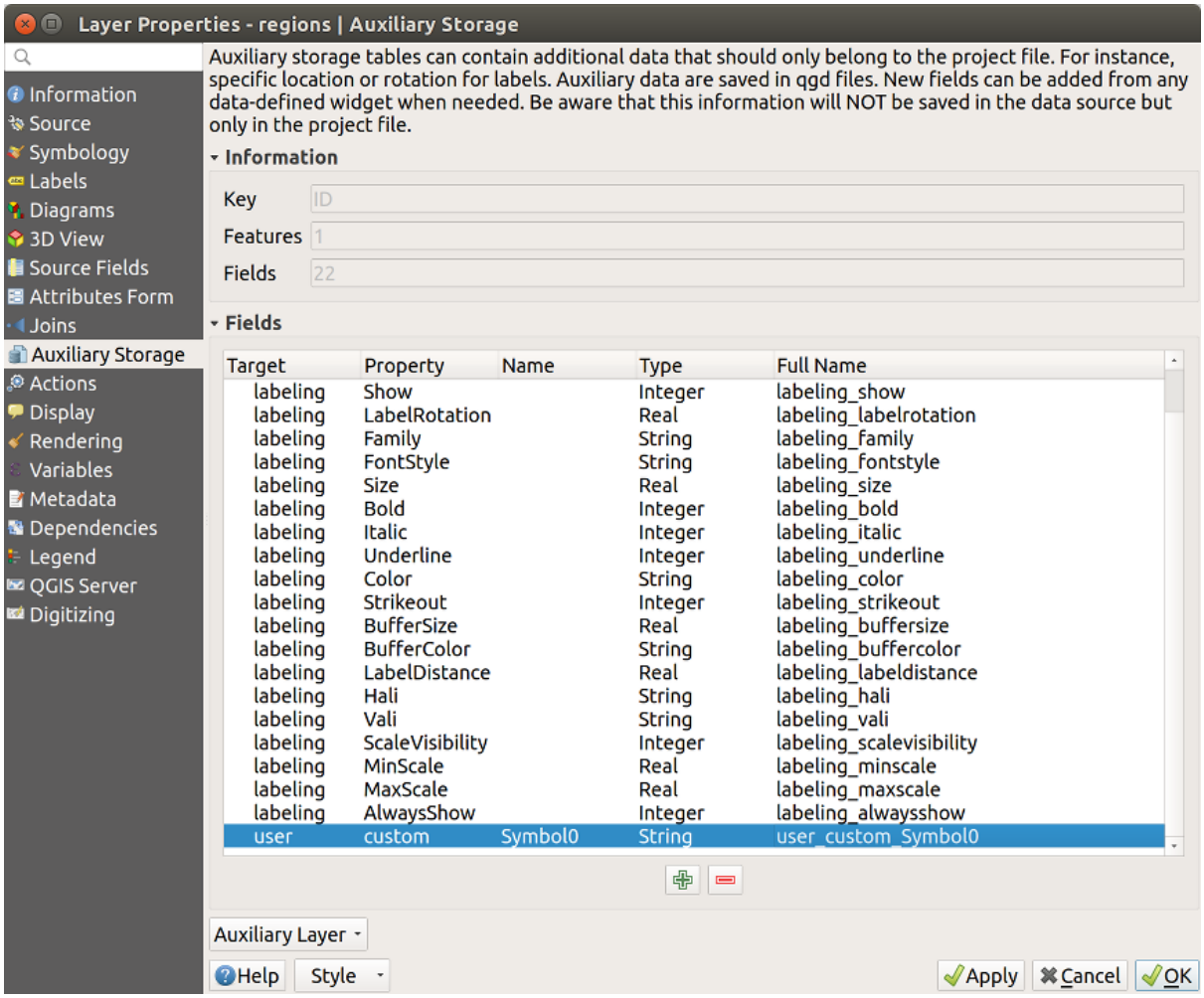


Abb. 12.55: Neues Hilfsfeld für das Symbol

Attributtabelle und Widgets

Hilfsfelder können über die *Attributtabelle* bearbeitet werden. Allerdings sind zunächst nicht alle Hilfsfelder in der Attributtabelle sichtbar.

Hilfsfelder, die Attribute der Symbologie, der Beschriftung, des Aussehens oder der Diagramme eines Layers darstellen, erscheinen automatisch in der Attributtabelle. Die Ausnahme sind Attribute, die mit der *Beschriftungswerkzeugleiste* geändert werden können, die standardmäßig ausgeblendet sind. Hilfsfelder, die eine `Farbe` darstellen, haben standardmäßig Bedienelement **Farbe** eingestellt, ansonsten wird für Hilfsfelder das Bedienelement **Text bearbeiten** verwendet.

Hilfsfelder, die Attribute darstellen, die mit der *Beschriftungswerkzeugleiste* geändert werden können, sind in der Attributtabelle standardmäßig **versteckt**. Um ein Feld sichtbar zu machen, öffnen Sie die Registerkarte *Attributformular* und ändern den Wert eines Hilfsfeldes *Bedienelementetyp* von **versteckt** auf einen anderen relevanten Wert. Ändern Sie zum Beispiel **auxiliary_storage_labeling_size** in **Texteditor** oder **auxiliary_storage_labeling_color** in das **Farbe** Bedienelement. Diese Felder werden nun in der Attributtabelle angezeigt.

Die Hilfsfelder in der Attributtabelle werden wie in der folgenden Abbildung dargestellt:

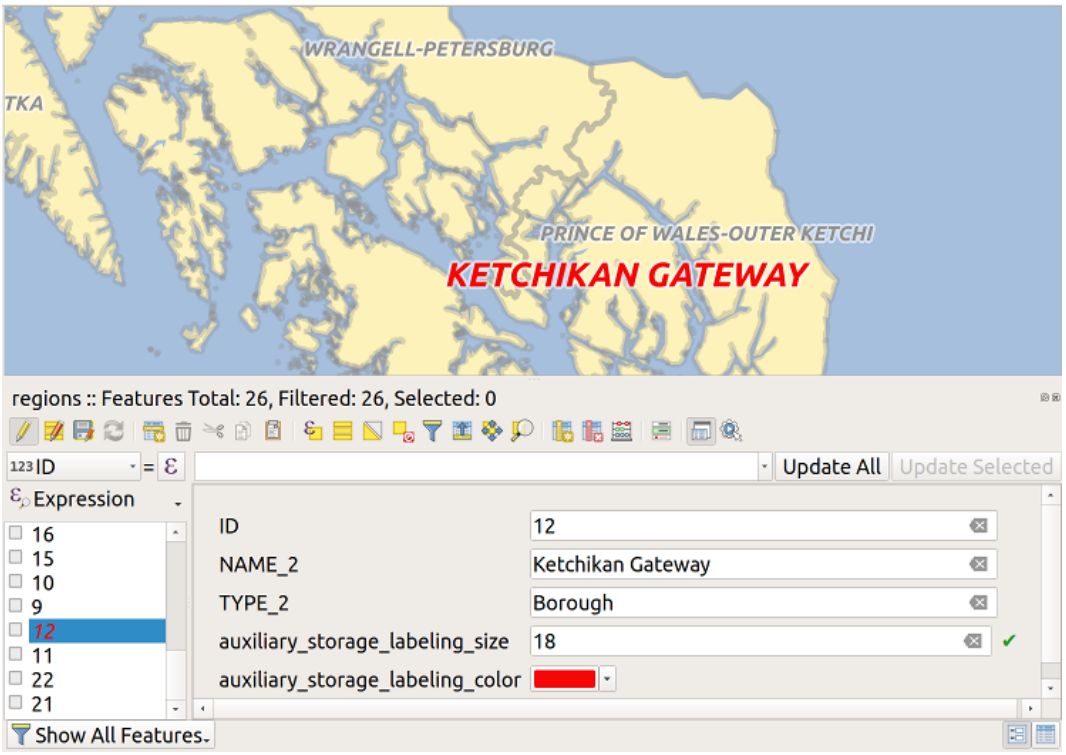


Abb. 12.56: Eingabemaske mit Hilfsfeldern

Management

Mit dem Menü *Hilfslayer* können Sie die Hilfsfelder verwalten:

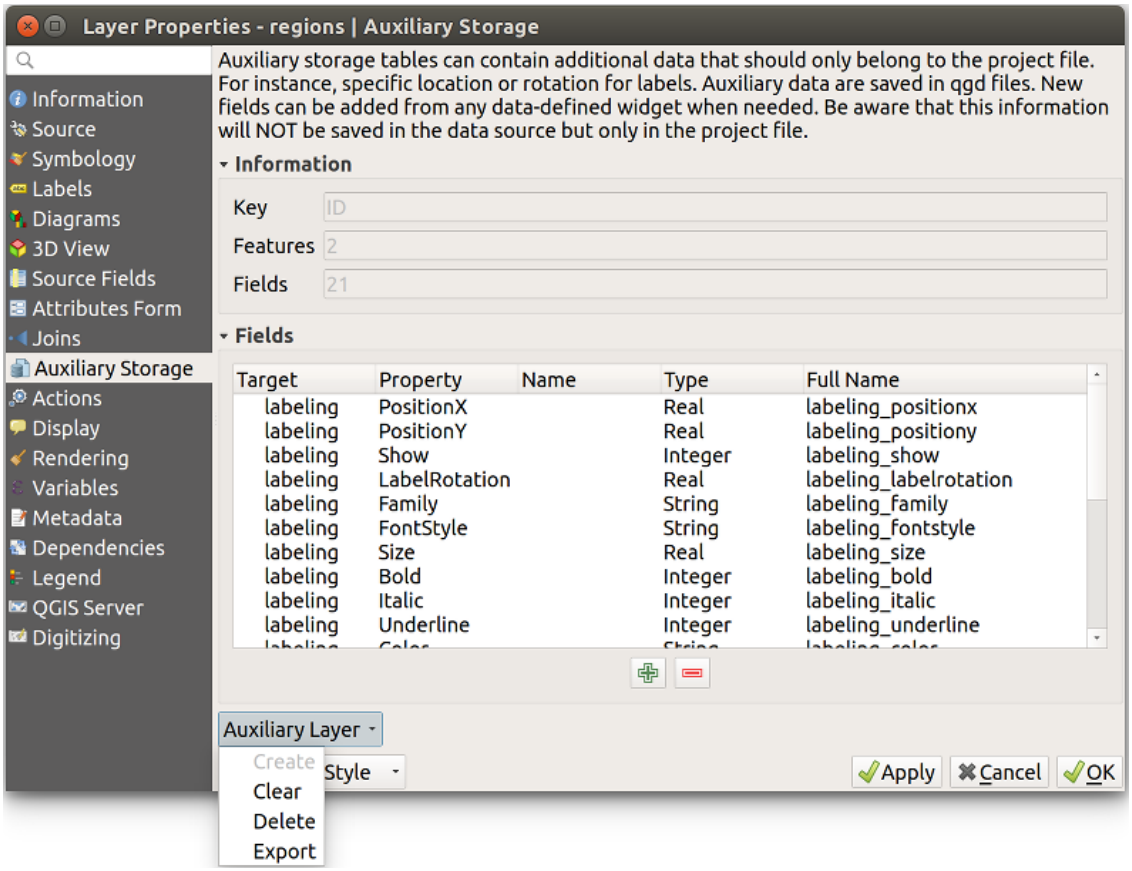


Abb. 12.57: Hilfslayer Management

Der erste Punkt *Erzeugen* ist in diesem Fall deaktiviert, da der zusätzliche Layer bereits erstellt wurde. Im Falle einer neuen Arbeit können Sie diese Aktion jedoch verwenden, um einen zusätzlichen Layer zu erstellen. Wie in *Beschriftung* erklärt, wird dann ein Primärschlüssel benötigt.

Die Aktion *Leeren* erlaubt es, alle Hilfsfelder zu behalten, aber ihren Inhalt zu entfernen. Auf diese Weise wird die Anzahl der Objekte, die diese Felder verwenden, auf `0` reduziert.

Mit der Aktion *Löschen* wird der zusätzliche Layer vollständig entfernt. Mit anderen Worten, die entsprechende Tabelle wird aus der zugrunde liegenden SQLite-Datenbank gelöscht und die Anpassung der Eigenschaften geht verloren.

Schließlich kann mit der Aktion *Export* der Hilfslayer als *Neuer Vektor-Layer* gespeichert werden. Beachten Sie, dass Geometrien nicht im Hilfsspeicher gespeichert werden. In diesem Fall werden die Geometrien jedoch auch aus der ursprünglichen Datenquelle exportiert.

Hilfsspeicher-Datenbank

Wenn Sie Ihr Projekt mit dem Format `.qgs` speichern, wird die SQLite-Datenbank, die als Hilfsspeicher verwendet wird, an der gleichen Stelle gespeichert, aber mit der Erweiterung `.qgd`.

Der Einfachheit halber kann stattdessen auch ein Archiv im `.qgz`-Format verwendet werden. In diesem Fall sind die Dateien `.qgd` und `.qgs` beide in das Archiv eingebettet.

12.1.12 Aktionen

Die Registerkarte *Aktionen* bietet die Möglichkeit, eine Aktion auf der Grundlage der Attribute eines Objekts auszuführen. Damit kann eine beliebige Anzahl von Aktionen durchgeführt werden, z. B. die Ausführung eines Programms mit Argumenten, die aus den Attributen eines Objekts gebildet werden, oder die Übergabe von Parametern an ein Web-Reporting-Tool.

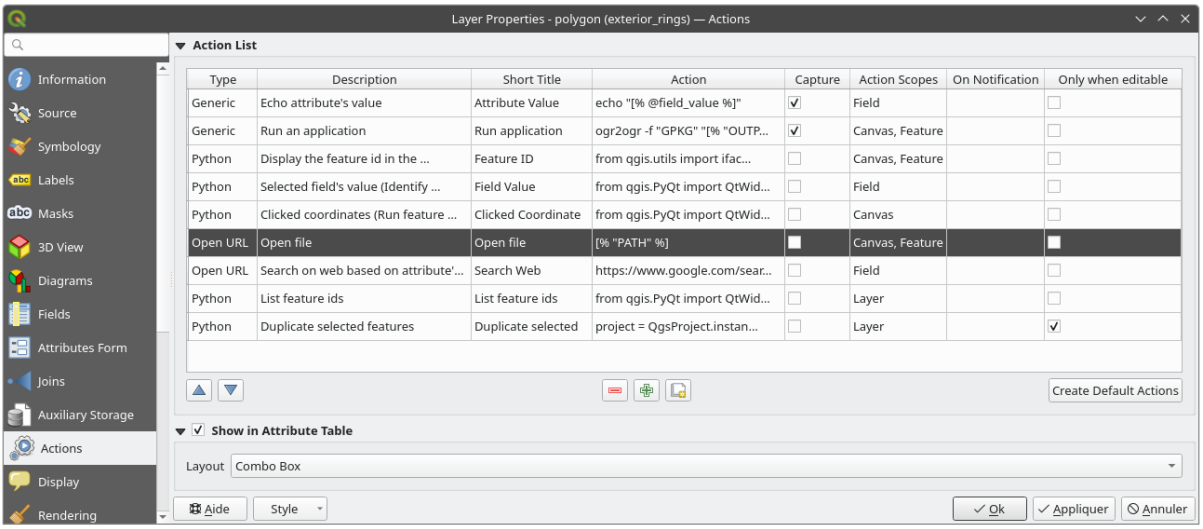


Abb. 12.58: Überblick über den Dialog Aktionen mit einigen Beispielaktionen

Aktionen sind nützlich, wenn Sie häufig eine externe Anwendung ausführen oder eine Webseite anzeigen möchten, die auf einem oder mehreren Werten in Ihrem Layer basiert. Es gibt verschiedene Arten von Aktionen, die wie folgt verwendet werden können:

- *Allgemein*, *macOS*, *Windows* und *Unix* Aktionen starten einen externen Prozess.
- *Python* Aktionen führen einen Python-Ausdruck aus.
- *Allgemein* und *Python* Aktionen sind überall sichtbar.
- *macOS*, *Windows* und *Unix* Aktionen sind nur auf der jeweiligen Plattform sichtbar (d. h. Sie können drei „Bearbeiten“-Aktionen definieren, um einen Editor zu öffnen, und die Benutzer können nur die eine „Bearbeiten“-Aktion für ihre Plattform sehen und ausführen, um den Editor zu starten).
- *URL öffnen*: Verwendet eine HTTP-GET-Anfrage zum Öffnen einer angegebenen URL.
- *Submit-URL (url-kodiert oder JSON)*: Wie die Aktion *URL öffnen*, aber unter Verwendung einer HTTP POST-Anfrage. Die Daten werden an eine URL gesendet, unter Verwendung von „application/x-www-form-urlencoded“ oder „application/json“, wenn der Body ein gültiges JSON ist.

  Ein Beispiel für einen Aktionsaufruf könnte sein:

  ```
  http://localhost:8000?/[% url_encode(map('file', 'index.php')) %]
  ```

- *Submit URL (mehrteilig)*: Entspricht der Aktion *URL öffnen*, verwendet aber eine HTTP POST-Anfrage. Die Daten werden unter Verwendung von „multipart/form-data“ an eine URL gesendet.

Im Dialog sind mehrere Beispiele enthalten. Sie können sie laden, indem Sie auf *Voreingestellte Aktionen erzeugen* klicken. Um eines der Beispiele zu bearbeiten, doppelklicken Sie auf die entsprechende Zeile. Ein Beispiel ist die Durchführung einer Suche auf der Grundlage eines Attributwerts. Dieses Konzept wird in der folgenden Diskussion verwendet.

Mit :guilabel: *In Attributtabelle anzeigen* können Sie im Dialog der Attributtabelle die ausgewählten Aktionen anzeigen lassen, entweder als *Kombinationsfeld* oder als *Einzelne Knöpfe* (siehe *Konfigurieren der Spalten*).

Aktionen definieren

Um eine Attribut-Aktion zu definieren, öffnen Sie den Vektor-Dialog *Layereigenschaften* und klicken auf die Registerkarte *Aktionen*. Hier klicken Sie auf Neue Aktion hinzufügen, um den Dialog *Aktion bearbeiten* zu öffnen.

Wählen Sie den Aktions- *Typ* und geben Sie einen beschreibenden Namen für die Aktion an. Die Aktion selbst muss den Namen der Anwendung enthalten, die ausgeführt wird, wenn die Aktion aufgerufen wird. Sie können ein oder mehrere Attributfeldwerte als Argumente für die Anwendung hinzufügen. Wenn die Aktion aufgerufen wird, wird jede Zeichenfolge, die mit einem `%` beginnt, gefolgt von dem Namen eines Feldes, durch den Wert dieses Feldes ersetzt. Die Sonderzeichen `%%` werden durch den Wert des Feldes ersetzt, das aus der Tabelle der Identifizierungsergebnisse oder Attribute ausgewählt wurde (siehe *Aktionen anwenden*). Doppelte Anführungszeichen können verwendet werden, um Text zu einem einzigen Argument für das Programm, Skript oder den Befehl zusammenzufassen. Doppelte Anführungszeichen werden ignoriert, wenn ihnen ein Backslash vorangestellt ist.

Aktionen können nur einen einzelnen Prozess mit Argumenten aufrufen, daher funktionieren boolesche Operatoren (wie `&`, `&&`, `;`, `|`) nicht. In UNIX-ähnlichen Betriebssystemen können mehrere Befehle über `bash -c` ausgeführt werden.

Mit *Aktionsbereich* können Sie festlegen, wo die Aktion verfügbar sein soll. Sie haben folgende Auswahlmöglichkeiten:

1. *Feld*: Die Aktion ist verfügbar, wenn Sie mit der rechten Maustaste auf die Zelle in der Attributtabelle, im Objektformular und in der Standardaktionsschaltfläche der Hauptsymbolleiste klicken.
2. *Objekt*: Die Aktion ist verfügbar, wenn Sie mit der rechten Maustaste in die Zelle der Attributtabelle klicken.
3. *Karte*: Die Aktion ist über die Hauptaktionsschaltfläche in der Symbolleiste verfügbar.
4. *Formular*: Die Aktion ist nur in einem Objekt-Formular verfügbar, das mit dem *Drag-and-Drop* Modus entworfen wurde.
5. *Layer*: Die Aktion ist über die Aktionsschaltfläche in der Symbolleiste der Attributtabelle verfügbar. Beachten Sie, dass diese Art von Aktion den gesamten Layer und nicht die einzelnen Objekte betrifft.

Wenn Sie Feldnamen vorfinden die Substrings anderer Feldnamen sind (z.B., `col1` und `col10`) sollten Sie das angeben, indem Sie den Feldnamen (und den % Buchstaben) mit eckigen Klammern umrunden (z.B. `[%col10]`). Hiermit wird vermieden, dass der Feldname `%col10` mit dem Feldnamen `%col1` mit einem `0` am Ende verwechselt wird. Die Klammern werden von QGS entfernt, wenn es im Wert des Feldes ersetzt. Wenn Sie wollen, dass das zu ersetzende Feld von eckigen Klammern umrandet wird, verwenden Sie ein zweites Set wie hier: `[[%col10]]`.

Wenn Sie das *Objekte abfragen* Werkzeug verwenden, können Sie den *Identifikationsergebnis* Dialog öffnen. Es enthält eine *(Abgeleitet)* Item das layertyprelevante Informationen enthält. Die Werte in diesem Element können auf ähnliche Weise den anderen Feldern zugeordnet werden indem dem Abgeleitet Feldnamen ein `(Derived)` vorangeht. Zum Beispiel hat ein Punktlayer ein `X` und `Y` Feld und die Werte dieser Felder können in der Aktion mit `%(Derived).X` und `%(Derived).Y` verwendet werden. Die Abgeleitet Attribute sind nur in der *Objekte abfragen* Dialog Box erhältlich, jedoch nicht in der *Attributtabelle* Dialogbox.

Nachfolgend werden zwei Beispielaktionen gezeigt:

- `konqueror https://www.google.com/search?q=%nam`
- `konqueror https://www.google.com/search?q=%%`

Im ersten Beispiel wird der Webbrowser konqueror aufgerufen und eine URL zum Öffnen übergeben. Die URL führt eine Google-Suche nach dem Wert des Feldes `nam` aus unserem Vektor Layer durch. Beachten Sie, dass die Anwendung oder das Skript, das durch die Aktion aufgerufen wird, im Pfad enthalten sein muss, oder Sie müssen den vollständigen Pfad angeben. Um sicherzugehen, könnten wir das erste Beispiel wie folgt umschreiben: `/opt/kde3/bin/konqueror https://www.google.com/search?q= %nam`. Dadurch wird sichergestellt, dass die Konqueror-Anwendung ausgeführt wird, wenn die Aktion aufgerufen wird.

Das zweite Beispiel nutzt den Ausdruck %%, welcher unabhängig ist von einem speziellen Feld. Beim Ausführen des Befehls wird der Ausdruck %% durch den Wert des jeweils selektierten Feldes aus *Objekte abfragen* oder der Attributtabelle ersetzt.

Aktionen duplizieren

QGIS ermöglicht es Ihnen, vorhandene Aktionen zu duplizieren. Um eine Attribut-Aktion zu duplizieren, öffnen Sie den Vektor-Dialog *Layereigenschaften* und klicken Sie auf die Registerkarte *Aktionen*. Klicken Sie in der Registerkarte *Aktionen* auf Aktion duplizieren, um den Dialog *Aktion duplizieren* zu öffnen. Sie müssen mindestens eine vorhandene Aktion ausgewählt haben, um ein Duplikat erstellen zu können.

Nehmen Sie im angezeigten Dialogfeld die erforderlichen Änderungen vor. Weitere Informationen finden Sie unter *Aktionen definieren*. Wenn Sie fertig sind, klicken Sie auf *OK*, um eine Kopie der Aktion mit den vorgenommenen Änderungen zu erstellen. Wenn Sie die Beschreibung nicht bearbeitet haben oder wenn Sie sie so geändert haben, dass sie mit der Beschreibung einer anderen vorhandenen Aktion identisch ist, wird am Ende „_1“ hinzugefügt.

Aktionen anwenden

QGIS bietet viele Möglichkeiten, um Aktionen auszuführen, die Sie auf einem Layer aktiviert haben. Abhängig von ihren Einstellungen können sie verfügbar sein:

- im Dropdown-Menü von Funktionsaktion ausführen aus der *Attribute-Symbolleiste* oder dem *Attribut-Tabelle*-Dialog;
- beim Rechtsklick auf ein Objekt mit dem Werkzeug Objekte abfragen (siehe *Objekte abfragen* für weitere Informationen);
- aus dem Bedienfeld *Identifikationsergebnis* unter dem Abschnitt *Aktionen*;
- als Elemente einer Spalte *Aktionen* im Dialog der *Attributtabelle*.

Wenn Sie eine Aktion mit `%%` Notation verwenden, machen Sie einen Rechtsklick auf den Feldwert im *Identifikationsergebnis* Dialog oder im *Attributtabelle* Dialog den Sie der Anwendung oder dem Skript übergeben wollen.

In einem weiteren Beispiel soll gezeigt werden, wie Attributwerte eines Vektorlayers abgefragt und in eine Textdatei mithilfe der Bash und des `echo` Kommandos geschrieben werden (funktioniert also nur unter und evtl.). Der Abfragelayer enthält die Felder Art `taxon_name`, Latitude `lat` und Longitude `long`. Wir möchten jetzt eine räumliche Selektion von Örtlichkeiten machen und diese Feldwerte in eine Textdatei für den ausgewählten Datensatz (in der QGIS Kartenansicht in gelb gezeigt) exportieren. Hier ist die Aktion, um dies zu erreichen:

```
bash -c "echo \"%taxon_name %lat %long\" >> /tmp/species_localities.txt"
```

Nachdem ein paar Orte auf dem Bildschirm ausgewählt wurden (diese erscheinen gelb hinterlegt), starten wir die Aktion mit der rechten Maustaste über den Dialog *Abfrageergebnisse* und können danach in der Textdatei die Ergebnisse ansehen:

```
Acacia mearnsii -34.0800000000 150.0800000000
Acacia mearnsii -34.9000000000 150.1200000000
Acacia mearnsii -35.2200000000 149.9300000000
Acacia mearnsii -32.2700000000 150.4100000000
```

Als Übung können wir eine Aktion erstellen, die eine Google-Suche auf dem Layer `Seen` durchführt. Zunächst müssen wir die URL ermitteln, die für eine Suche nach einem Schlüsselwort erforderlich ist. Dies ist leicht zu bewerkstelligen, indem Sie Google aufrufen, eine einfache Suche durchführen und dann die URL aus der Adressleiste Ihres Browsers abrufen. Nach dieser kleinen Anstrengung sehen wir, dass das Format https://www.google.com/search?q=QGIS ist, wobei `QGIS` der Suchbegriff ist. Mit diesen Informationen ausgestattet, können wir fortfahren:

1. Laden Sie den Layer file:*lakes.shp*.
2. Öffnen Sie den Dialog *Layereigenschaften*, indem Sie einen Doppelklick auf den Layer in der Legende machen oder aus dem Kontextmenü (Rechtsklick) des Layers *Eigenschaften…* auswählen.
3. Klicken Sie auf den Reiter *Aktionen*.

4. Klicken Sie auf Neue Aktion hinzufügen.
5. Wählen Sie den Aktionstyp *URL öffnen*,
6. Geben Sie einen Namen für die Aktion ein, z.B. `Google Search`.
7. Zusätzlich können Sie ein *Kurznamen* oder auch ein *Icon* hinzufügen.
8. Wählen Sie den *Aktionsbereich*. Siehe *Aktionen definieren* für weitere Informationen. Belassen Sie die Standardeinstellungen für dieses Beispiel.
9. Für die Aktion fügen Sie die URL ein, die für eine Google-Suche verwendet wird, bis zum, aber nicht einschließlich des Suchbegriffs: `https://www.google.com/search?q=`.
10. Der Text im Feld *Aktion* sollte nun wie folgt aussehen:

```
https://www.google.com/search?q=
```

11. Klicken Sie auf das Dropdown-Feld mit den Feldnamen für den Layer `lakes`. Es befindet sich direkt links neben der Schaltfläche *Einfügen*.
12. Wählen Sie in der Dropdown-Box *NAMES* und klicken Sie auf *Einfügen*.
13. Ihr Aktionstext sieht nun wie folgt aus:

```
https://www.google.com/search?q=[%NAMES%]
```

14. Um die Aktion fertigzustellen und hinzuzufügen, klicken Sie auf die Schaltfläche *OK*.

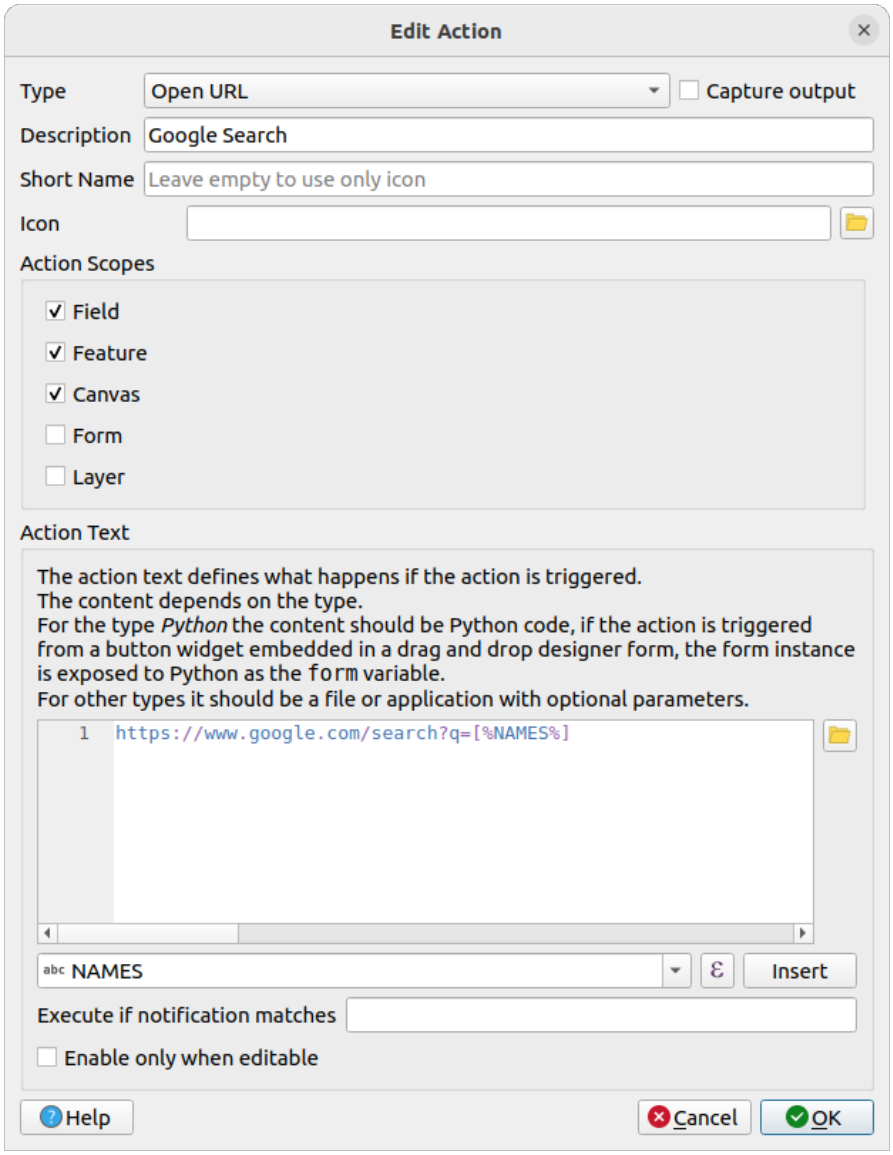


Abb. 12.59: Dialogfeld Neue Aktion hinzufügen

Damit ist der Vorgang abgeschlossen, und die Aktion ist einsatzbereit.

Schließen Sie das Dialogfeld *Layereigenschaften* und zoomen Sie auf ein Gebiet von Interesse. Vergewissern Sie sich, dass der Layer `lakes` aktiv ist und identifizieren Sie einen See. In den Identifikationsergebnissen sehen Sie nun, dass die erstelle Aktion sichtbar ist:

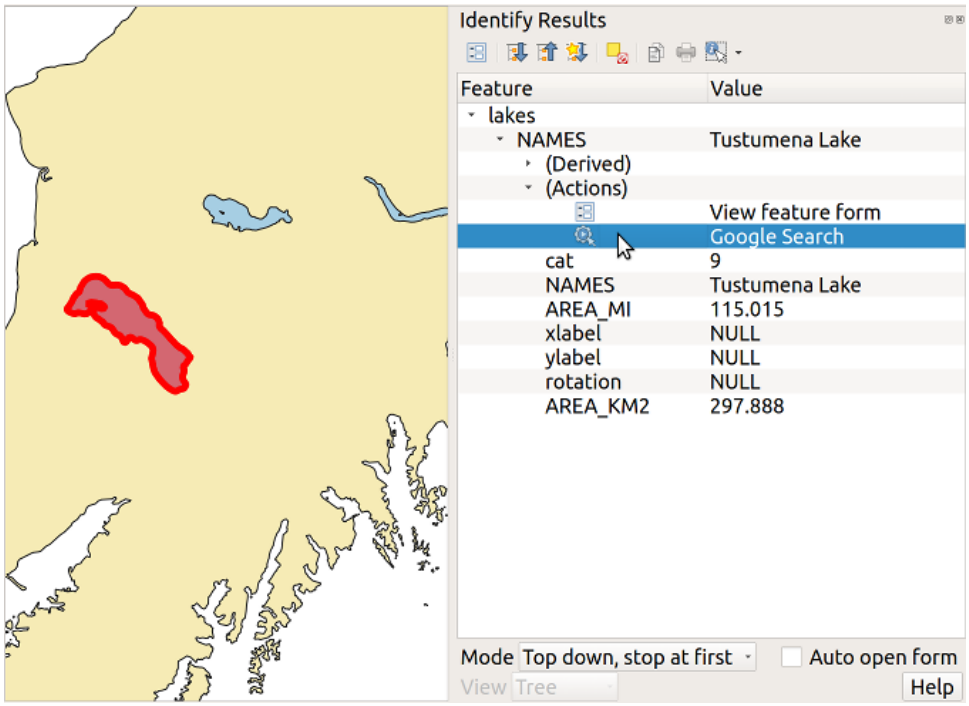


Abb. 12.60: Eine benutzerdefinierte Aktion in den Identifikationsergebnissen

Wenn wir auf die Aktion klicken, wird unser Standardbrowser aufgerufen und die URL https://www.google.com/search?q=Tustumena aufgerufen. Es ist auch möglich, der Aktion weitere Attributfelder hinzuzufügen. Dazu können Sie ein + an das Ende des Aktionstextes anhängen, ein weiteres Feld auswählen und auf *Feld einfügen* klicken. In diesem Beispiel gibt es einfach kein anderes Feld, nach dem zu suchen sinnvoll wäre.

Sie können auch mehrere Aktionen für einen Layer definieren. Sie alle werden dann bei der Abfrage von Objekten im *Identifikationsergebnis* Dialog angezeigt.

Sie können auch die aus der Attributtabelle aufrufen, indem Sie eine Zeile mit der rechten Maustaste auswählen und dann die Aktion aus dem Pop-up-Menü wählen.

Sie sehen, man kann sich eine Vielzahl interessanter Aktionen ausdenken. Wenn Sie z.B. einen Punktlayer mit einzelnen Punkten haben, an denen Photos geschossen wurden, dann können Sie eine Aktion erstellen, über die Sie dann das entsprechende Foto anzeigen lassen können, wenn Sie auf den Punkt in der Karte klicken. Man kann auch zu bestimmten Attributen webbasierte Information ablegen (z.B. in einer HTML-Datei) und diese dann über eine Aktion anzeigen lassen, etwa so wie in dem Google Beispiel.

Wir können auch komplexere Beispiele erstellen, indem wir z.B. **Python** Aktionen verwenden.

Wenn wir eine Aktion zum Öffnen einer Datei mit einer externen Anwendung erstellen, können wir in der Regel absolute Pfade oder auch relative Pfade verwenden. Im zweiten Fall ist der Pfad relativ zum Speicherort der ausführbaren Datei des externen Programms. Aber was ist, wenn wir relative Pfade verwenden müssen, relativ zum ausgewählten Layer (einem dateibasierten, wie Shapefile oder SpatiaLite)? Mit dem folgenden Code ist das kein Problem:

```
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 ] )
```

Wir müssen uns nur ins Gedächtnis rufen dass es sich um eine *Python* Aktion handelt und dass das Ändern der *command* und *imagerelpath* Variablen auf unsere Bedürfnisse angepasst wird.

Was aber wenn der relative Pfad relativ zur (gespeicherten) Projektdatei sein muss? Der Code der Python Aktion würde wie folgt lauten:

```
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 ] )
```

Ein anderes Python Aktion Beispiel ist das mit wir dem Projekt neue Layer hinzufügen können. Z.B. wird in den folgenden Beispielen dem Projekt ein Vektorlayer beziehungsweise ein Rasterlayer hinzugefügt. Die Namen der Dateien, die dem Projekt hinzugefügt werden sollen, und die Namen, die den Layern gegeben werden, sind datengesteuert (*filename* und *layername* sind Spaltennamen der Attributtabelle des Vektorlayers in dem die Aktion erstellt wurde):

```
qgis.utils.iface.addVectorLayer('/yourpath/[% "filename" %].shp',
  '[% "layername" %]', 'ogr')
```

Um eine Rasterdatei hinzuzufügen (ein TIF-Bild in diesem Beispiel) wird daraus:

```
qgis.utils.iface.addRasterLayer('/yourpath/[% "filename" %].tif',
  '[% "layername" %]')
```

12.1.13 Anzeige

Die Registerkarte *Anzeigen* hilft Ihnen bei der Konfiguration von Feldern, die zur Identifizierung von Objekten verwendet werden:

- Der *Anzeigename*: basiert auf einem Feld oder einem *Ausdruck*. Standardmäßig wird dies auf das erste Feld im Layer gesetzt, wenn kein Feld existiert, in dem `<name>` vorkommt. Dies wird verwendet als:
  - die Bezeichnung, die über den Objektinformationen in den Objekte abfragen-Ergebnissen angezeigt wird
  - das Feld, das in der *Suchen-Leiste* bei der Suche nach Objekten in allen Layern verwendet wird
  - die Kennung des Objekts in der *Formularansicht* der Attributtabelle
  - die Kennung des Objekts, wenn die Karte oder das Layout in ein Ausgabeformat mit Layern wie Geospatial PDF exportiert wird
  - die Kartentipp-Informationen, d.h. die Meldung, die in der Kartenansicht angezeigt wird, wenn der Mauszeiger über ein Objekt des aktiven Layers bewegt wird und das Symbol Kartenhinweise gedrückt ist. Anwendbar, wenn *Kartentipps einschalten* aktiv ist und kein *HTML-Kartenhinweis* gesetzt ist.
- *Kartentipps einschalten* steuert, ob Kartentipps für den Layer angezeigt werden sollen
- Die *HTML-Kartenhinweis* bietet einen komplexen und vollständigen HTML-Texteditor für Kartenhinweise, der QGIS-Ausdrücke und HTML-Stile und -Tags (mehrzeilig, Schriftarten, Bilder, Hyperlinks, Tabellen usw.) kombiniert. Sie können das Ergebnis Ihres Code-Beispiels im *Vorschau*-Fenster überprüfen (auch praktisch für die Vorschau der *Anzeigename*-Ausgabe). Darüber hinaus können Sie vorhandene Ausdrücke mit der Schaltfläche *Ausdruck einfügen/bearbeiten* auswählen und bearbeiten.

Bemerkung

Verhalten der Schaltfläche „Ausdruck einfügen/bearbeiten“ verstehen

Wenn Sie Text innerhalb eines Ausdrucks (zwischen „[%“ und „%]“) auswählen oder wenn kein Text ausgewählt ist, sich der Cursor jedoch innerhalb eines Ausdrucks befindet, wird automatisch der gesamte Ausdruck zur Bearbeitung ausgewählt. Befindet sich der Cursor oder ausgewählter Text außerhalb eines Ausdrucks, wird das Dialogfeld mit der Auswahl geöffnet.

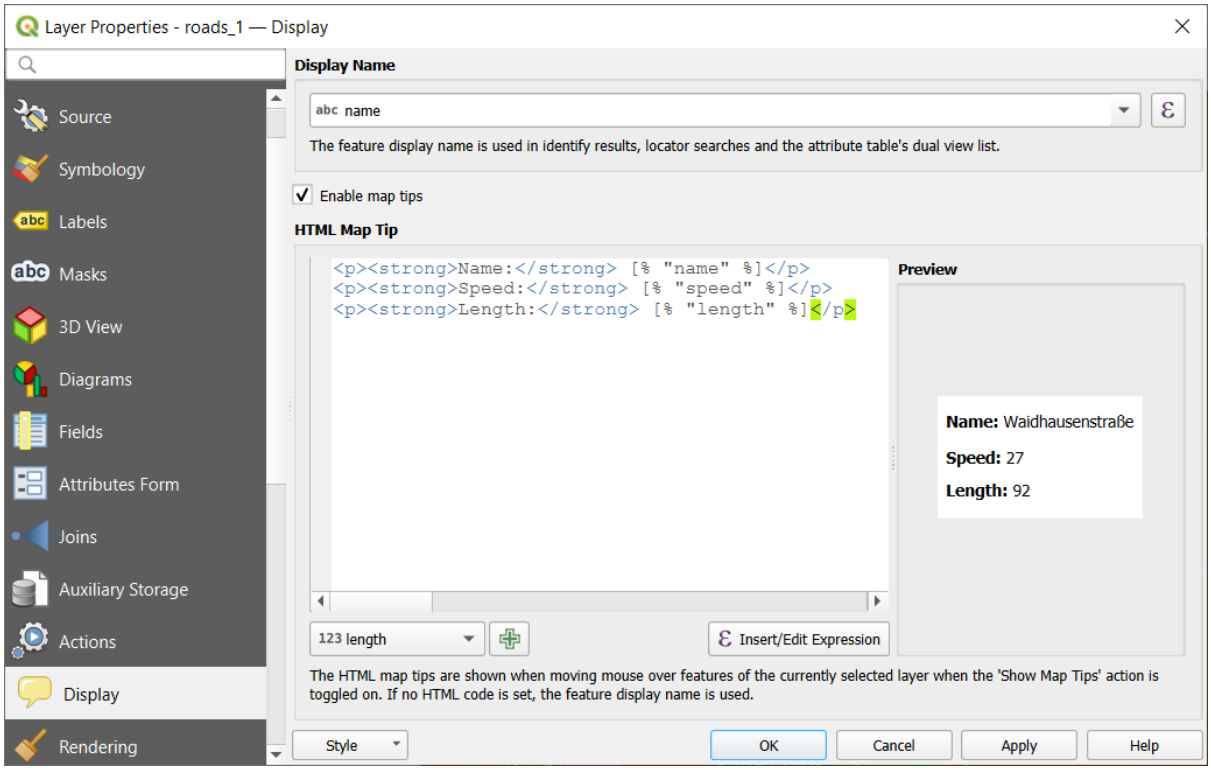


Abb. 12.61: HTML-Code für Kartenhinweis

So zeigen Sie Kartentipps an:

1. Wählen Sie den Menüpunkt *Ansicht* ► *Kartenhinweise anzeigen* oder klicken Sie auf das Symbol Kartenhinweise anzeigen in der *Attributwerkzeugleiste*.
2. Vergewissern Sie sich, dass der Layer, auf den Sie abzielen, aktiv ist und dass die Eigenschaft *Kartentipps anzeigen* aktiviert ist.
3. Bewegen Sie den Mauszeiger über ein Objekt, und die entsprechenden Informationen werden eingeblendet.

Der Kartentipp ist ein ebenenübergreifendes Objekt, d. h., wenn er einmal aktiviert ist, bleibt er eingeschaltet und gilt für alle mit dem Kartentipp versehenen Layer im Projekt, bis er wieder ausgeschaltet wird.

Abb. 12.62: Kartenhinweis erstellt mit HTML-Code

12.1.14 Darstellung

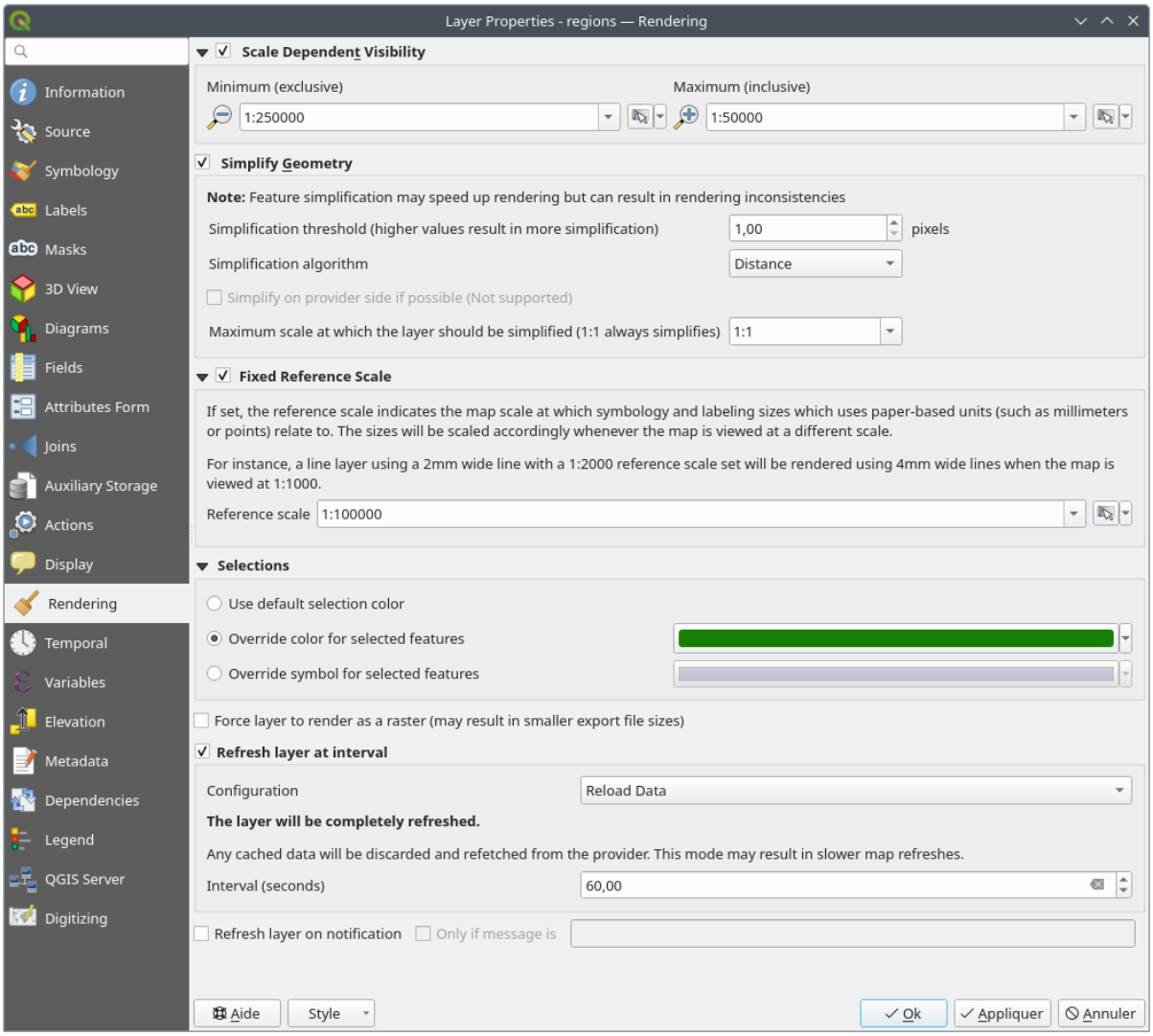


Abb. 12.63: Der Darstellung-Reiter

Der Reiter *Darstellung* bietet folgende Eigenschaften:

- Unter *Maßstabsabhängige Sichtbarkeit* können Sie die Maßstäbe *Maximum (inklusive)* und *Minimum (exklusive)* festlegen und damit einen Bereich von Maßstäben definieren, in dem die Objekte des Layers sichtbar sind. Außerhalb dieses Bereichs werden sie ausgeblendet. Die Schaltfläche Auf aktuellen Kartenmaßstab setzen hilft Ihnen, den aktuellen Maßstab der Kartenansicht als Grenze für den Sichtbarkeitsbereich zu verwenden. Siehe *Einstellen des Sichtbarkeitsmaßstabs* für weitere Informationen.

  Bemerkung

  Sie können die skalierungsabhängige Sichtbarkeit eines Layers auch über das Bedienfeld *Layer* aktivieren: Klicken Sie mit der rechten Maustaste auf den Layer und wählen Sie im Kontextmenü *Maßstabsabhängige Sichtbarkeit des Layers setzen*.

- QGIS bietet Unterstützung für die on-the-fly Generalisierung von Objekten. Dies kann die Renderingzeiten beim Zeichnen vieler komplexer Objekte in kleinen Maßstäben verbessern. Dieses Objekt kann in den Layer-

Einstellungen mit der Option *Geometrie vereinfachen* aktiviert oder deaktiviert werden. Es gibt auch eine globale Einstellung, die die Generalisierung standardmäßig für neu hinzugefügte Layer aktiviert (siehe *Globale Vereinfachung* für weitere Informationen).

> **Bemerkung**
>
> Objektgeneralisierung, kann in einigen Fällen, Artefakte in Ihre gerenderte Ausgabe einführen. Dies können Splitter zwischen Polygonen und ungenau Darstellung bei der Verwendung von Offset-basierten Symbolayern sein.

- Mit *Festgelegter Referenzmaßstab* können Sie einen Kartenmaßstab angeben, auf den sich Symbologie- und Beschriftungsgrößen beziehen, die papierbasierte Einheiten (wie Millimeter oder Punkte) verwenden. Die Größen werden entsprechend skaliert, wenn die Karte in einem anderen Maßstab betrachtet wird.

  Ein Layer mit 2 mm breiten Linien und einem Maßstab von 1:2.000 *Referenzmaßstab* wird beispielsweise mit 4 mm breiten Linien dargestellt, wenn die Karte im Maßstab 1:1.000 betrachtet wird.

- Mit der Gruppe *Auswahlen* können Sie steuern, ob eine bestimmte Farbe oder ein bestimmtes Symbol anstelle der Standardeinstellungen (*Projekteigenschaften* ► *Allgemein* ► *Auswahlfarbe*) für einen bestimmten Layer verwendet werden soll. Dies ist nützlich, um die Sichtbarkeit von ausgewählten Objekten mit bestimmten Symbolen zu verbessern:

  – *Vorgabeauswahlfarbe verwenden*

  – *Farbe für gewählte Objekte übersteuern*: z. B. wenn der Layer standardmäßig eine gelbe Farbe verwendet und die gelbe Standardauswahl nicht sichtbar ist.

  – *Symbol für gewählte Objekte übersteuern*: Wenn z. B. ein Layer mit Linien ein dünnes Symbol verwendet und die Linien durch Einfärben nicht ausreichend sichtbar werden, kann es hilfreich sein, das Symbol durch eine dickere Linie zu überschreiben. Wenn der Layer eine Rastersymbolik oder Farbverlaufsfüllungen/Linien/Shapeburst mit Farbrampen-Symbolik verwendet, wird die Standard-Auswahlfarbe nicht angewendet; die Möglichkeit, ein bestimmtes einfacheres Symbol für ausgewählte Objekte im Layer festzulegen, kann helfen.

- Das Rendern von extrem detaillierten Layern (z. B. Polygon-Layern mit einer großen Anzahl von Knoten) kann dazu führen, dass Layout-Exporte im PDF/SVG-Format sehr groß werden, da alle Knoten in die exportierte Datei aufgenommen werden. Dies kann auch dazu führen, dass sich die resultierende Datei nur sehr langsam in anderen Programmen bearbeiten/öffnen lässt.

  Aktivieren von *Darstellung des Layers als Raster erzwingen* zwingt diese Layer dazu gerastert zu werden, sodass die exportierten Dateien nicht alle enthaltenen Knoten in diesen Layern enthalten müssen und die Wiedergabe daher beschleunigt wird.

  Sie können dies auch tun, indem Sie den Export des Layouts als Raster erzwingen, aber das ist eine Alles-oder-Nichts-Lösung, da die Rasterung auf alle Layer angewendet wird. Alternativ können Sie sich auf die Geometrievereinfachung in *Layout-Exporteinstellungen* verlassen.

- *Layeraktualisierungsintervall*: steuert, ob und wie regelmäßig ein Layer aufgefrischt werden kann. Verfügbare *Konfiguration*-Optionen sind:

  – *Daten neu laden*: Der Layer wird vollständig aktualisiert. Alle zwischengespeicherten Daten werden verworfen und vom Anbieter neu geholt. Dieser Modus kann dazu führen, dass die Karte langsamer aktualisiert wird.

  – *Nur Layer neu zeichnen*: Dieser Modus ist nützlich für Animationen oder wenn der Stil des Layers in regelmäßigen Abständen aktualisiert werden soll. Die Aktualisierung der Kartenansicht wird aufgeschoben, um ein mehrfaches Aktualisieren zu vermeiden, wenn für mehr als einen Layer ein Intervall für die automatische Aktualisierung festgelegt wurde.

  Es ist auch möglich, das *Intervall (Sekunden)* zwischen aufeinanderfolgenden Auffrischungen einzustellen.

- Abhängig vom Datenprovider (z.B. PostgreSQL) können Benachrichtigungen an QGIS gesendet werden, wenn Änderungen an der Datenquelle aus QGIS heraus vorgenommen werden. Verwenden Sie die Option *Layer bei Benachrichtigung aktualisieren*, um eine Aktualisierung auszulösen. Sie können die Aktualisierung des Layers auch auf eine bestimmte Nachricht beschränken, die Sie im Textfeld *Nur bei Nachricht* einstellen.

12.1.15 Zeitlich

Die Registerkarte *Zeitlich* bietet Optionen zur Steuerung der Darstellung des Layers im Zeitverlauf. Für eine solche dynamische Darstellung muss die *zeitabhängige Darstellung* in der Kartenansicht aktiviert sein.

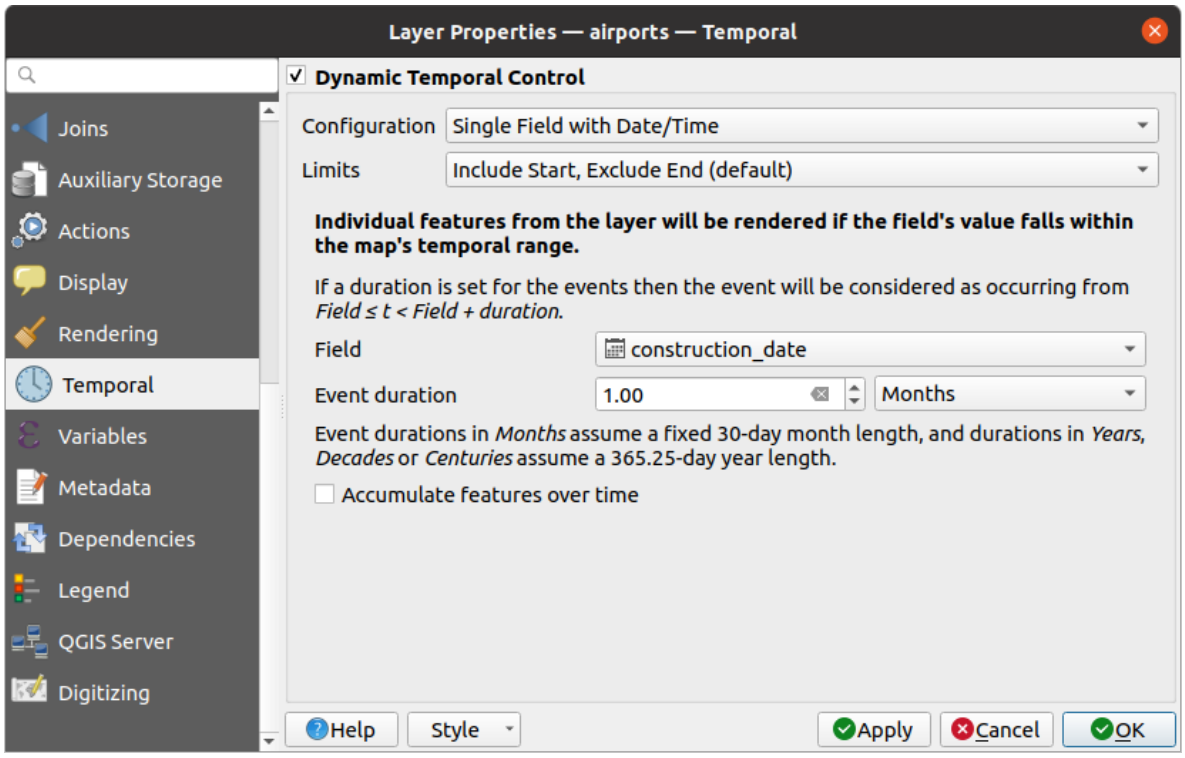


Abb. 12.64: Dialog für zeitliche Eigenschaften von Vector Layern

Aktivieren Sie das Kontrollkästchen *Dynamische Zeitsteuerung*, um das zeitliche Rendering des Vector Layers zu konfigurieren. Je nach der Struktur Ihres Datensatzes sollten Sie eine der angebotenen *Konfiguration* verwenden:

- *Fester Zeitraum*: alle Objekte werden gerendert, wenn sich der zeitliche Rahmen der Kartenansicht mit dem angegebenen Bereich *Startdatum* und *Enddatum* überschneidet.
- *Einzelfeld mit Datum/Uhrzeit*: Objekte werden gerendert, wenn der Wert ihres *Feldes* innerhalb des zeitlichen Rahmens der Kartenansicht liegt. Eine *Ereignisdauer* kann gesetzt werden. Wenn die Option *Objekte über Zeit sammeln* aktiviert ist, werden alle Objekte, die vor oder innerhalb des Zeitrahmens der Karte auftreten, weiterhin gerendert. Die Ereignisdauer wird dabei ignoriert.
- *Getrennte Felder für Anfangs- und Enddatum/-zeit*: Objekte werden gerendert, wenn der durch die Werte *Beginnfeld* und *Endfeld* angegebene Bereich die Kartenansicht zeitlich überlappt.
- *Getrennte Felder für Anfang und Ereignisdauer*: Objekte werden gerendert, wenn der durch ihre Werte im *Beginnfeld* und *Ereignisdauerfeld* definierte Bereich die Kartenansicht zeitlich überlappt.
- *Anfangs- und Enddatum/-zeit aus Ausdrücken*: Objekte werden gerendert, wenn der durch die Felder *Anfangsausdruck* und *Endeausdruck* angegebene Zeitbereich die Kartenansicht zeitlich überlappt.

- *Nur Layer neu zeichnen*: Der Layer wird bei jedem neuen Animationsframe neu gezeichnet, aber es wird keine zeitbasierte Filterung auf die Objekte angewendet. Dies ist nützlich, wenn der Layer zeitbasierte Expression-Werte für Renderer-Einstellungen verwendet (z. B. datendefinierte Symbologie).

Es ist auch möglich, die *Grenzen* des Objekts als Zeitbereich zu setzen:

- *Start einschließen, Ende ausschließen* das ist der Vorgabewert
- *Start und Ende einschließen*

12.1.16 Variablen

Der Reiter *Variablen* listet alle Variablen auf, die auf der Ebene des Layers verfügbar sind (einschließlich aller globalen und Projektvariablen).

Sie ermöglicht dem Benutzer auch die Verwaltung von Variablen auf Layer-Ebene. Klicken Sie auf die Schaltfläche , um eine neue benutzerdefinierte Variable auf Layer-Ebene hinzuzufügen. Ebenso können Sie eine benutzerdefinierte Variable auf Layer-Ebene aus der Liste auswählen und auf die Schaltfläche klicken, um sie zu entfernen.

Weitere Informationen zur Verwendung von Variablen finden Sie im Abschnitt *Speichern von Werten in Variablen*.

12.1.17 Höhe

Die Registerkarte |elevationcale| *Höhe* bietet Optionen zur Steuerung der Layer-Höheneigenschaften in einer *3D-Kartenansicht* und ihrer Darstellung im *Geländehöhenprofil*. Im Einzelnen können Sie einstellen:

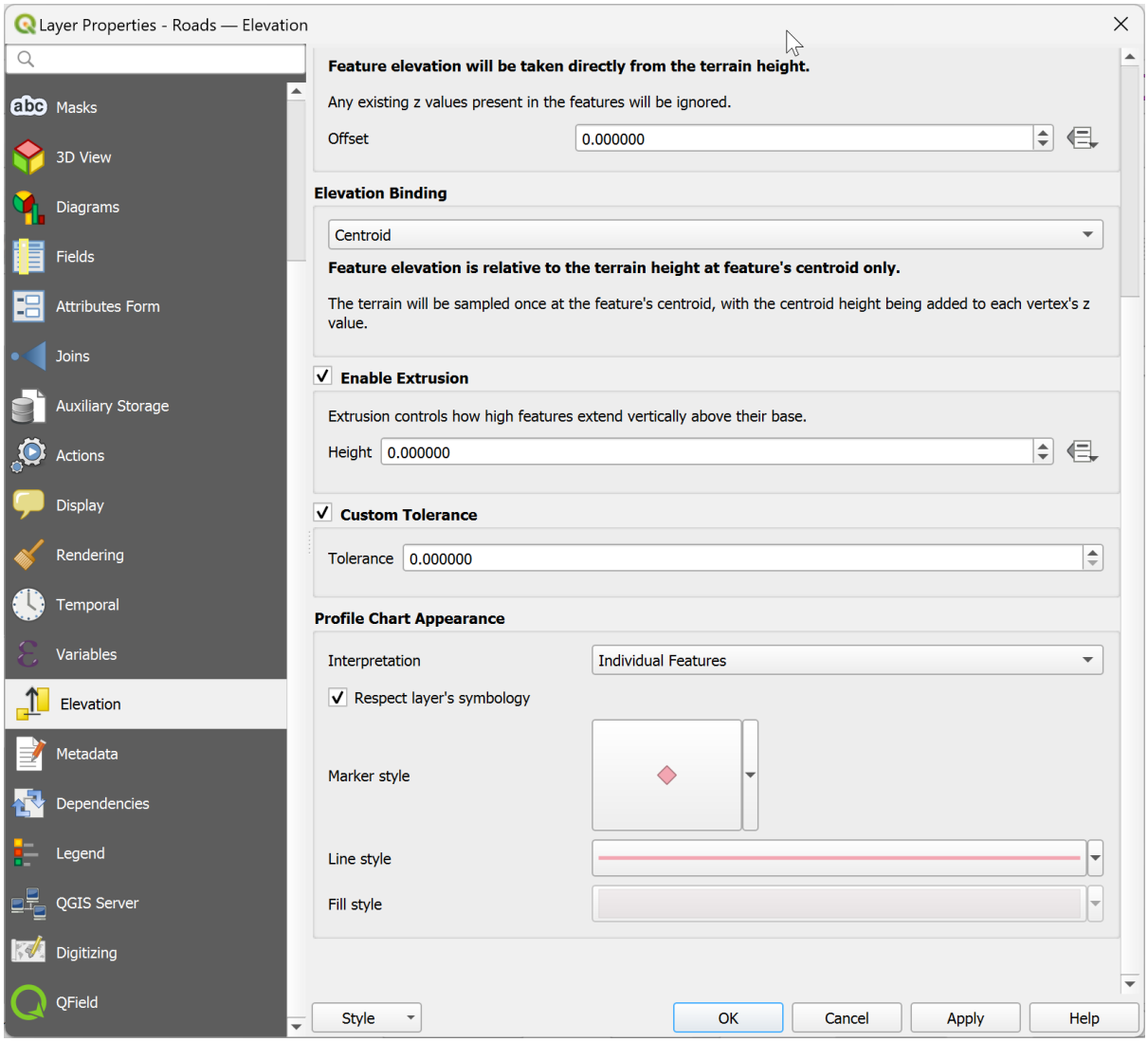


Abb. 12.65: Dialogfeld für die Höhen-Eigenschaften von Vektorlayern

- *Vertikales Bezugssystem*: Wenn das KBS Ihrer Vektorebene ein zusammengesetztes KBS ist (einschließlich einer Z-Dimension), dann ist das für den Layer verwendete vertikale KBS die vertikale Komponente des Layer-KBS. In diesem Fall können Sie kein anderes vertikales KBS manuell festlegen. Wenn Ihr Layer-KBS horizontal (2D) ist, können Sie ein bestimmtes vertikales KBS auswählen, indem Sie auf KBS auswählen klicken. Vertikale Referenzsysteme werden für Vektor-Layer unterstützt durch:
  - **|newElevationProfile|** *Geländehöhenprofilen <label_elevation_profile_view>*
  - *Objekte-abfragen-Werkzeug*
  - *3D-Kartenansichten*
- *Höhenbegrenzung*: definiert, wie und ob die Objekte in der Höhe sein sollen:
  - *Auf Gelände begrenzen*: Übernimmt die Höhe direkt von der Geländehöhe und ignoriert alle vorhandenen Z-Werte in den Objekten. Es kann auch ein von den Daten definierter *Versatz*-Wert vom Terrain gefüllt werden.
  - *Relativ zum Gelände*: alle vorhandenen Z-Werte in den Objekten werden zur Geländehöhe addiert. Ein *Maßstabsfaktor* gefolgt von einem datendefinierten *Versatz* kann zur Anpassung der Höhe verwendet werden. Diese Option ist für 2D-Geometrie-Layer nicht verfügbar.
  - *Absolut*: ignoriert die Geländehöhe und übernimmt direkt die Z-Werte aus den Objekten für die Höhe. Ein *Maßstab*-Faktor, gefolgt von einem datendefinierten *Offset* kann zur Anpassung der Höhe verwendet

werden. Bei 2D-Geometrie-Layern (ohne Z-Werte) kann stattdessen ein datendefiniertes *Basishöhe* gesetzt werden.

- ☐ *Extrudierung aktivieren*: Sie können ein *Höhe* setzen, um festzulegen, wie hoch Objekte vertikal über ihre Basis hinausragen. Dies ist praktisch, um anzuzeigen, dass ein Layer mit 2D-Geometrie, z. B. ein Layer mit polygonalen Gebäudegrundrissen, tatsächlich 3D-Objekte darstellt.
- ☐ *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.
- *Höhenbindung*: Diese Option ist nur von Bedeutung, wenn eine *Höhenbegrenzung*, die sich auf das Terrain bezieht, mit einem Linien- oder Polygon-Layer kombiniert wird. Sie steuert, wie die Höhe des Objekts relativ zur Terrainhöhe festgelegt wird. Das Terrain kann abgetastet werden:
  - am *Zentroid* des Objekts, wobei die Höhe des Schwerpunkts zum z-Wert jedes Scheitelpunkts addiert wird
  - an jedem einzelnen *Stützpunkt*, bevor er zum z-Wert des Scheitelpunkts addiert wird
- *Profildiagrammenerscheinungsbild*: steuert, wie Objekte beim Zeichnen eines Profildiagramms dargestellt werden. Für die *Interpretation* sind zwei Modi verfügbar:
  - *Einzelne Objekte*: tastet diskrete Positionen ab, an denen sich die Querschnittslinie des Profils mit den Vektorobjekten schneidet. Dieser Schnittpunkt kann als Punkt, Linie oder Fläche dargestellt werden, je nach Layer-Typ und ob eine Extrusion angewendet wird.

    Wenn Sie das Kontrollkästchen *Layersymbolik beachten* aktivieren, werden Objekte auf dem Profildiagramm mit dem entsprechenden *Layer-Symboltyp* dargestellt (sodass z.B. kategorisierte Klassen auf dem Profildiagramm sichtbar sind). Wenn der Profilsymboltyp nicht mit den Symboltypen des Layer-Renderers übereinstimmt, wird nur die Symbolfarbe des Renderers auf das Profilsymbol angewendet.

    Abhängig von den Layer-Einstellungen können Profilsymbole mit einem benutzerdefinierten Stil dargestellt werden, indem sie verwendet werden:

    * :ref:`Markierungsstil <vector_marker_symbols> `: für nicht-extrudierte Punkt- und Linienobjekte und für nicht-extrudierte Polygonobjekte, die von der Profillinie berührt werden
    * *Linienstil*: für extrudierte Punkt- und Linienobjekte und für nicht-extrudierte Polygonobjekte, die von der Profillinie geschnitten werden
    * :ref:`Füllstil <vector_fill_symbols> `: für extrudierte Polygonobjekte
  - als *Kontinuierliche Oberfläche (z.B. Höhenlinien)*: Das Höhendiagramm wird als eine Oberfläche anstelle von separaten Objekten dargestellt, indem die abgetasteten Höhenergebnisse zu einer kontinuierlichen Linie verbunden werden. Dies kann die Visualisierung verbessern und ist für Layer gedacht, die eine kontinuierliche Oberfläche darstellen, z. B. Höhenlinien oder vermessene Höhenpunkte. Der Profil- *Stil* kann wie folgt eingestellt werden:

    * eine *Linie* mit einem bestimmten *Linenstil*
    * eine Höhenfläche, die mit einem Füllsymbol entweder über (*Oberhalb füllen*) oder unter (*Unten füllen*) der Höhenkurvenlinie dargestellt wird. Die Oberflächensymbolik wird mit dargestellt:
      · einem *Füllstil*
      · und einer *Begrenzung*: die maximale (bzw. minimale) Höhe, die bestimmt, wie hoch die Füllfläche sein wird

    Außerdem können Sie *Abtastpunkte markieren* aktivieren, um diese über der Interpretationslinie sichtbar zu machen und ihnen einen *Markierungsstil* zuzuweisen.

12.1.18 Metadaten

Der Reiter *Metadata* bietet Ihnen Optionen zum Erstellen und Bearbeiten eines Metadatenberichts zu Ihrem Layer. Siehe *Metadaten* für weitere Informationen.

12.1.19 Abhängigkeiten

Der Reiter *Abhängigkeiten* erlaubt es, Datenabhängigkeiten zwischen Layern zu deklarieren. Eine Datenabhängigkeit tritt auf, wenn eine Datenänderung in einem Layer, die nicht durch direkte Benutzermanipulation erfolgt, Daten anderer Layer verändern kann. Dies ist z. B. der Fall, wenn die Geometrie eines Layers durch einen Datenbank-Trigger oder ein benutzerdefiniertes PyQGIS-Skript aktualisiert wird, nachdem die Geometrie eines anderen Layers geändert wurde.

Auf der Registerkarte *Abhängigkeiten* können Sie alle Layer auswählen, die die Daten des aktuellen Layers von außen verändern können. Die korrekte Angabe von abhängigen Layern ermöglicht es QGIS, die Caches für diesen Layer ungültig zu machen, wenn die abhängigen Layer geändert werden.

12.1.20 Legende

Der Reiter *Legend* bietet Ihnen erweiterte Einstellungen für das *Layer Bedienfeld* und/oder die *Drucklayout Legende*. Diese Optionen umfassen:

- Abhängig von der auf den Layer angewendeten Symbologie können Sie mehrere Einträge in der Legende erhalten, die nicht unbedingt lesbar oder nützlich für die Anzeige sind. Das *Legendenplatzhalterbild* hilft Ihnen bei der *Auswahl eines Bildes* zum Ersetzen, das sowohl im Bedienfeld *Layer* als auch in der Drucklayoutlegende angezeigt wird.
- *Beschriftungslegende anzeigen*: Zeigt Übersichten über die verschiedenen Etiketteneinstellungen als Einträge in den Legenden an. Der *Label-Stil* wird zusammen mit der Beschreibung in der Vorschau angezeigt.
- *Text auf Symbolen*: In manchen Fällen kann es sinnvoll sein, den Symbolen in der Legende zusätzliche Informationen hinzuzufügen. Mit diesem Rahmen können Sie jedem der in der Layer-Symbologie verwendeten Symbole einen Text zuordnen, der sowohl im Bedienfeld *Layer* als auch in der Drucklayoutlegende über dem Symbol angezeigt wird. Diese Zuordnung erfolgt durch Eingabe des Textes neben dem Symbol im Tabellen-Widget oder durch Ausfüllen der Tabelle mit der Schaltfläche *Beschriftungen aus Ausdruck setzen*. Das Aussehen des Textes wird über die Schriftart- und Farbauswahl-Widgets der Schaltfläche *Textformat* gesteuert.

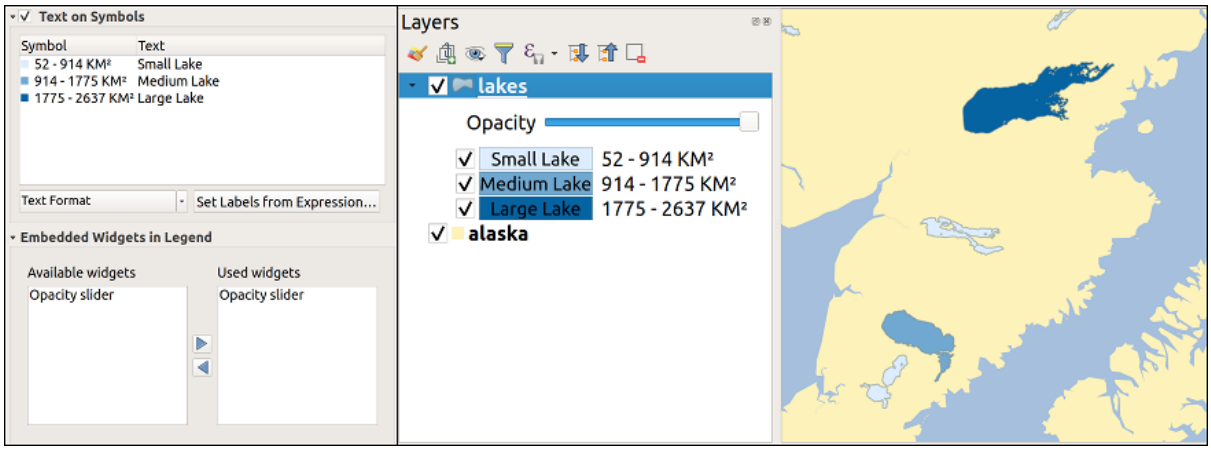


Abb. 12.66: Setzen von Text auf Symbolen (links) und seine Darstellung im Bedienfeld *Layer* (rechts)

- eine Liste von Widgets, die Sie in den Layer-Baum im Bedienfeld Layers einbetten können. Die Idee ist, eine Möglichkeit zu haben, schnell auf einige Aktionen zuzugreifen, die häufig mit dem Layer verwendet werden (Einrichten von Transparenz, Filtern, Auswahl, Stil oder anderen Dingen…).

  Standardmäßig stellt QGIS ein Transparenz-Widget zur Verfügung, das jedoch durch Plugins erweitert werden kann, die ihre eigenen Widgets registrieren und den von ihnen verwalteten Layern eigene Aktionen zuweisen.

12.1.21 QGIS Server

Der Reiter *QGIS Server* besteht aus den Abschnitten *Beschreibung*, *Beschreibung*, *Metadaten-URL* und *Legend-URL*.

Im Abschnitt *Beschreibung* können Sie den *Kurznamen* ändern, der in Anfragen zur Referenzierung des Layers verwendet wird (weitere Informationen zu Kurznamen finden Sie unter services_basics_short_name). Sie können hier auch einen *Titel*, einen alternativen *WFS-Titel* und eine *Zusammenfassung* für die Layer hinzufügen oder bearbeiten oder eine *Schlüsselwortliste* definieren. Diese Schlüsselwortlisten können in einem Metadatenkatalog verwendet werden. Wenn Sie einen Titel aus einer XML-Metadatendatei verwenden möchten, müssen Sie einen Link in das Feld *Daten-URL* eingeben.

Verwenden Sie *Attribution*, um Attributdaten aus einem XML-Metadatenkatalog zu erhalten.

In *Metadaten URL* können Sie die allgemeinen Pfade zum XML-Metadatenkatalog angeben. Diese Informationen werden in der QGIS-Projektdatei für spätere Sitzungen gespeichert und für QGIS Server verwendet.

Im Abschnitt *Legenden-URL* können Sie die URL eines Legendenbildes in das Feld url eingeben. Über die Dropdown-Option Format können Sie das entsprechende Bildformat auswählen. Derzeit werden die Bildformate png, jpg und jpeg unterstützt.

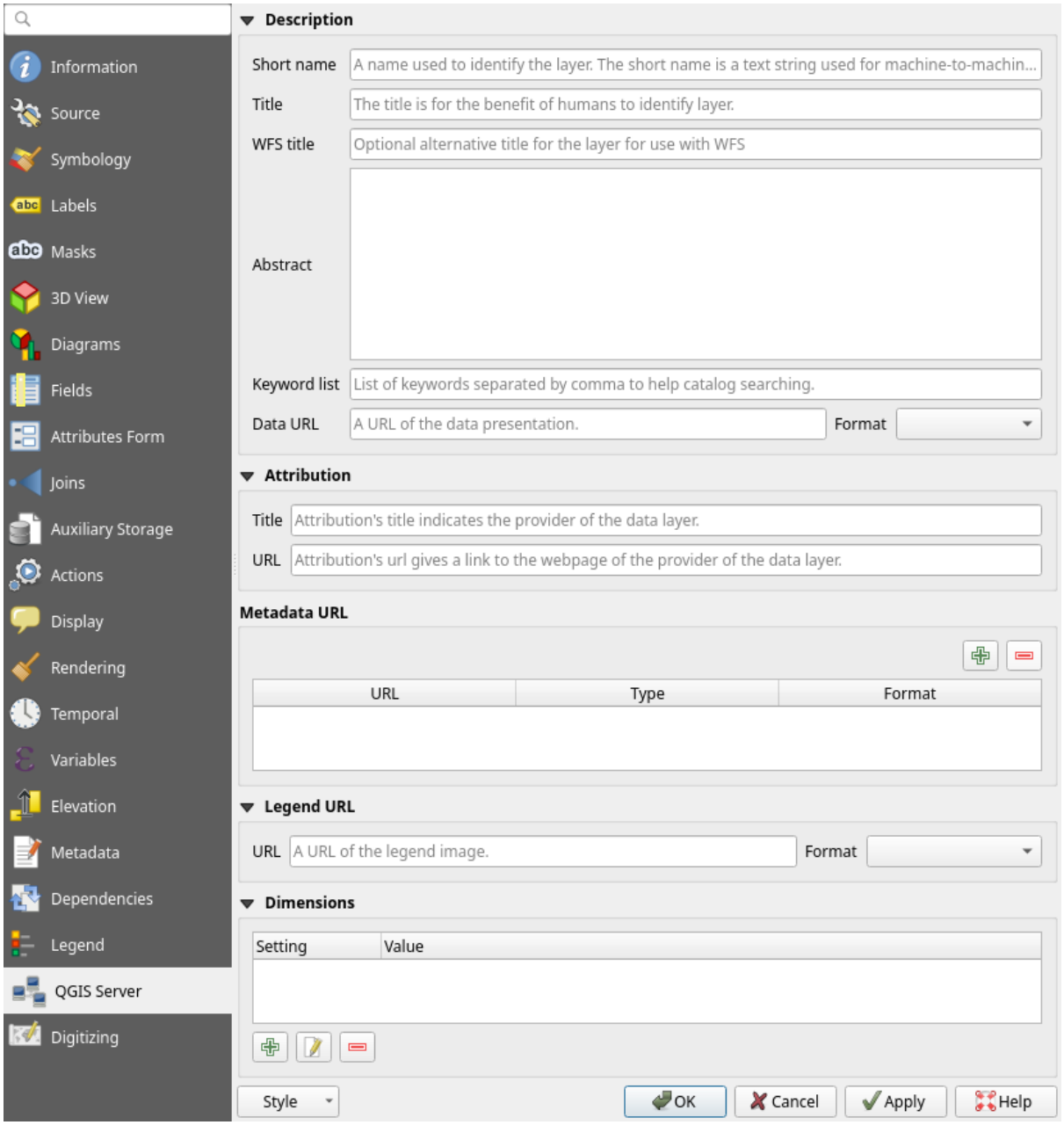


Abb. 12.67: QGIS Server-Registerkarte im Eigenschaftsdialog für Vektorlayer

Um mehr über QGIS Server zu erfahren, lesen Sie die QGIS-Server-manual.

12.1.22 Digitalisieren

Der Reiter *Digitalisieren* ermöglicht den Zugriff auf Optionen, die dazu beitragen, die Qualität der digitalisierten Geometrien zu gewährleisten.

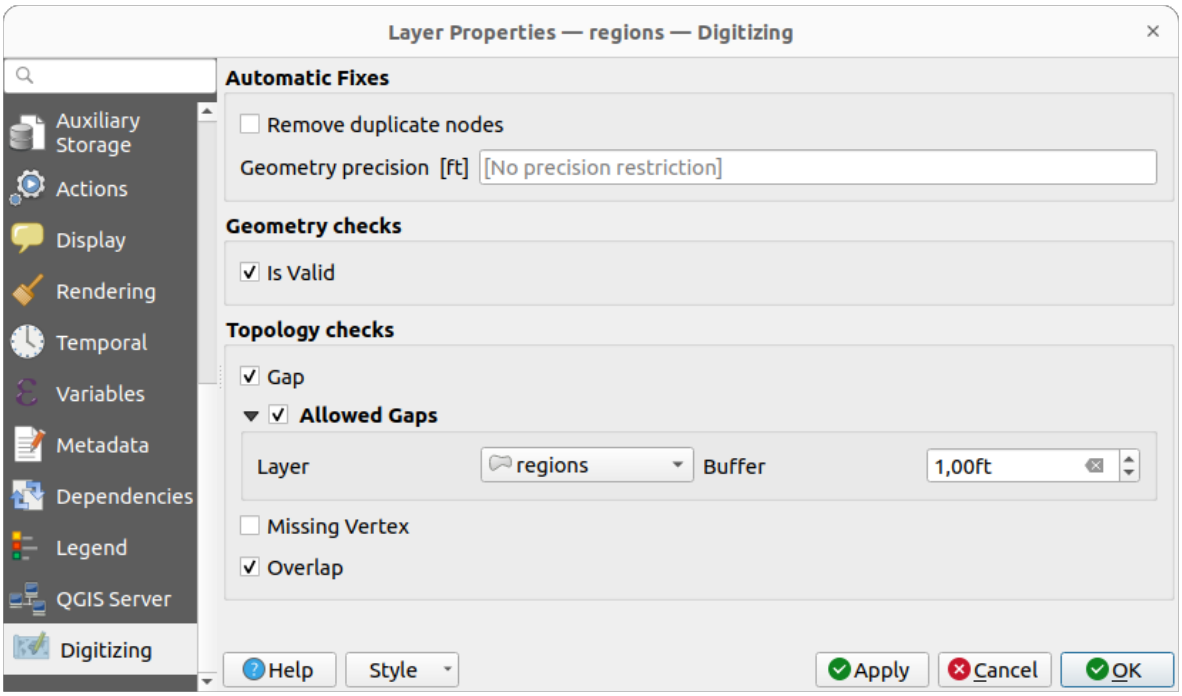


Abb. 12.68: Die QGIS-Registerkarte „Digitalisieren“ im Eigenschaftsdialog für Vektor-Layer

Automatische Reparaturen

Die Optionen im Abschnitt *Automatische Reparaturen* wirken sich direkt auf die Eckpunkte jeder hinzugefügten oder geänderten Geometrie aus. Wenn die Option *Doppelte Knoten entfernen* aktiviert ist, werden zwei aufeinanderfolgende Knoten mit exakt den gleichen Koordinaten entfernt. Wenn die Option *Geometriegenauigkeit* gesetzt ist, werden alle Scheitelpunkte auf das nächste Vielfache der konfigurierten Geometriegenauigkeit gerundet. Die Rundung erfolgt im Layer-Koordinatenreferenzsystem. Z- und M-Werte werden nicht gerundet. Bei vielen Kartenwerkzeugen wird während des Digitalisierens ein Raster auf der Kartenansicht angezeigt.

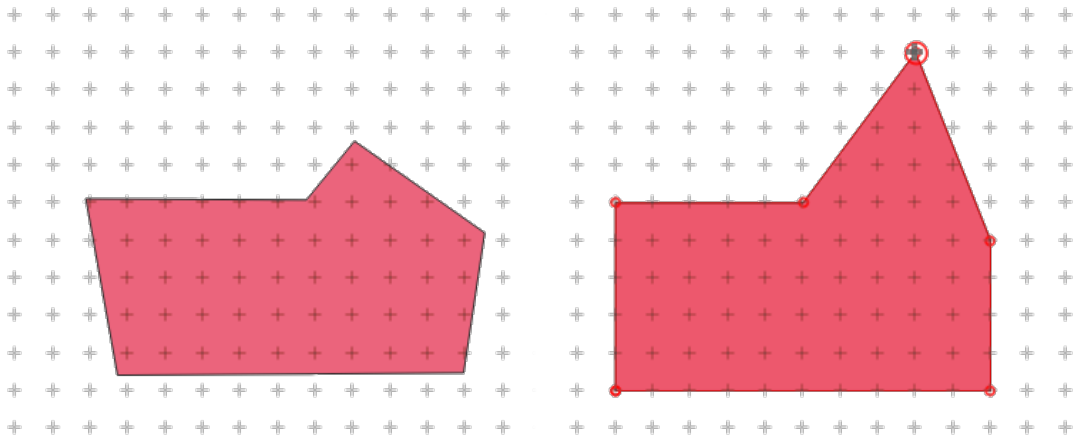


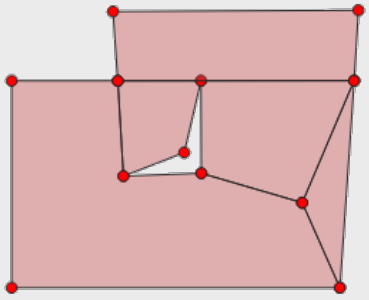
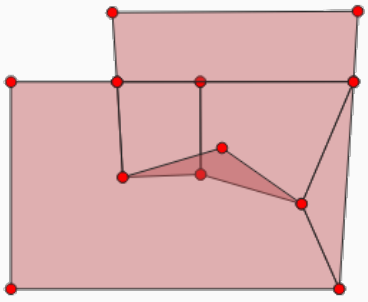
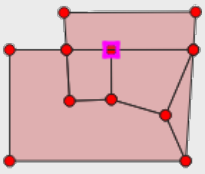
Abb. 12.69: Wird der oberste Stützpunkt verschoben, werden alle Stützpunkte auf das Gitter eingerastet

Geometrieprüfungen

Im Abschnitt *Geometrieprüfungen* können zusätzliche Prüfungen für jede einzelne Geometrie aktiviert werden. Unmittelbar nach jeder Geometrieänderung wird dem Benutzer im Bedienfeld *Geometrievalidierung* ein Fehlschlagen dieser Prüfungen gemeldet. Solange eine Prüfung fehlschlägt, ist es nicht möglich, den Layer zu speichern. Die *Ist gültig*-Prüfung führt grundlegende Gültigkeitsprüfungen wie z.B. die Selbstüberschneidung von Geometrien durch.

Topologieprüfungen

Im Abschnitt *Topologieprüfungen* können zusätzliche Topologieprüfungen aktiviert werden. Die Topologieprüfungen werden ausgeführt, wenn der Benutzer den Layer speichert. Prüffehler werden im Bedienfeld *Geometrievalidierung* angezeigt. Solange Validierungsfehler vorhanden sind, kann der Layer nicht gespeichert werden. Topologieprüfungen werden im Bereich der Bounding Box der geänderten Objekte durchgeführt. Da im selben Bereich auch andere Objekte vorhanden sein können, werden topologische Fehler, die diese Objekte betreffen, ebenso gemeldet wie Fehler, die in der aktuellen Bearbeitungssitzung entstanden sind.

Option Topologieprüfung	Illustration
Das Kontrollkästchen *Lücke* prüft auf Lücken zwischen benachbarten Polygonen.	
Das Kontrollkästchen *Überlappt* prüft auf Überlappungen zwischen benachbarten Polygonen.	
Die *Stützpunkt fehlt* prüft auf gemeinsame Grenzen benachbarter Polygone, bei denen eine Grenze einen Scheitelpunkt vermisst, der auf der anderen vorhanden ist.	

Ausnahmen bei der Lückenprüfung

Manchmal ist es wünschenswert, Lücken innerhalb eines Gebiets in einem Layer mit Polygonen zu lassen, das ansonsten vollständig von Polygonen bedeckt ist. Zum Beispiel kann ein Landnutzungs Layer akzeptable Lücken für Seen haben. Es ist möglich, Bereiche zu definieren, die bei der Lückenprüfung ignoriert werden. Da Lücken innerhalb dieser Bereiche erlaubt sind, werden sie als *Allowed Gaps*-Bereiche bezeichnet.

In den Optionen für die Lückenprüfungen unter *Erlaubte Lücken* kann ein *Erlaubte Lücken Layer* konfiguriert werden.

Wenn die Lückenprüfung durchgeführt wird, werden Lücken, die durch ein oder mehrere Polygone im *Erlaubte Lücken Layer* abgedeckt sind, nicht als Topologiefehler gemeldet.

Es ist auch möglich, einen zusätzlichen *Puffer* zu konfigurieren. Dieser Puffer wird auf jedes Polygon auf dem *Erlaubte Lücken Layer* angewendet. Dies ermöglicht es, die Tests weniger anfällig für kleine Änderungen der Umrisse an den Grenzen der Lücken zu machen.

Wenn *Erlaubte Lücken* aktiviert sind, ist eine zusätzliche Schaltfläche (*Erlaubte Lücke hinzufügen*) für erkannte Lückenfehler im Geometrievalidierungsdock verfügbar, wo Lücken während der Digitalisierung gemeldet werden. Wird die Schaltfläche *Erlaubte Lücke hinzufügen* betätigt, wird ein neues Polygon mit der Geometrie der erkannten Lücke in den *Layer für erlaubte Lücken* eingefügt. Dies ermöglicht es, Lücken schnell als erlaubt zu markieren.

Bedienfeld für die Geometrieprüfung

Das Bedienfeld *Geometrieprüfung* wird ausgelöst, wenn bei einer der oben genannten Digitalisierungsprüfungen ein Fehler festgestellt wird. Das Dialogfeld enthält eine Liste der Fehler und deren Beschreibung. Sie können die Liste mit den Pfeiltasten der Tastatur oder den entsprechenden Pfeilen durchblättern.

Bevor Sie die Bearbeitungen auf dem Layer speichern können, müssen Sie alle Probleme beheben. Tun Sie das:

1. Wählen Sie einen Fehler aus, und Sie können ihn beheben:
   - *Auf Objekt(e) zoomen*
   - *Auf Problem zoomen*
2. Wählen Sie die üblichen *Digitalisierungswerkzeuge*, um das Problem zu beheben.

12.2 Arbeiten mit der Attributtabelle

Die Attributtabelle zeigt Informationen zu Objekten eines ausgewählten Layers an. Jede Zeile in der Tabelle steht für ein Objekt (mit oder ohne Geometrie), und jede Spalte enthält eine bestimmte Information über das Objekt. Objekte in der Tabelle können gesucht, ausgewählt, verschoben oder auch bearbeitet werden.

12.2.1 Vorbemerkungen: Räumliche und nicht räumliche Tabellen

Mit QGIS können Sie räumliche und nicht-räumliche Layer laden. Dazu gehören derzeit Tabellen, die von GDAL unterstützt werden und durch Trennzeichen getrennter Text (CSV), sowie die Anbieter PostgreSQL, MS SQL Server, SpatiaLite und Oracle. Alle geladenen Layer werden in dem Bedienfeld *Layer* aufgelistet. Ob ein Layer räumlich aktiviert ist oder nicht, bestimmt, ob Sie mit ihm auf der Karte interagieren können.

Nicht-räumliche Tabellen können über die Attributtabellenansicht durchsucht und bearbeitet werden. Außerdem können sie für die Feldsuche verwendet werden. Sie können beispielsweise Spalten einer nicht-räumlichen Tabelle verwenden, um Attributwerte oder einen Bereich zulässiger Werte zu definieren, die während der Digitalisierung zu einem bestimmten Vektor-Layer hinzugefügt werden. Schauen Sie sich das Editier-Widget im Abschnitt *Attributformular* genauer an, um mehr darüber zu erfahren.

12.2.2 Einführung in die Benutzeroberfläche der Attributtabelle

Um die Attributtabelle für einen Vektor-Layer zu öffnen, aktivieren Sie den Layer, indem Sie ihn in der *Layer-Bedienfeld* anklicken. Wählen Sie dann im Hauptmenü *Layer* den Befehl *Attributtabelle öffnen*. Sie können auch mit der rechten Maustaste auf den Layer klicken und aus dem Kontextmenü den Befehl *Attributtabelle öffnen* wählen, oder Sie klicken auf die Schaltfläche *Attributtabelle öffnen* in der Symbolleiste Attribute. Wenn Sie Tastaturkürzel bevorzugen, können Sie mit `F6` die Attributtabelle öffnen, mit `Umschalt+F6` die Attributtabelle gefiltert nach ausgewählten Objekten und mit `Strg+F6` die Attributtabelle gefiltert nach sichtbaren Objekten öffnen.

Dadurch wird ein neues Fenster geöffnet, das die Objektattribute für den Layer anzeigt (*figure_attributes_table*). Je nach Einstellung im Menü *Einstellungen* ► *Optionen* ► *Datenquellen* wird die Attributtabelle in einem angedockten Fenster oder in einem normalen Fenster geöffnet. Die Gesamtzahl der Objekte im Layer und die Anzahl der aktuell selektierten/gefilterten Objekte werden im Titel der Attributtabelle angezeigt, ebenso, ob der Layer räumlich begrenzt ist.

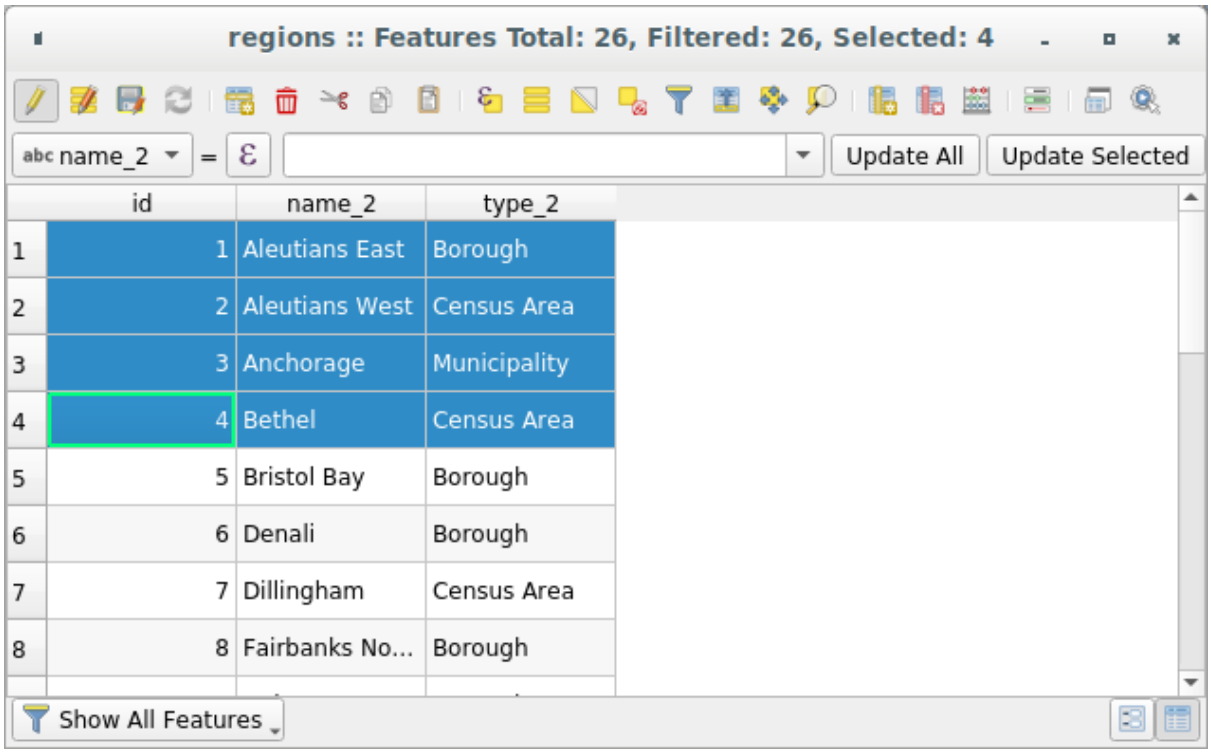


Abb. 12.70: Attributtabelle des regions Layer

Die anderen Knöpfe im Kopf der Attributtabelle stellen die folgenden Funktionalitäten zur Verfügung:

Tab. 12.1: Verfügbare Werkzeuge

Icon	Bezeichnung	Zweck
	Bearbeitungsmodus umschalten	Aktivieren der Bearbeitungsfunktionen
	Mehrfachbearbeitungsmodus umschalten	Mehrere Felder vieler Objekte aktualisieren
	Änderungen speichern	Die aktuellen Änderungen speichern
	Attributtabelle aktualisieren	
	Objekt hinzufügen	Hinzufügen eines neuen, geometrielosen Objekts
	Gewählte Objekte löschen	Ausgewählte Objekte aus dem Layer entfernen

Fortse

Tab. 12.1 – Fortsetzung der vorherigen Seite

Icon	Bezeichnung	Zweck
	Gewählte Zeilen in Zwischenablage ausschneiden	
	Ausgewählte Objekte in Zwischenablage kopieren	
	Objekte aus Zwischenablage einfügen	Einfügen neuer Objekte aus kopierten Objekten
	Objekte über Ausdruck wählen	
	Alle Objekte wählen	Alle Objekte eines Layers auswählen
	Objektauswahl umkehren	Die aktuelle Auswahl im Layer umkehren
	Alle Objektauswahl aufheben	Deselektieren aller Objekte im aktuellen Layer
	Objekte mit Formular wählen/filtern	
	Auswahl nach oben	Ausgewählte Zeilen an den Anfang der Tabelle verschieben
	Karte zu gewählten Objekten verschieben	
	Zu ausgewählten Objekten zoomen	
	Neue Spalte	Hinzufügen eines neuen Feldes zur Datenquelle
	Spalte löschen	Ein Feld aus der Datenquelle entfernen
	Spalten anordnen	Felder aus der Attributtabelle ein-/ausblenden
	Feldrechner öffnen	Feld für viele Objekte in einer Reihe aktualisieren
	Bedingte Formatierungsregeln	Aktivieren der Tabellenformatierung
	Attributtabelle docken	Ermöglicht das Andocken/Abkoppeln der Attributtabelle im Hauptfenst
	Actions	Listet die mit dem Layer verbundenen Aktionen auf

Achtung

Abhängig vom Format der Daten und der GDAL-Bibliothek, die mit Ihrer QGIS-Version erstellt wurde, können einige Werkzeuge nicht verfügbar sein.

Unterhalb dieser Schaltflächen befindet sich die Schnellfeldberechnungsleiste (nur im *Bearbeitungsmodus* aktiviert), die es ermöglicht, schnell Berechnungen auf alle oder einen Teil der Objekte im Layer anzuwenden. Diese Leiste verwendet die gleichen *Ausdrücke* wie der Feldrechner (siehe *Bearbeiten von Attributwerten*).

Tabellenansicht und Formularansicht

QGIS bietet zwei Ansichtsmodi, um die Daten in der Attributtabelle einfach zu bearbeiten:

- Die Tabellenansicht zeigt die Werte mehrerer Objekte in einem tabellarischen Modus an, wobei jede Zeile ein Objekt und jede Spalte ein Feld darstellt. Ein Rechtsklick auf die Spaltenüberschrift ermöglicht die *Konfiguration der Tabellenanzeige*, während ein Rechtsklick auf eine Zelle die *Interaktion mit dem Objekt* ermöglicht.

  Die Attributtabelle unterstützt `Umschalt+Mausrad` im Modus Tabellenansicht, um zwischen vertikalen und horizontalen Bildlaufbewegungen zu wechseln. Dies kann auch erreicht werden, indem man unter macOS die Maus durch das Trackpad ersetzt.

- Die Formularansicht ist zweigeteilt und zeigt auf der linken Seite nur den *Anzeigenamen* jeden Objekts. Auf der rechten Seite werden die Attribute des links ausgewählten Objekts angezeigt. Am oberen Rand der linken

Seite befindet sich ein Pull-Down-Menü, in dem die Standardanzeige in diesem Bereich in ein anderes Attribut (*Spaltenvoransicht*) oder über einen *Ausdruck* geändert werden kann. Das Pull-down enthält auch die letzten 10 Ausdrücke zur Wiederverwendung. Die Formularansicht verwendet die Konfiguration der Layer-Felder (siehe *Attributformular*).

Mit den Pfeilen am unteren Rand des linken Bereichs können Sie durch die einzelnen Objekte blättern. Die Attribute der Objekte werden im rechten Bereich aktualisiert, wenn Sie weiterblättern. Es ist auch möglich, das aktive Objekt in der Kartenansicht zu identifizieren oder dorthin zu wechseln, indem Sie eine der Schaltflächen drücken:

- Aktuelles Objekt auf Karte hervorheben wenn es in der Kartenansicht sichtbar ist
- Automatisch zum aktuellen Objekt verschieben
- Automatisch zum aktuellen Objekt zoomen

Sie können zwischen Tabellenansicht und Formularansicht wechseln, indem Sie auf das entsprechende Symbol unten rechts im Dialogfeld klicken.

Sie können auch den Modus *Voreingestellte Ansicht* beim Öffnen der Attributtabelle im Menü *Einstellungen* ► *Optionen* ► *Datenquellen* festlegen. Dies kann ‚Letzte Ansicht merken', ‚Tabellenansicht' oder ‚Formularansicht' sein.

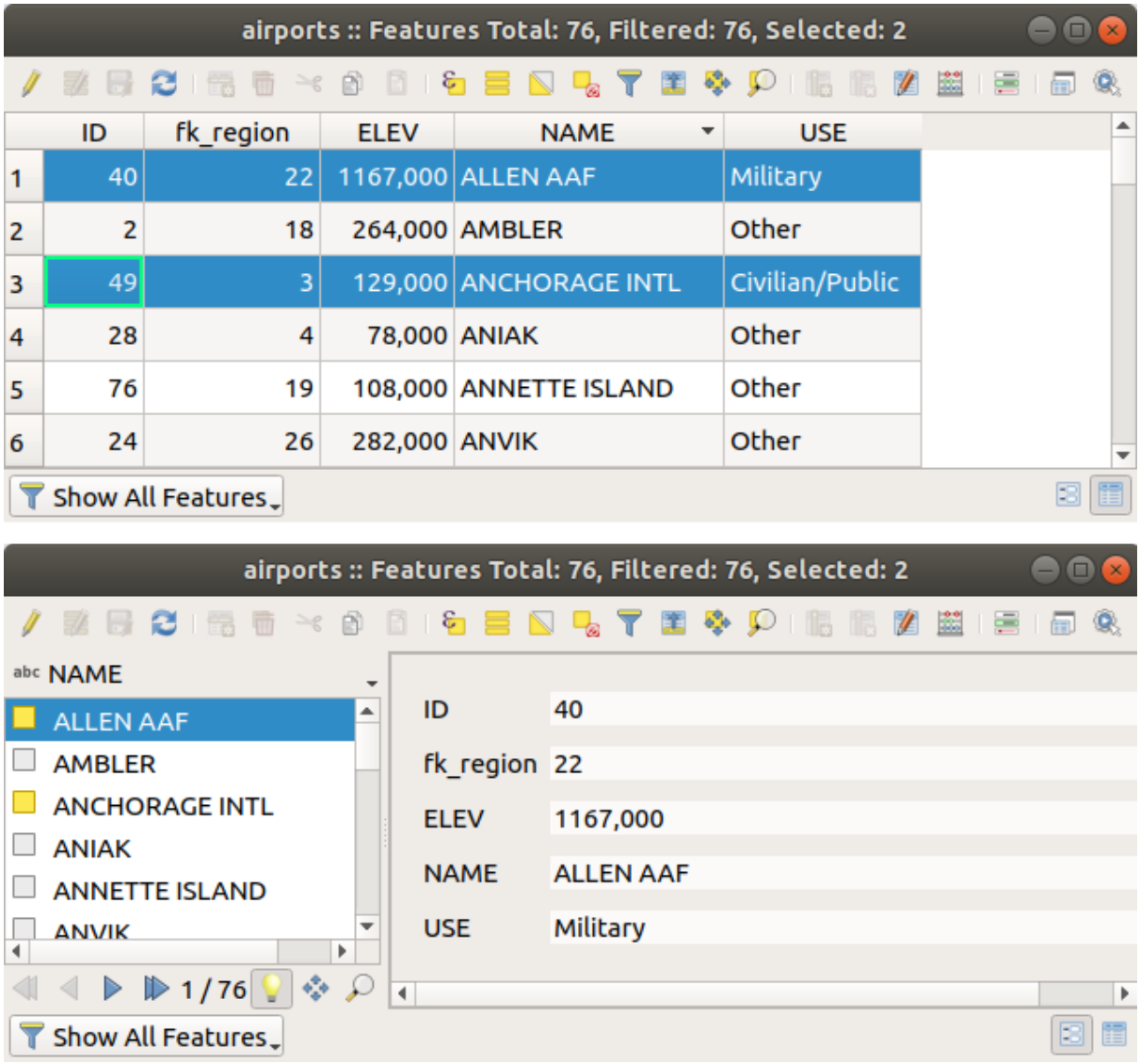


Abb. 12.71: Attributtabelle in Tabellenansicht (oben) vs. Formularansicht (unten)

Konfigurieren der Spalten

Klicken Sie in der Tabellenansicht mit der rechten Maustaste auf eine Spaltenüberschrift, um Zugriff auf Werkzeuge zu erhalten, die Ihnen die Kontrolle erleichtern:

- die *Spaltenbreite*
- die *Spaltensichtbarkeit und -sortierung*
- die *Sortierreihenfolge der Daten*

Ändern der Spaltenbreiten

Die Spaltenbreite kann durch einen Rechtsklick auf die Spaltenüberschrift eingestellt werden und wählen Sie entweder:

- *Breite setzen…*, um den gewünschten Wert einzugeben (der aktuelle Wert wird im sich öffnenden Fenster angezeigt)
- *Alle Spaltenbreiten setzen…* auf den gleichen Wert
- *Autogröße*, um die Größe der Spalte bestmöglich anzupassen.
- *Alle Spaltengrößen automatisch einstellen*

Die Größe einer Spalte kann auch durch Ziehen der Begrenzung rechts von ihrer Überschrift geändert werden. Die neue Größe der Spalte wird für den Layer beibehalten und beim nächsten Öffnen der Attributtabelle wiederhergestellt.

Unter Einstellungen > Optionen können Sie auf dem Reiter *Datenquellen* das Kontrollkästchen *Beim Öffnen der Attributtabelle die Spaltengröße automatisch anpassen* aktivieren, wodurch diese Einstellung als Standardansicht verwendet wird, wenn Attributtabellen geöffnet werden.

Ausblenden und Organisieren von Spalten und Aktivieren von Aktionen

Durch einen Rechtsklick auf eine Spaltenüberschrift können Sie guilabel:*Spalte verbergen* auswählen, um diese aus der Attributtabelle auszublenden (im Modus „Tabellenansicht“). Für erweiterte Steuerungsmöglichkeiten klicken Sie auf die Schaltfläche Spalten anordnen… in der Dialog-Symbolleiste oder wählen Sie *Spalten anordnen…* im Kontextmenü der Spaltenüberschrift.

Im Dialogfeld *Spalten anordnen…* haben Sie auch die folgenden Optionen:

- Aktivieren/deaktivieren Sie die Spalten, die Sie ein- oder ausblenden möchten: Eine ausgeblendete Spalte verschwindet aus allen Instanzen des Attributtabellendialogs, bis sie aktiv wiederhergestellt wird. Außerdem ist es möglich:
  - Wählen Sie *Alle anzeigen*, um alle Felder (Spalten) und Aktionen in der Tabelle anzuzeigen.
  - Wählen Sie *Alle verbergen*, um alle Felder (Spalten) und Aktionen in der Tabelle auszublenden.
  - Verwenden Sie *Auswahl umschalten*, um die Sichtbarkeit der aktuell ausgewählten Spalten umzukehren. Mit den gängigen *Tastenkombination* können Sie mehrere Spalten auswählen.
- Ziehen und Ablegen von Elementen, um die Spalten in der Attributtabelle neu anzuordnen. Beachten Sie, dass sich diese Änderung auf die Darstellung der Tabelle bezieht und die Reihenfolge der Felder in der Layer-Datenquelle nicht ändert
- eine neue virtuelle Spalte *Actions* hinzufügen, die in jeder Zeile eine Dropdown-Box oder eine Button-Liste mit aktivierten Aktionen anzeigt. Siehe *Aktionen* für weitere Informationen über Aktionen.

Zeilen sortieren

Die Zeilen können nach jeder Spalte sortiert werden, indem Sie auf die Spaltenüberschrift klicken. Ein kleiner Pfeil zeigt die Sortierreihenfolge an (nach unten zeigend bedeutet absteigende Werte von der obersten Zeile abwärts, nach oben zeigend bedeutet aufsteigende Werte von der obersten Zeile abwärts). Sie können die Zeilen auch mit der Option *Sortieren…* im Kontextmenü der Spaltenüberschrift sortieren und einen Ausdruck eingeben. Um z.B. die Zeilen nach mehreren Spalten zu sortieren, können Sie `concat(col0, col1)` schreiben.

In der Formularansicht können die Objekte mit der Option *Sortieren nach Vorschauausdruck* sortiert werden.

Beachten Sie, dass sich die Sortierung der Zeilen nur auf die Darstellung der Tabelle auswirkt und die Reihenfolge der Objekte in der Layer-Datenquelle nicht verändert.

Tipp

Sortieren nach Spalten unterschiedlichen Typs

Der Versuch, eine Attributtabelle auf der Grundlage von Spalten mit String- und numerischen Typen zu sortieren, kann zu unerwarteten Ergebnissen führen, da der Ausdruck `concat(„USE“, „ID“)` String-Werte zurückgibt (z.B. `'Borough105' < 'Borough6'`). Sie können dies umgehen, indem Sie z.B. `concat(„USE“, lpad(„ID“, 3, 0))` verwenden, das `'Borough105' > 'Borough006'` liefert.

Formatierung von Tabellenzellen unter Verwendung von Bedingungen

Mit den Einstellungen für die bedingte Formatierung können Sie in der Attributtabelle die Objekte hervorheben, die Sie besonders herausstellen möchten, indem Sie benutzerdefinierte Bedingungen für die Objekte verwenden:

- Geometrie (z.B. Identifizierung von mehrteiligen Objekten, kleinflächigen Objekten oder in einem definierten Kartenausschnitt…);
- oder Feldwert (z. B. Vergleich der Werte mit einem Schwellenwert, Erkennung leerer Zellen, Duplikate, …).

Sie können das Bedienfeld für die bedingte Formatierung aktivieren, indem Sie auf die Schaltfläche Conditional formatting oben rechts im Attributfenster in der Tabellenansicht klicken (in der Formularansicht wird die Darstellung nicht beeinflusst).

Das Werkzeug ermöglicht es dem Benutzer, neue Regeln zur Formatierung von einem *Feld* oder für eine *ganze Zeile* hinzuzufügen. Das Hinzufügen einer neuen Regel öffnet ein Formular zur Definition:

- der Name der Regel;
- eine Bedingung mit einer Funktion des *Ausdruckeditors*;
- die Formatierung: Sie kann aus einer Liste von vordefinierten Formaten ausgewählt oder anhand von Eigenschaften wie
  - Hintergrund- und Textfarben;
  - Verwendung des Symbols;
  - fett, kursiv, unterstrichen oder durchgestrichen;
  - Schriftart.

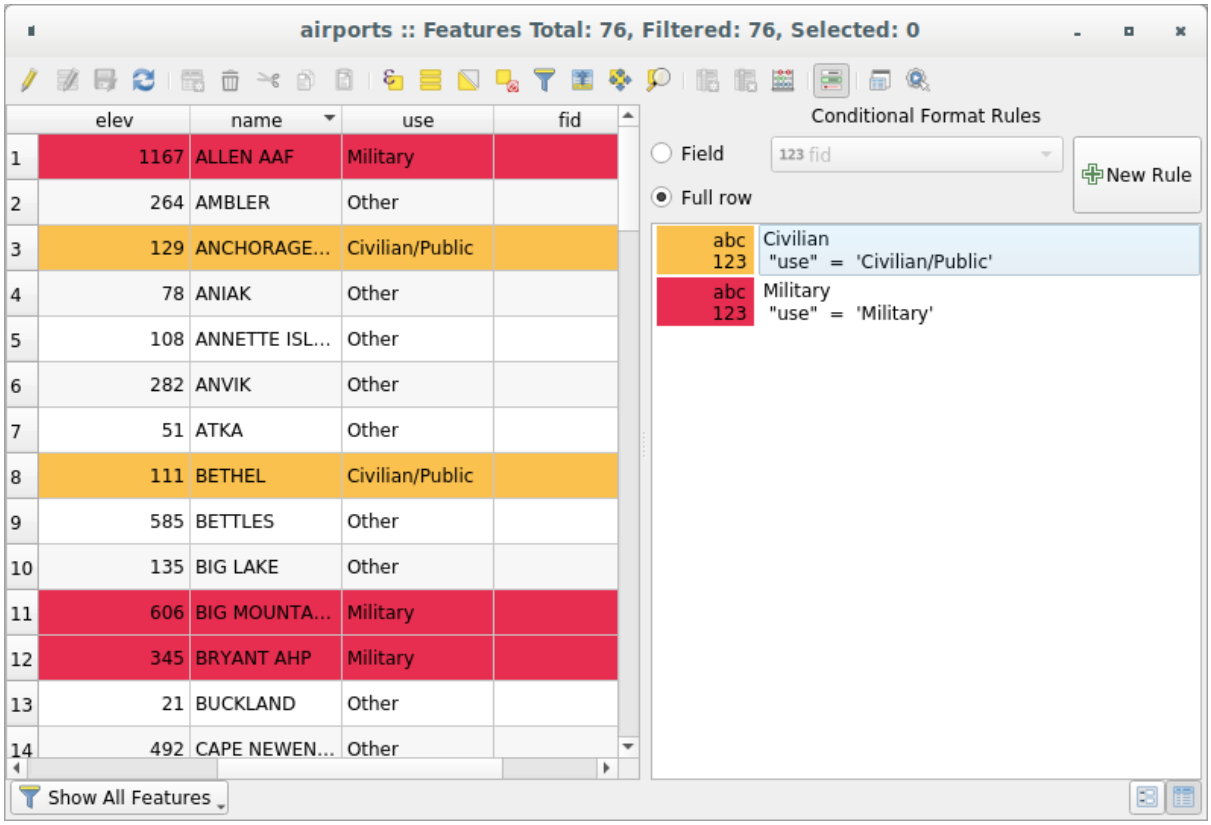


Abb. 12.72: Bedingte Formatierungsregeln für Attributtabellen

12.2.3 Mit den Objekten in einer Attributtabelle interagieren

Objekte auswählen

In der Tabellenansicht zeigt jede Zeile in der Attributtabelle die Attribute eines bestimmten Objekts im Layer an. Durch die Auswahl einer Zeile wird das Objekt ausgewählt, und ebenso wird durch die Auswahl eines Objekts in der Kartenansicht (vorrausgesetzt, der Layer enthälte raumliche Daten) die Zeile in der Attributtabelle ausgewählt. Wird die Menge der in der Kartenansicht (oder der Attributtabelle) ausgewählten Objekte geändert, wird die Auswahl in der Attributtabelle (oder der Kartenansicht) ebenfalls entsprechend aktualisiert.

Zeilen können ausgewählt werden, indem Sie auf die Zeilennummer links neben der Zeile klicken. **Mehrere Zeilen** können ausgewählt werden, indem die `Strg` Taste während der Auswahl gedrückt wird.Eine **kontinuierliche Auswahl** ist möglich, indem Sie bei der Selektion die `Umschalt` Taste gedrückt halten, während Sie die Zeilennummern auswählen. Alle Zeilen zwischen der aktuell ausgewählten Zeile und der Mauspfeilposition werden dadurch selektiert. Bewegt man den Mauszeiger in der Attributtabelle indem man in eine Zelle in der Tabelle klickt ändert die Zeilenauswahl nicht. Verändert man die Auswahl im Kartenfenster bewegt sich der Mauszeiger in der Attributtabelle nicht.

In der Formularansicht der Attributtabelle werden Objekte im linken Bedienfeld standardmäßig durch den Wert des Anzeigenamens (siehe *Anzeige*) repräsentiert. Dieser Anzeigenamen kann über die Dropdown-Liste am oberen Rand des Bedienfelds ersetzt werden, entweder durch Auswahl eines vorhandenen Feldes oder durch einen benutzerdefinierten Ausdruck. Sie können die Liste der Objekte auch über das Dropdown-Menü sortieren lassen.

Klicken Sie auf einen Wert im linken Bedienfeld, um die Attribute des Objekts im rechten Bedienfeld anzuzeigen. Um ein Objekt auszuwählen, müssen Sie in das quadratische Symbol links neben dem Bezeichner klicken. Standardmäßig wird das Symbol gelb. Wie in der Tabellenansicht können Sie mehrere Objekte mit den zuvor beschriebenen Tastenkombinationen auswählen.

Neben der Auswahl von Objekten mit der Maus können Sie auch eine automatische Auswahl auf der Grundlage

von Objektattributen durchführen, indem Sie die in der Symbolleiste der Attributtabelle verfügbaren Werkzeuge verwenden, wie z. B. (siehe Abschnitt *Automatische Auswahl von Objekten* und folgende für weitere Informationen und Anwendungsfälle):

- *Objekt über Ausdruck wählen…*
- *Objekt nach Wert wählen…*
- *Alle Objektauswahl aufheben*
- *Alle Objekte wählen*
- *Objektauswahl umkehren.*

Darüber hinaus ist es möglich, *Objekte über ein Formular zu wählen*.

Objekte filtern

Sobald Sie Objekte in der Attributtabelle ausgewählt haben, ist es möglich, nur diese Datensätze in der Tabelle anzeigen zu lassen. Dazu verwenden Sie das Element *Alle gewählten Objekte anzeigen* aus der Dropdown-Liste unten links im Dialogfeld der Attributtabelle. Diese Liste bietet die folgenden Filter:

- *Alle Objekte anzeigen*
- *Alle gewählten Objekte anzeigen* - wie *Attributtabelle öffnen (Gewählte Objekte)* aus dem *Layer* Menü oder der *Attributwerkzeuglseiste* oder Drücken von `Umschalt+F6`
- *Auf der Karte sichtbare Objekte anzeigen* - wie *Attributtabelle öffnen (Sichtbare Objekte)* aus dem *Layer* Menü oder der *Attributwerkzeuglseiste* oder Drücken von `Strg+F6`
- *Gegen Restriktionen verstoßende Objekte anzeigen* - Objekte werden so gefiltert, dass nur diejenigen angezeigt werden, die nicht erfüllte *Einschränkungen* haben. Je nachdem, ob es sich um eine harte oder weiche Einschränkung handelt, werden die nicht erfüllten Feldwerte in dunkel- bzw. hellorangen Zellen angezeigt.
- *Bearbeitet und neue Objekte anzeigen* - wie *Attributtabelle öffnen (Geänderte und neue Objekte)* aus dem *Layer* Menü oder der *Attributwerkzeuglseiste*
- *Felderfilter* - erlaubt dem Benutzer, nach dem Wert eines Feldes zu filtern: wählen Sie eine Spalte aus einer Liste, geben Sie einen Wert ein oder wählen Sie einen Wert und drücken Sie `Enter`, um zu filtern. Es werden dann nur die Objekte in der Attributtabelle angezeigt, die dem Ausdruck `num_field = value` oder `string_field ilike '%value%` entsprechen. Bei Zeichenketten können Sie *Case sensitive* ankreuzen, um Groß- und Kleinschreibung zu berücksichtigen.
- *Fortgeschrittener Filter (Ausdruck)* - Öffnet den Dialog zur Erstellung von Ausdrücken. Darin können Sie *komplexe Ausdrücke* erstellen, um Tabellenzeilen abzugleichen. Sie können zum Beispiel die Tabelle nach mehr als einem Feld filtern. Nach der Anwendung wird der Filterausdruck unten im Formular angezeigt.
- *Filterausdrücke gespeichert* ►: eine Verknüpfung zu *gespeicherten Ausdrücken*, die häufig zum Filtern Ihrer Attributtabelle verwendet werden.

Darüber hinaus ist es möglich, *Objekte über ein Formular zu filtern*.

Bemerkung

Durch das Herausfiltern von Datensätzen aus der Attributtabelle werden Objekte nicht aus dem Layer entfernt; sie werden lediglich vorübergehend aus der Tabelle ausgeblendet und können über die Kartenansicht oder durch

Entfernen des Filters wieder aufgerufen werden. Wenn Sie Objekte grundsätzlich aus dem Layer ausblenden wollen, verwenden Sie die *Abfrageerstellung*.

Tipp

Aktualisierung der Datenquellenfilterung mit `Auf der Karte sichtbare Objekte anzeigen`

Wenn aus Leistungsgründen die in der Attributtabelle angezeigten Objekte beim Öffnen auf diejenigen begrenzt sind, die in der Kartenansicht sichtbar sind (siehe *Datenquellen-Optionen* für eine Anleitung), wird durch die Auswahl von *Auf der Karte sichtbare Objekte anzeigen* die räumliche Begrenzung aktualisiert.

Filterausdrücke speichern

Ausdrücke, die Sie für die Filterung von Attributtabellen verwenden, können für weitere Aufrufe gespeichert werden. Wenn Sie die Einträge *Feldfilter* oder *Fortgeschrittener Filter (Ausdruck)* verwenden, wird der verwendete Ausdruck in dem Textfeld im unteren Bereich des Attributtabellendialogs angezeigt. Drücken Sie die Schaltfläche ☆ Ausdruck mit Text als Name speichern neben dem Textfeld, um den Ausdruck im Projekt zu speichern. Über das Dropdown-Menü neben der Schaltfläche können Sie den Ausdruck auch unter einem benutzerdefinierten Namen speichern (*Ausdruck speichern als …*). Sobald ein gespeicherter Ausdruck angezeigt wird, wird die Schaltfläche ★ aktiviert und über das Dropdown-Menü besteht die Möglichkeiten, den Ausdruck und ggf. den Namen zu bearbeiten (*Ausdruck bearbeiten*) oder man kann den **:guelabel:`gespeicherten Ausdruck löschen`**.

Gespeicherte Filterausdrücke werden im Projekt gespeichert und sind über das Menü *Filterausdrücke gespeicherte* in der Attributtabelle verfügbar. Sie unterscheiden sich von den *Benutzerausdrücken*, die von allen Projekten des aktiven Benutzerprofils gemeinsam genutzt werden.

Filterung und Auswahl von Objekten über Formulare

Wenn Sie auf die Schaltfläche Objekte mit Formular wählen/filtern klicken oder `Strg+F` drücken, wechselt der Dialog für die Attributtabelle in die Formularansicht und für jedes Feld wird ein eigenes Suchfeld angezeigt.

Von diesem Punkt an ähnelt die Funktionalität dieses Werkzeugs derjenigen, die in *Objekte nach Wert auswählen* beschrieben ist, wo Sie Beschreibungen aller Operatoren und Auswahlmodi finden können.

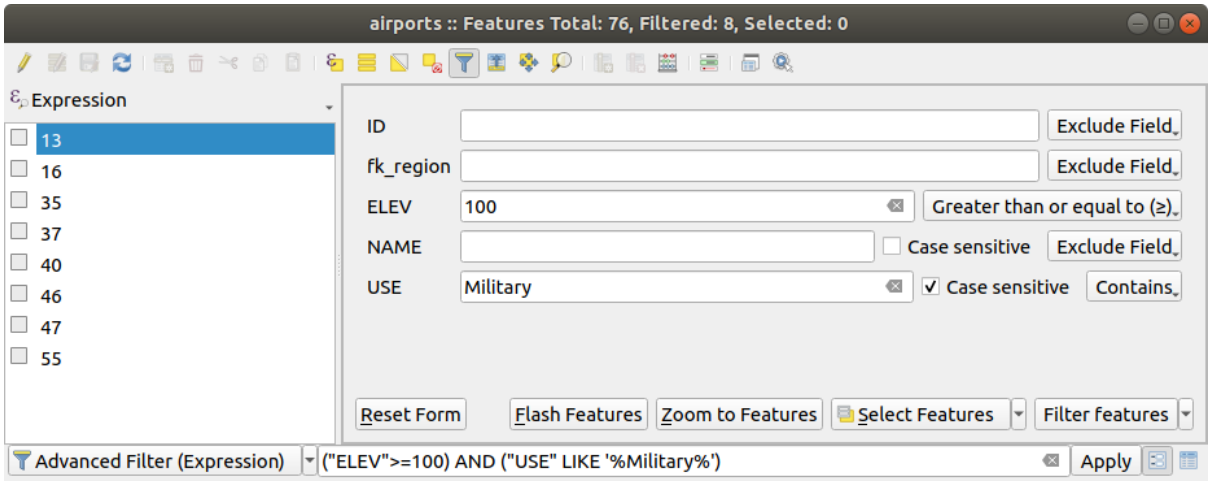


Abb. 12.73: Attributtabelle gefiltert durch das Filterformular

Bei der Auswahl / Filterung von Objekten aus der Attributtabelle gibt es unten rechts eine Schaltfläche *Objekte filtern*, mit der Filter definiert und verfeinert werden können. Ihre Verwendung löst die Option *Fortgeschrittener Filter (Ausdruck)* aus und zeigt den entsprechenden Filterausdruck in einem editierbaren Textfeld am unteren Rand des Formulars an.

Wenn es bereits gefilterte Objekte gibt, können Sie den Filter über die Dropdown-Liste neben der Schaltfläche *Objekte filtern* verfeinern. Die Optionen sind:

- *Filter innerhalb („AND“)*
- *Filter erweitern („OR“)*

Um den Filter zu löschen, wählen Sie entweder die Option *Alle Objekte anzeigen* aus dem Pulldown-Menü unten links, oder löschen Sie den Ausdruck und klicken Sie auf *Anwenden* oder drücken Sie `Eingabe`.

Mehr Aktionen für Objekte

Benutzer haben mehrere Möglichkeiten, Objekte in einer Attributtabelle zu bearbeiten. Klicken Sie mit der rechten Maustaste in eine Zelle und Sie können:

- *Alle wählen* (`Ctrl+A`) wählt alle Objekte des Layers aus;
- Kopieren Sie den Inhalt einer Zelle in die Zwischenablage mit *Zellinhalt kopieren*;
- *Zum Objekt zoomen* ohne es vorher auswählen zu müssen;
- *Zum Objekt bewegen*, ohne es vorher auswählen zu müssen;
- *Objekt aufleuchten lassen*, um es in der Kartenansicht hervorzuheben;
- *Formular öffnen*: schaltet die Attributtabelle in die Formularansicht um, wobei der Fokus auf dem angeklickten Objekt liegt.
- Zeigt eine *Liste der Aktionen* an, die zuvor auf der Registerkarte *Layereigenschaften* ► *Aktionen* aktiviert wurde.

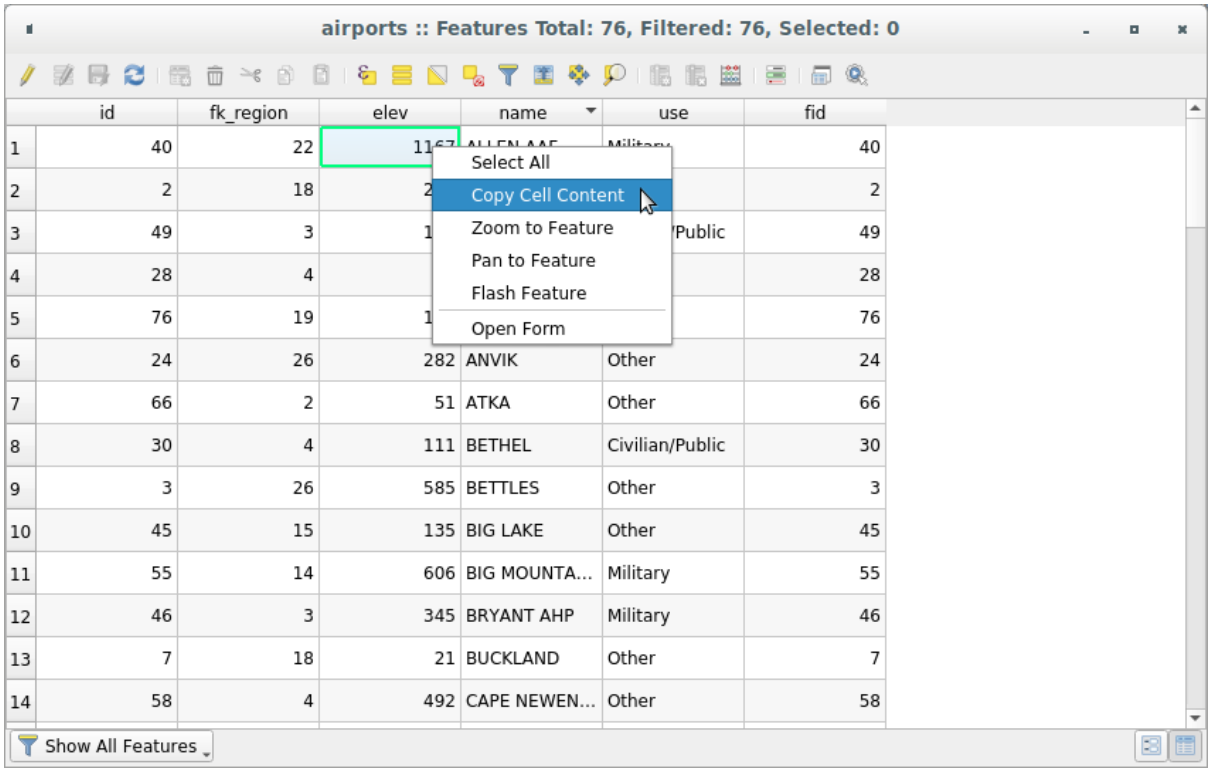


Abb. 12.74: Schaltfläche Zellinhalt kopieren

Wenn Sie Daten aus Attributtabellen in externen Programmen (wie Excel, LibreOffice oder einer benutzerdefinierten Webanwendung) verwenden möchten, wählen Sie eine oder mehrere Zeilen aus und verwenden Sie die Schaltfläche Ausgewählte Objekte in Zwischenablage kopieren oder drücken Sie `Strg+C`. Darüber hinaus können Sie im Menü *Einstellungen* ► *Optionen* ► *Datenquellen* mit der Option *Elemente kopieren als* das Format für das Einfügen festlegen. Weitere Details finden Sie unter *Datenquellen*.

12.2.4 Bearbeiten von Attributwerten

Um Daten in einer Attributtabelle zu ändern, müssen Sie zunächst den Layer zur Bearbeitung umschalten. Drücken Sie die Schaltfläche Bearbeitungsmodus umschalten. Abhängig vom Geometrietyp des Layers und dem Status der Zwischenablage werden einige weitere Werkzeuge in der oberen Symbolleiste der Attributtabelle aktiviert.

Die Bearbeitung von Attributwerten kann dann wie folgt erfolgen:

- den neuen Wert direkt in die Zelle eintippen, unabhängig davon, ob sich die Attributtabelle in der Tabellen- oder Formularansicht befindet. Änderungen werden also Zelle für Zelle, Objekt für Objekt vorgenommen;
- Mit dem *Feldrechner*: aktualisieren Sie in einer Zeile ein Feld, das bereits existieren kann oder aber für mehrere Objekte erstellt werden soll. Es können auch virtuelle Felder erstellt werden;
- Verwendung der *Berechnungsleiste*: wie oben, aber nur für das vorhandene Feld;
- oder mit dem *Mehrfachbearbeitungsmodus*: in einer Reihe mehrere Felder für mehrere Objekte aktualisieren.

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 *Objekte ausschneiden, kopieren und einfügen*.

Verwendung des Feldrechners

Mit der Schaltfläche Feldrechner in der Attributtabelle können Sie Berechnungen auf der Grundlage vorhandener Attributwerte oder definierter Funktionen durchführen, um z.B. Länge oder Fläche von geometrischen Objekten zu berechnen. Die Ergebnisse können verwendet werden, um ein bestehendes Feld zu aktualisieren, oder in ein neues Feld geschrieben werden (das auch ein *virtuelles Feld* sein kann).

Der Feldrechner ist auf jedem Layer verfügbar, der die Bearbeitung unterstützt. Wenn Sie auf das Feldrechner-Symbol klicken, öffnet sich das Dialogfeld (siehe Abb. 12.75). Wenn sich der Layer nicht im Bearbeitungsmodus befindet, wird eine Warnung angezeigt und die Verwendung des Feldrechners führt dazu, dass der Layer in den Bearbeitungsmodus versetzt wird, bevor die Berechnung durchgeführt wird.

Basierend auf dem Dialogfeld *Ausdruckseditor* bietet das Feldrechner-Dialogfeld eine vollständige Schnittstelle, um einen Ausdruck zu definieren und ihn auf ein bestehendes oder neu erstelltes Feld anzuwenden. Um den Feldberechnungsdialog zu verwenden, müssen Sie auswählen, ob Sie:

1. Berechnung auf den gesamten Layer oder nur auf ausgewählte Objekte anwenden
2. ein neues Feld für die Berechnung anlegen oder ein bestehendes aktualisieren.

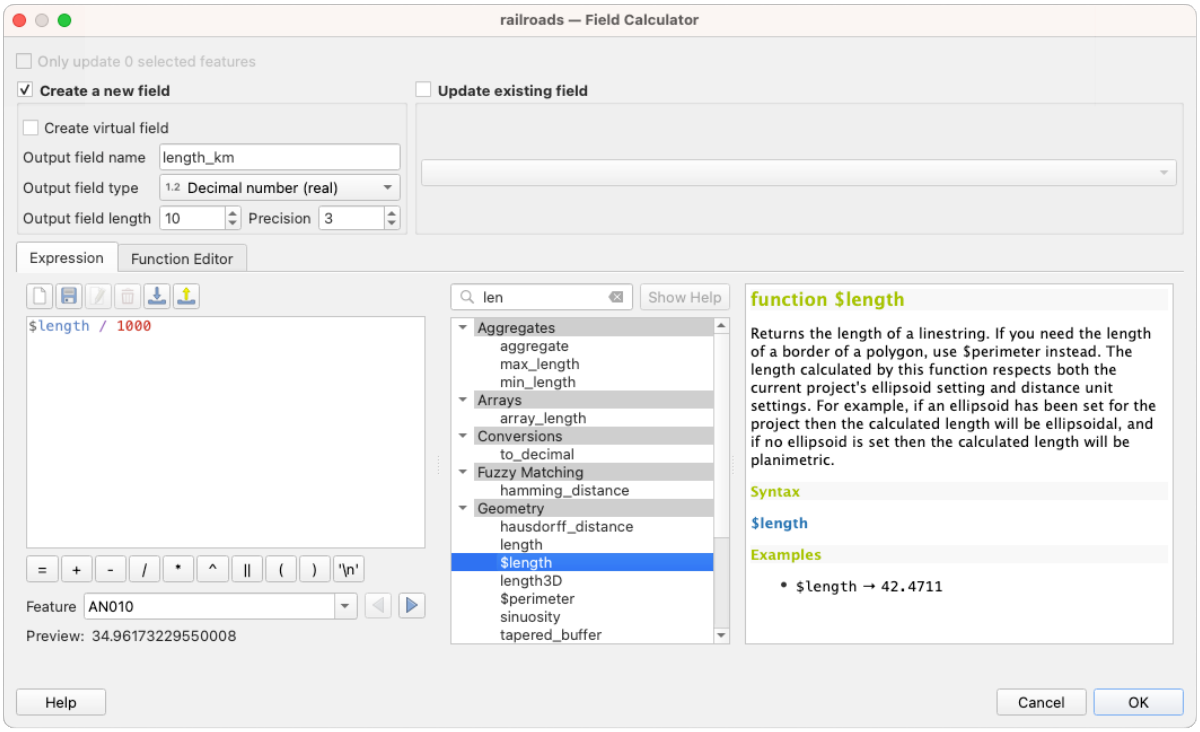


Abb. 12.75: Feldrechner

Wenn Sie ein neues Feld hinzufügen möchten, müssen Sie einen Feldnamen, einen Feldtyp (Integer, Real, Datum oder String) und, falls erforderlich, die gesamte Feldlänge und die Feldgenauigkeit eingeben. Wenn Sie zum Beispiel eine Feldlänge von 10 und eine Feldgenauigkeit von 3 wählen, bedeutet dies, dass Sie 7 Stellen vor dem Punkt und 3 Stellen für den Dezimalteil haben.

Ein kurzes Beispiel zeigt wie der Feldrechner arbeitet, wenn Sie den Reiter *Ausdruck* verwenden. Wir wollen die Länge des ‚railroads' Layers aus dem QGIS Beispieldatensatz in km berechnen:

1. Laden Sie das Shape `railroads.shp` in QGIS und öffnen Sie die den Dialog Attributtabelle öffnen.
2. Klicken Sie auf Bearbeitungsmodus umschalten und öffnen Sie den Feldrechner Dialog.
3. Wählen Sie das *Neues Feld anlegen* Kontrollkästchen um die Berechnungen in ein neues Feld zu speichern.
4. Setzen Sie *Ausgabefeldname* auf `length_km`.
5. Wählen Sie `Dezimalzahl (real)` als *Ausgabefeldtyp*
6. Setzen Sie den *Ausgabefeldlänge* auf `10` und die *Genauigkeit* auf `3`
7. Doppelklicken Sie auf `$length` in der Gruppe *Geometrie*, um die Länge der Geometrie in das *Ausdruck*-Feld einzufügen (Sie werden eine Vorschau der Ausgabe unter dem *Ausdruck*-Feld sehen, die in Echtzeit aktualisiert wird, während der Ausdruck zusammengestellt wird).
8. Vervollständigen Sie den Ausdruck, indem Sie `/ 1000` in das Feld Feldberechnungsausdruck eingeben und auf *OK* klicken.
9. In der Attributtabelle ist nun ein neues Feld *length_km* zu finden.

Ein virtuelles Feld erstellen

Ein virtuelles Feld ist ein Feld, das auf einem Ausdruck basiert, der im laufenden Betrieb berechnet wird, d. h. sein Wert wird automatisch aktualisiert, sobald sich ein zugrunde liegender Parameter ändert. Der Ausdruck gilt für alle Objekte im Layer und wird einmalig festgelegt. Sie müssen das Feld nicht mehr jedes Mal neu berechnen, wenn sich die zugrunde liegenden Werte ändern. Sie können beispielsweise ein virtuelles Feld verwenden, wenn Sie beim Digitalisieren von Objekten eine Fläche auswerten müssen oder um automatisch eine Dauer zwischen Daten zu berechnen, die sich ändern können (z. B. mit der Funktion `now()`).

Die Erstellung eines virtuellen Feldes erfolgt über den Dialog *Feldrechner* und folgt dem *gleichen Verfahren* wie für reguläre Felder. Sie müssen nur die Option *Virtuelles Feld erstellen* aktivieren und einen Feldtyp verwenden, der mit den Daten kompatibel ist, die Ihr Ausdruck erzeugen würde.

AusdruckAusdruckSchaltflächeAusdruckDie Bearbeitung eines virtuellen Feldes erfolgt über den Reiter *Felder* des Layereigenschaften-Dialogs (siehe *Eigenschaften Felder*). Der Ausdruck, der das Feld definiert, wird in der Spalte *Kommentar* angezeigt, und das Drücken der Schaltfläche daneben öffnet das Ausdruckseditor-Fenster zur Bearbeitung.

Bemerkung

Virtuelle Felder benutzen

- Ein Feld kann nur bei seiner Erstellung als virtuelles Feld festgelegt werden.
- Virtuelle Felder sind nicht dauerhaft in den Layerattributen, was bedeutet, dass sie in der Projektdatei gespeichert und verfügbar sind, in der sie gespeichert wurden.

Verwendung der Feldberechnungsleiste

Während der Feldrechner immer verfügbar ist, ist die Leiste für die schnelle Feldberechnung oben in der Attributtabelle nur sichtbar, wenn sich der Layer im Bearbeitungsmodus befindet. Dank des direkten Zugriffs auf den Ausdruckseditor bietet sie einen schnelleren Zugang zur Bearbeitung eines bereits vorhandenen Feldes:

1. Wählen Sie in der Dropdown-Liste das zu aktualisierende Feld aus.
2. Füllen Sie das Textfeld mit einem Wert, einem Ausdruck, den Sie direkt schreiben oder über die Schaltfläche Ausdruck erstellen.
3. Klicken Sie je nach Bedarf auf die Schaltfläche *Alle aktualisieren*, *Gewählte aktualisieren* oder *Gefilterte aktualisieren*.

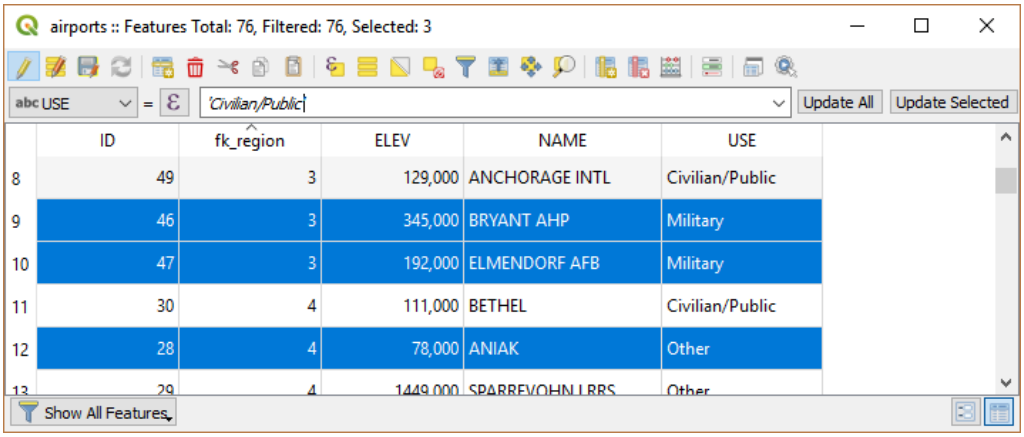


Abb. 12.76: Feldberechnungsleiste

Mehrere Felder bearbeiten

Im Gegensatz zu den vorherigen Werkzeugen können im Multibearbeitungsmodus mehrere Attribute verschiedener Objekte gleichzeitig bearbeitet werden. Wenn der Layer zur Bearbeitung umgeschaltet wird, sind die Mehrfachbearbeitungsfunktionen zugänglich:

- mit der Schaltfläche Mehrfachbearbeitungsmodus umschalten aus der Symbolleiste im Dialog der Attributtabelle;
- oder über das Menü *Bearbeiten* :menuselection:` Attribute bearbeiten –> Attribute der gewählten Objekte ändern`.

Bemerkung

Im Gegensatz zum Werkzeug aus der Attributtabelle erhalten Sie mit der Option *Bearbeiten* ► *Attribute bearbeiten* ► *Attribute der gewählten Objekte ändern* einen vereinfachten Dialog zum Ausfüllen von Attributänderungen. Daher ist die Auswahl von Objekten vor der Ausführung erforderlich.

Um mehrere Felder in einer Reihe zu bearbeiten:

1. Wählen Sie die Objekte aus, die Sie bearbeiten möchten.
2. Klicken Sie in der Symbolleiste der Attributtabelle auf die Schaltfläche . Dadurch wird der Dialog in die Formularansicht umgeschaltet. Die Auswahl der Objekte kann auch in diesem Schritt noch erfolgen.
3. Auf der rechten Seite der Attributtabelle werden die Felder (und Werte) der ausgewählten Objekte angezeigt. Neben jedem Feld erscheinen neue Widgets, die die Anzeige des aktuellen Mehrfachbearbeitungsstatus ermöglichen:
   - Das Feld enthält verschiedene Werte für ausgewählte Objekte. Es wird leer angezeigt und jedes Objekt behält seinen ursprünglichen Wert. Sie können den Wert des Feldes in der Dropdown-Liste des Widgets zurücksetzen.
   - Alle ausgewählten Objekte haben den gleichen Wert für dieses Feld und der im Formular angezeigte Wert wird beibehalten.
   - Das Feld wurde bearbeitet und der eingegebene Wert wird auf alle ausgewählten Objekte angewendet. Oben im Dialogfeld wird eine Meldung angezeigt, in der Sie aufgefordert werden, Ihre Änderung entweder zu übernehmen oder zurückzusetzen.

   Wenn Sie auf eines dieser Widgets klicken, können Sie entweder den aktuellen Wert für das Feld festlegen oder den ursprünglichen Wert wiederherstellen, d. h. Sie können Änderungen für jedes einzelne Feld zurücknehmen.

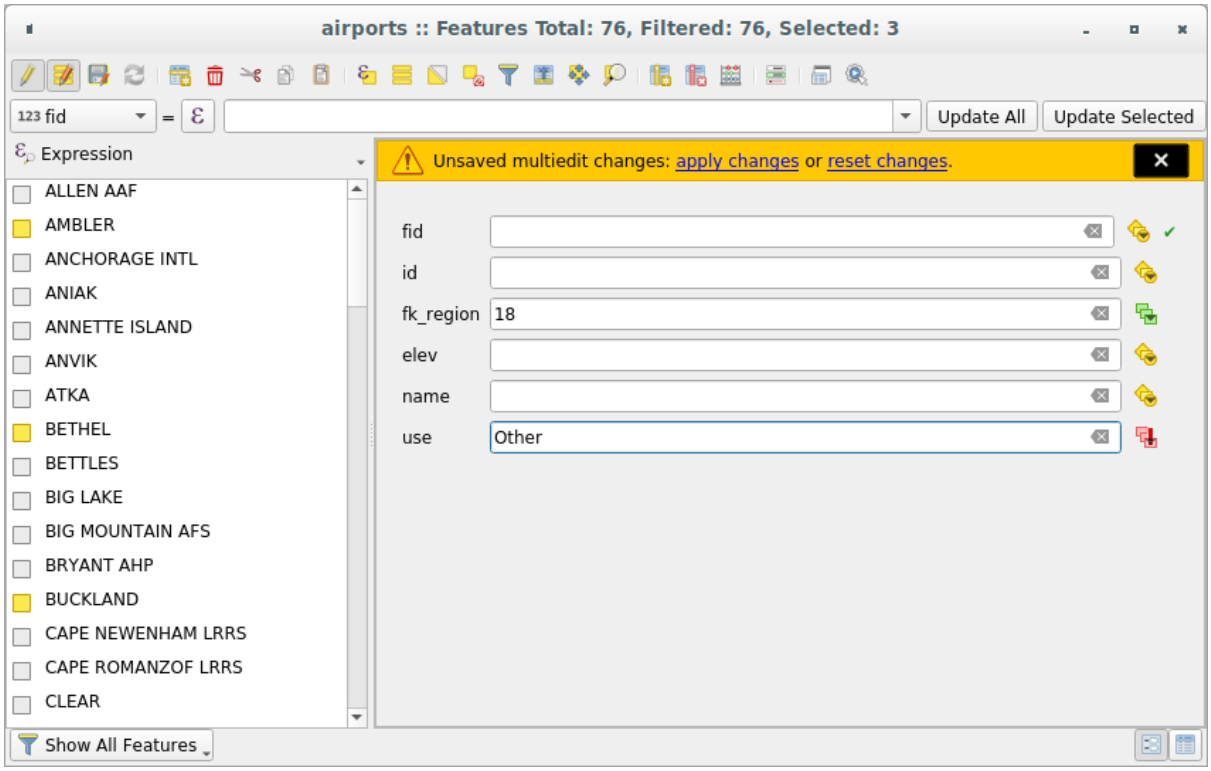


Abb. 12.77: Bearbeitung von Feldern für mehrere Objekte

4. Nehmen Sie die gewünschten Änderungen an den Feldern vor.
5. Klicken Sie auf **anwenden** im oberen Nachrichtentext oder auf ein anderes Objekt im linken Bedienfeld.

Die Änderungen werden auf **alle ausgewählten Objekte** angewendet. Wenn kein Objekt ausgewählt ist, wird die gesamte Tabelle mit Ihren Änderungen aktualisiert. Änderungen werden als ein einziger Bearbeitungsbefehl durchgeführt. Wenn Sie also Rückgängig drücken, werden die Attributänderungen für alle ausgewählten Objekte auf einmal zurückgenommen.

Bemerkung

Der Mehrfachbearbeitungsmodus ist nur für automatisch generierte und Drag & Drop-Formulare verfügbar (siehe *Anpassen eines Formulars für Ihre Daten*); er wird nicht von benutzerdefinierten Formularen unterstützt.

12.2.5 Erkundung von Attribute mit dem Objekte-abfragen-Werkzeug

Das Werkzeug *Objekte abfragen* kann verwendet werden, um alle Attribute eines Objekts in der Kartenansicht anzuzeigen. Es ist eine schnelle Möglichkeit, alle Daten zu sehen und zu überprüfen, ohne sie in der Attributtabelle suchen zu müssen.

Um das Werkzeug *Objekte abfragen* für Vektor-Layer zu verwenden, gehen Sie wie folgt vor:

1. Wählen Sie den Vektorlayer im „Layer“-Bedienfeld aus.
2. Klicken Sie auf das Werkzeug *Objekte abfragen* in der Werkzeugleiste oder drücken Sie `Strg+Umschalt+I`.
3. Klicken Sie auf ein Objekt in der Kartenansicht.

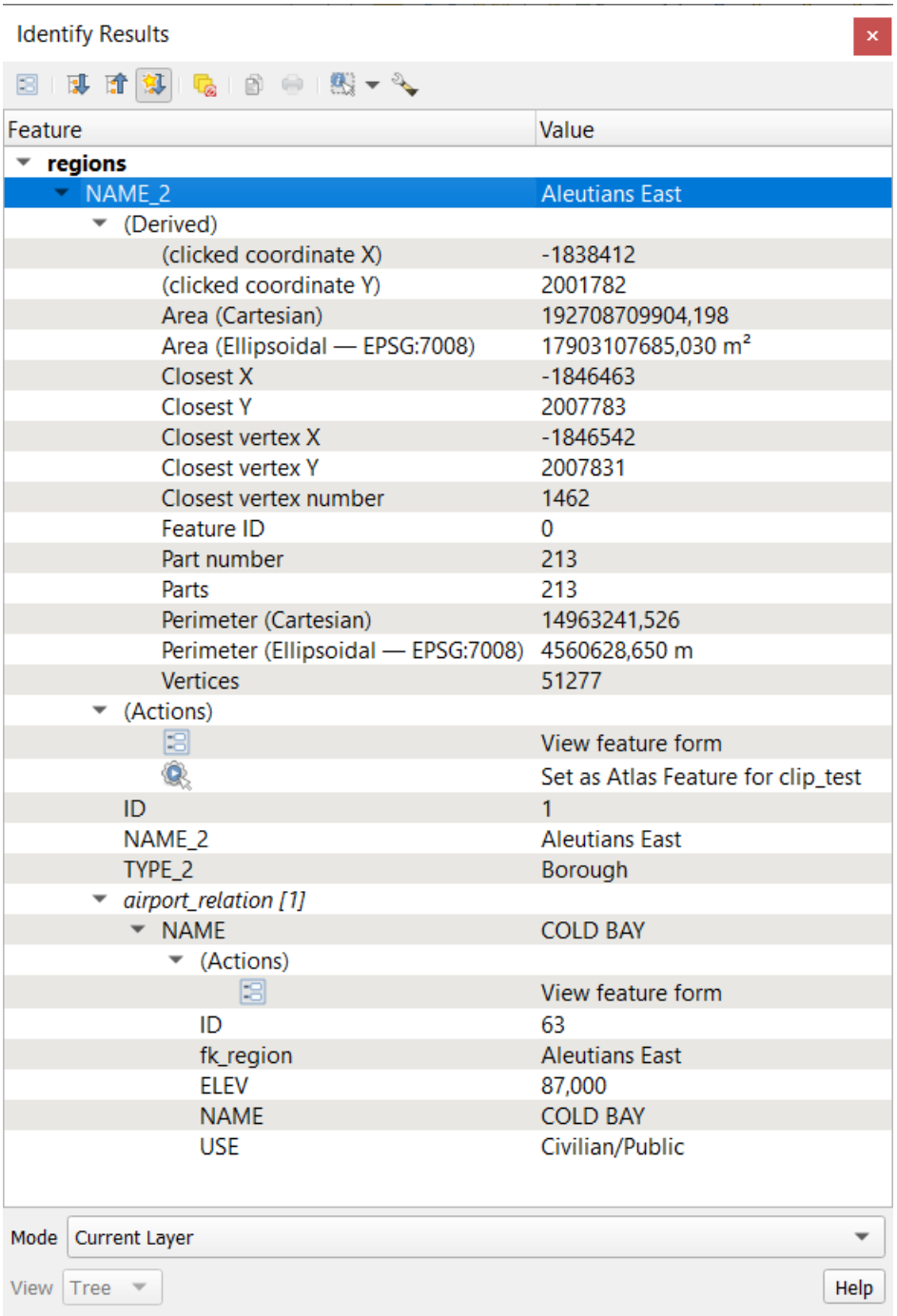


Abb. 12.78: Identify Results dialog

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:

- Abschnitt **Abgeleitet** - das sind die Informationen, die aus anderen Informationen im Layer berechnet oder abgeleitet wurden. Zum Beispiel, die Fläche eines Polygons oder die Länge einer Linie. Allgemeine Informationen, die in diesem Abschnitt gefunden werden können:
  - 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.
  - Abhängig vom Geometrietyp und wenn ein Ellipsoid im Dialog *Projekteigenschaften* (*Allgemein* ► *Messungen*) eingestellt ist, werden ellipsoidische Werte für Länge, Umfang oder Fläche in den

angegebenen Einheiten angezeigt.

- Die Anzahl der Geometrieteile im Objekt und die Nummer des angeklickten Teils.
- Die Anzahl der Knotenpunkte des Objekts.
- 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 *Aktionen*).
- **Datenattribute**: Dies ist die Liste von Attributfeldern und Werten für das Objekt, welches angeklickt wurde.

> **Bemerkung**
>
> 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 Editierfunktionen

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.

> **Bemerkung**
>
> Die Vorgehensweise GRASS Layer zu bearbeiten ist anders - siehe Abschnitt *Digitalisieren und Editieren eines GRASS Vektorlayers* für Details.

Achtung

Zeitgleiches Editieren

QGIS verfolgt nicht, ob jemand anderes das gleiche Objekt zur gleichen Zeit wie Sie bearbeitet. Die letzte Person, die die Änderungen speichert, gewinnt.

Tipp

Bearbeitungen validieren

Sie können die fortwährende Validierung für einzelne Layer über den Tab *Layereigenschaften* ► *Digitalisieren* aktivieren. Mehr dazu unter *Digitalisieren*.

12.3.1 Einrasttoleranz und Suchradius einstellen

Unter dem Menüpunkt *Einstellungen* ► *Optionen…* ► *Digitalisieren* bietet QGIS eine Anzahl von Parametern, mit der Sie das Standardverhalten der Editierwerkzeuge einstellen können. Weitere Informationen finden Sie unter *Digitalisieren*.

Damit Sie die Geometrien von Vektorlayern optimal und akkurat bearbeiten können, ist es notwendig, dass Sie geeignete Werte für die Einrasttoleranz und den Suchradius für die Stützpunkte eines Objekts einstellen. Die Gruppe *Einrasten* bietet entsprechende Optionen, nämlich die Einstellung der Einrasttoleranz und des Suchradius für die Stützpunktbearbeitung.

- Einrasttoleranz: Wenn Sie einen neuen Stützpunkt digitalisieren oder einen vorhandenen verschieben, ist die Einrasttoleranz diejenige Distanz, innerhalb derer QGIS nach dem nächsten Stützpunkt oder Segment sucht, um Ihren Stützpunkt damit zu verbinden. Wenn Sie sich außerhalb der Einrasttoleranz befinden, wird QGIS den Stützpunkt an derjenigen Stelle platzieren, an der Sie die Maustaste loslassen, anstatt ihn an einem vorhandenen Stützpunkt oder Segment einzurasten.

  Die Einstellung für die Einrasttoleranz beeinflusst alle Werkzeuge, die mit Einrasten arbeiten und wird standardmäßig auf neue Layer und Projekte angewendet. Auf Layerebene kann sie jedoch geändert werden (siehe *Optionen für das Einrasten und Digitalisieren*).

- Suchradius: Der *Suchradius für Stützpunktbearbeitung* ist diejenige Distanz, die QGIS benutzt, um nach dem zu selektierenden Stützpunkt zu `suchen`, wenn Sie in die Karte klicken. Befindet sich der Klick nicht innerhalb des Suchradius, wird QGIS keinen Stützpunkt für die Bearbeitung auswählen.

Einrasttoleranz und Suchradius stellen Sie in `Karteneinheiten` oder `Pixeln` ein, so dass Sie vielleicht experimentieren müssen, um sie passend einzustellen. Wenn Sie eine zu große Toleranz angeben, fängt QGIS vielleicht den falschen Stützpunkt, besonders wenn Sie es mit einer großen Anzahl von Stützpunkten innerhalb kurzer Distanz zu tun haben. Stellen Sie den Suchradius zu klein ein, wird nichts zum Verschieben gefunden.

12.3.2 Optionen für das Einrasten und Digitalisieren

Globale *Einstellungen für das Einrasten und Digitalisieren* (Einrastmodus, Toleranzwert und Einheiten…) können im Projekt über das Menü *Projekt* ► *Einrastoptionen…* überschrieben werden. In den *Schnapp- und Digitalisierungsoptionen* können Sie auch einige andere Eigenschaften konfigurieren (Layer, Skalierungsgrenze, Topologie…) Die *Schnapp-Symbolleiste* bietet Zugriff auf die meisten dieser Objekte.

Standardmäßig ist Einrasten in einem Projekt deaktiviert, bis Sie auf den Knopf Einrasten einschalten oder auf die Taste `S` drücken. Sie können Einrastmodus, Toleranz und Einheiten auch in dieser Werkzeugleiste konfigurieren.

Einrastoptionen

Sie haben drei Möglichkeiten, um die Layer, auf die eingrastet werden soll, auszuwählen:

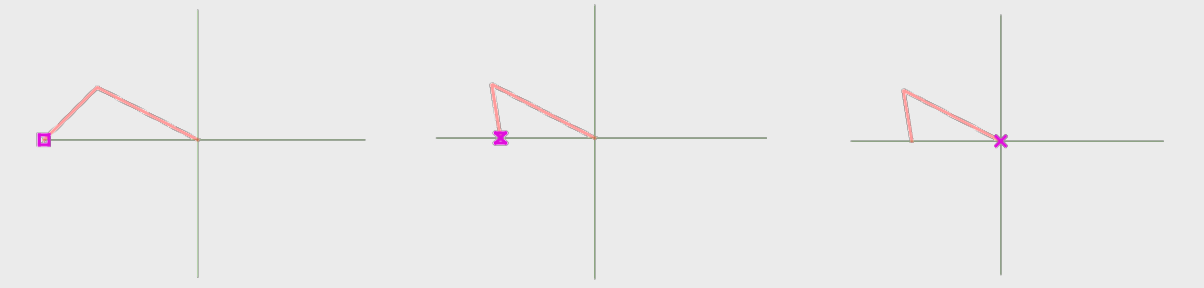
- *Alle Layer*: eine schnelle und einfache Einstellung für alle sichtbaren Layer im Projekt, so dass der Zeiger alle Stützpunkte und/oder Segmente fängt. In den meisten Fällen reicht es aus, diesen Fangmodus zu verwenden; passen Sie aber auf, wenn Sie diesen Modus in Projekten mit vielen Vektorlayern benutzen, weil es die Anwendung deutlich verlangsamen kann.
- *Aktueller Layer*: nur der aktuelle Layer wird genutzt, eine praktische Möglichkeit die Topologie des Layers abzusichern, der bearbeitet wird.
- Mit *Erweiterte Konfiguration*: können Sie für jeden Layer den Typ, die Toleranz und die Einheiten des Einrastens, die Überlappung sowie den Einrastmaßstab einschalten und anpassen (siehe Abb. 12.79). Wenn Sie einen Layer editieren und möchten seine Stützpunkte an einem anderen Layer einrasten, stellen Sie bitte sicher, dass der andere Layer angehakt ist und erhöhen Sie die Einrasttoleranz. Ist ein Layer in den Projekteinrasteinstellungen nicht angehakt, können Sie auf ihn nicht einrasten.

Beim Verschieben oder Erstellen von Scheitelpunkten können Sie sich für die folgenden Fangmodi entscheiden:

- *Stützpunkt*
- *Segment*: rastet entlang einer Linie oder eines Polygonumfangs ein. Wenn die topologische Bearbeitung aktiviert ist, wird ein neuer Stützpunkt an der Fangposition hinzugefügt.
- *Fläche*: garantiert, dass der Fangpunkt irgendwo auf der Fläche eines Polygons liegt, nicht notwendigerweise auf seiner Grenze
- *Zentroid*: rastet auf den Schwerpunkt der Geometrie eines Objekts ein. Im Falle einer mehrteiligen Geometrie kann der Zielpunkt von den vorhandenen Teilen verschieden sein.
- *Mitte der Segmente* bei Linien- oder Polygonobjekten
- *Linienendpunkte*: rastet am ersten oder letzten Scheitelpunkt eines jeden Teils einer Linie oder eines Polygonobjekts ein.

QGIS wird die fangbaren Punkte unterschiedlich darstellen, je nachdem welcher Einrastmodus aktiv ist:

Tab. 12.2: Icons der verschiedenen Einrast-Optionen



Einrasten auf einen Stützpunkt: Quadrat	Einrasten auf Segment: Sanduhr	Einrasten auf Schnittpunkte: Kreuz

Sie können die Farbe dieser Symbole in den Einstellungen im Bereich *Digitalisieren* ändern.

Die Toleranz können Sie entweder in `Karteneinheiten` des Projekts oder in `Pixel` einstellen. Der Vorteil von `Pixel` ist, dass damit der Fangradius bei unterschiedlichen Kartenmassstäben konstant bleibt. 10 bis 12 px ist normalerweise ein guter Wert, der aber von der DPI-Zahl Ihres Bildschirms abhängig ist. Mit Karteneinheiten benutzen Sie für die Toleranz Realwelteinheiten. Wenn Sie z.B. eine Mindestdistanz zwischen Ihren Elementen sicherstellen wollen, hilft Ihnen diese Einstellung dabei, die Stützpunkte nicht zu dicht beieinander zu setzen.

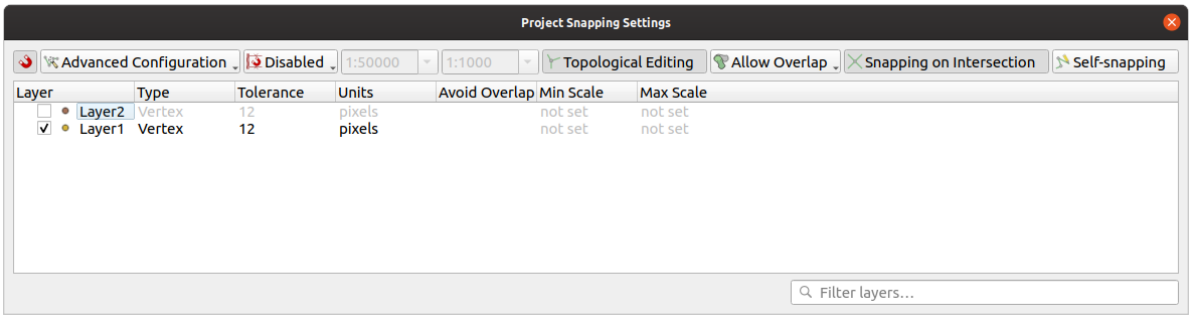


Abb. 12.79: Einrasteinstellungen (Erweiterte Konfiguration)

Bemerkung

In der Voreinstellung können nur sichtbare Objekte gefangen werden (d.h. Objekte, deren Stil dargestellt wird, mit Ausnahme von Layern, deren Symbolisierung „Keine Symbole“ ist). Sie können das Fangen unsichtbarer Objekte durch Anhaken von *Einrasten auf verborgene Objekte aktivieren* unter *Einstellungen* ► *Optionen* ► *Digitalisieren* anschalten.

Tipp

Einrasten voreinstellen

Sie können das Einrasten als Voreinstellung für alle neuen Projekte im Reiter *Einstellungen* ► *Optionen* ► *Digitalisierung* anschalten. Sie können auch den Einrastmodus, die Einrasttoleranz und die Einrasteinheiten voreinstellen, die dann im Dialog *Einrasteinstellungen* vorausgewählt sind.

Fang auf Schnittpunkte aktivieren

Eine andere Möglichkeit stellt das *Einrasten auf Schnittpunkte aktivieren* dar. Sie können damit auf Schnittpunkte der Geometrien von Layern einrasten, für die Einrasten aktiviert ist, selbst wenn es keinen Stützpunkt auf dem Schnittpunkt gibt.

Maßstabsabhängiges Einrasten setzen

In einigen Fällen kann das Einrasten sehr langsam werden. Dies wird häufig durch die Menge an Features in einigen Layern verursacht, für deren Berechnung und Verwaltung ein umfangreicher Index erforderlich ist. Einige Parameter sind vorhanden, um das Fangen nur zu ermöglichen, wenn sich die Kartenansicht innerhalb eines relevanten Maßstabsbereichs befindet. Dies erlaubt, die kostspielige Indexberechnung im Zusammenhang mit dem Fangen nur in einem Maßstab durchzuführen, der für das Zeichnen relevant ist.

Sie können die Maßstabsabhängigkeit beim Einrasten in *Projekt* ► *Einrastoptionen…* setzen. Die Maßstabsabhängigkeit st nur in der *Erweiterten Konfiguration* möglich.

Es stehen drei Modi für die Maßstabsabhängigkeit beim Einrasten zur Verfügung:

- *deaktiviert*: Das Einrasten ist aktiv, unabhängig davon, wie der aktuelle Kartenmaßstab ist. Dies ist die Voreinstellung.
- *Global*: Das Einrasten ist beschränkt und nur aktiv wenn der aktuelle Kartenmaßstab zwischen einem globalen Minimal- und Maximalwert liegt. Wenn Sie diesen Modus auswählen, werden zwei Eingabefelder aktiv, mit denen Sie die Maßstabsbereiche, in denen das Einrasten aktiv ist, konfigurieren können.

- *Je Layer*: Legen Sie den Maßstabsbereich für das Einrasten für jeden Layer einzeln fest. Wenn Sie diesen Modus auswählen, werden zwei Spalten aktiv, mit denen Sie die Maßstabsbereiche für jeden Layer einzeln konfigurieren können.

Bitte beachten Sie, dass der Minimal- und der Maximalmaßstab den QGIS-Konventionen folgt: Der Minimalmaßstab ist am weitesten „rausgezoomt“, während der Maximalmaßstab am weitesten „reingezoomt“ ist. Sollten Sie den Minmal- oder Maximalmaßstab auf „0“ oder „nicht gesetzt“ einstellen, wird die Einstellung nicht beachtet.

Selbsteinrasten

Mit der Einstellung Selbsteinrasten können Sie auf diejenige Geometrie einrasten, die Sie gerade bearbeiten. Zusammen mit dem *Bedienfeld erweiterte Digitalisierung*, haben Sie damit eine praktische Möglichkeit, neue Kanten relativ zu vorhandenen Kanten oder Stützpunkten zu digitalisieren. Bitte benutzen Sie Selbsteinrasten mit Vorsicht, da es ungültige Geometrien erzeugen kann.

Auf ein eigenes Gitter einrasten

Sie können die Einrastentfernung für jeden Layer individuell in den Layereigenschaften im Tab *Digitalisieren* einstellen. Indem Sie die Entfernung für die *Geometriegenauigkeit* einstellen aktivieren Sie ein Punktraster, dass sichtbar wird, sobald der Kartenmaßstab sich in einem für die Darstellung sinnvollen Bereich bewegt. Sie können nun auf die Gitterpunkte einrasten: alle Stützpunkte einer neuen oder geänderten Geometrie werden automatisch auf den nächstgelegenen Gitterpunkt eingerastet. Weitere Informationen bietet *Digitalisieren*.

12.3.3 Topologisches Editieren

Im Dialog *Einrasteinstellungen…* ` (*Projekt* ► *Einrastoptionen*) und in der *Einrastwerkzeugleiste* können Sie einige topolgische Funktionalitäten ein- und ausschalten.

Topologisches Editieren aktivieren

Die Schaltfläche Topologisches Editieren aktivieren unterstützt Sie beim Editieren und Pflegen von Objekten mit gemeinsamen Grenzen. Wenn diese Option eingeschaltet ist, ‚erkennt‘ QGIS gemeinsame Objektgrenzen; wenn Sie gemeinsame Stützpunkte oder Segmente verschieben, wird QGIS sie auch in den benachbarten Objektgeometrien verschieben.

Das topologische Editieren funktioniert mit Objekten aus verschiedenen Layern, solange die Layer sichtbar sind und sich im Bearbeitungsmodus befinden.

In Layern mit Z- oder M-Werten wird bei der topologischen Bearbeitung der Z- oder M-Wert des Scheitelpunkts auf der Grundlage des Werts der für die Verbindung verwendeten Kante interpoliert.

Kontrolle über die Überlappungen

Durch die Steuerung der Überlappung können Sie neue Objekte zeichnen, die bestehende im ausgewählten Layer nicht überlappen, dies beschleunigt die Digitalisierung von aneinander angrenzenden Polygonen. Das Verhalten können Sie mit der Überlappungseinstellung steuern. Drei Modi stehen zu Auswahl:

1. *Überlappung zulassen* (Standardeinstellung)
2. *Überlapppung auf aktiven Layer vermeiden*: verhindert jegliche Überlappung mit anderen Objekten des Bearbeitungslayers. Wenn Sie die neuen Geometrien so digitalisieren, dass sie ihre Nachbarflächen überlappen, wird QGIS überlappende Teile der neuen Geometrien abschneiden und sie auf die Grenzen der vorhandenen Objekte einrasten. Dadurch müssen Sie die gemeinsamen Stützpunkte auf der Grenzlinie nicht eigens digitalisieren.

3. *Erweiterter Konfiguration folgen*: damit können Sie die Überlappungseinstellungen für jeden Layer individuell im Modus *Erweiterte Konfiguration* setzen.

Bemerkung

Wenn die neue Geometrie vollständig durch vorhandene abgedeckt ist, wird sie gelöscht und und QGIS wird eine Fehlermeldung anzeigen.

Warnung

Benutzen Sie die Option *Schnittpunkte vermeiden* mit Vorsicht

Da diese Option neue überlappende Geometrien in jedem Polygonlayer zuschneidet, können Sie unerwartete Ergebnisse erhalten, wenn Sie es vergessen, sie auszuschalten, wenn Sie sie nicht mehr benötigen.

Automatische Spurverfolgung

Wenn Sie Erfassungs-Kartenwerkzeuge (Objekt hinzufügen, Teil hinzufügen, Ring hinzufügen, Objekte überarbeiten oder Objekte zerteilen) benutzen, müssen Sie normalerweise jeden einzelnen Stützpunkt des Objekts anklicken. Mit der automatischen Spurverfolgung können Sie den Digitalisierprozess beschleunigen, weil sie nicht mehr jeden Stützpunkt manuell setzen müssen:

1. Schalten Sie das Werkzeug Spurverfolgung über die Schaltfläche ein oder drücken Sie die Taste `T`.
2. *Fangen* Sie einen Stützpunkt oder ein Segment des Objekts, an dessen Grenze entlang Sie digitalisieren wollen.
3. Bewegen Sie die Maus über einen andern Stützpunkt oder ein anderes Segment, das sie fangen möchten. Anstelle der geraden Linie, die Sie überlicherweise erhalten, folgt das Gummiband dem Weg vom letztgefangenen Punkt bis zur aktuellen Mausposition.

   QGIS benutzt die Topologie des zugrundeliegenden Objekts, um den kürzesten Weg zwischen den beiden Punkten zu bestimmen. Die Spurverfolgung erfordert, dass das Einrasten in dem Layer, in dem sich das zu verfolgende Objekte befindet, eingeschaltet ist. Während Sie digitalisieren sollten Sie einen vorhanden Stützpunkt oder ein vorhandenes Segment fangen und sicherstellen, dass Anfangs- und Endpunkt durch vorhandene Objektgrenzen topologisch verbunden sind. Ansonsten ist QGIS nicht in der Lage sie zu verbinden und erzeugt eine gerade Verbindungslinie zwischen ihnen.
4. Wenn Sie klicken, plaziert QGIS alle Zwischenstützpunkte entlang des dargestellten Weges.

Klappen Sie die Spurverfolgung einschalten Schaltfläche auf und setzen Sie die *Offset*-Option, um einen Weg parallel zu den Objekten zu digitalisieren anstatt auf ihrer Grenze; ein positiver Wert verschiebt die neue Linie in Spurverfolgungsrichtung nach links und ein negativer Wert nach rechts.

Bemerkung

Justieren Sie den Kartenmaßstab oder die Fangeinstellungen für eine optimale Verfolgung

Wenn zu viele Objekte in der Karte gezeigt werden, ist Verfolgung möglicherweise deaktiviert, um potentielle langsame Verfolgung und große Speicherzuschläge zu vermeiden. Nach dem hereinzoomen oder dem ausblenden einiger Layer, ist die Funktion wieder aktiviert.

Bemerkung

Kein topologisches Editieren

Dieses Werkzeug fügt bestehenden Polygonen keine neuen Punkte hinzu, auch wenn *Topologisches Editieren* eingeschaltet ist. Falls die Geometriegenauigkeit des Bearbeitungslayers gesetzt ist, kann die sich ergebende Geometrie von der bestehenden abweichen.

Tipp

Schalten Sie die automatische Spurverfolgung schnell an oder aus, indem Sie `T` drücken

Durch Drücken der `T` Taste, können Sie die Spurverfolgung jederzeit an- und ausschalten, sogar während Sie ein Objekt digitalisieren. Damit ist es möglich, einige Teile eines Objekts mit aktivierter Spurverfolgung zu digitalisieren und andere nicht. Alle Werkzeuge verhalten sich normal, wenn die Spurverfolgung ausgeschaltet ist.

Tipp

Die Spurverfolgung in Kurvengeometrien umwandeln

Über *Einstellungen* ► *Optionen* ► *Digitalisieren* ► *Spurverfolgung* können Sie während des Digitalisierens Kurvengeometrien erzeugen. Siehe *Optionen Digitalisieren*.

12.3.4 Einen vorhandenen Layer editieren

In der Voreinstellung lädt QGIS Vektorlayer im reinen Lesemodus, um ungewolltes Editieren zu verhindern wenn Sie versehentlich die Maus verrutschen. Ansonsten können Sie aber alle Ebenen editieren, vorausgesetzt der Datenanbieter unterstützt das (siehe *Datenformate und Felder erkunden*) und die Datenquelle ist überhaupt schreibbar (d.h. ihre Dateien sind schreibbar).

Tipp

Beschränkung der Editierbarkeit von Layern innerhalb eines Projekts

Sie können jeden Layer in der Tabelle *Projekt* ► *Eigenschaften…* ► *Datenquellen* ► *Layereigenschaften* unabhängig vom Datenanbieter auf schreibgeschützt setzen. In einer Mehrbennutzerumgebung kann das praktisch sein, um zu verhindern, dass Nutzer fälschlicherweise einen Layer (z.B. ein Shapefile) editieren, und damit die Daten möglicherweise beschädigen. Beachten Sie aber, dass die Einstellung nur im aktuellen Projekt gültig ist.

Die Werkzeuge zum Bearbeiten von Vektorlayern sind in die beiden Werkzeugleisten Digitalisierung und Erweiterte Digitalisierung, die im Abschnitt *Erweiterte Digitalisierung* beschrieben wird, aufgeteilt. Sie können beides unter *Ansicht* ► *Werkzeugkästen* ► an- oder abwählen. Wenn Sie die Digitalisierungswerkzeugleiste verwenden, können Sie folgende Funktionen ausführen:

In den grundlegenden Digitalisierfunktionen sind folgende Funktionen verfügbar:

Tab. 12.3: Basis-Werkzeuge in der Digitalisierung-Werkzeugleiste für Vektorlayer

Werkzeug	Zweck	Werk	Zweck
	Änderungen in allen ausgewählten Layern gleichzeitig speichern, rückgängig machen oder abbrechen		Den Bearbeitungsstatus des ausgewählten Layers oder der ausgewählten Layer auf der Grundlage ihres aktuellen Status an- oder ausschalten.
	Die Änderungen im aktiven Layer speichern		
	Digitalisieren mit geraden Segmenten		Digitalisieren mit Kurvenlinien
	Freihand-Digitalisierung aktivieren		Polygon mit regelmäßiger Form digitalisieren
	Datensatz hinzufügen		Objekt hinzufügen: Punkt hinzufügen
	Objekt hinzufügen: Linie hinzufügen		Objekt hinzufügen: Polygon hinzufügen
	Knotenwerkzeug (alle Layer)		Knotenwerkzeug (aktueller Layer)
Stützpunkt-Editor anzeigen	Legen Sie fest, ob das Bedienfeld des Vertex-Editors automatisch geöffnet werden soll		Attribute aller gewählten Objekten gleichzeitig ändern
	Ausgewählte Objekte des aktiven Layers löschen		Ausgewählte Objekte des aktiven Layers ausschneiden
	Ausgewählte Objekte des aktiven Layers kopieren		Objekte in den aktiven Layer einfügen
	Änderungen im aktiven Layer rückgängig machen		Änderungen im aktiven Layer wiederholen

Beachten Sie, dass Sie während der Nutzung des Digitalisierungswerkzeugs, weiterhin in dem Kartenfenster *zoom or pan* können, ohne den Fokus auf das Werkzeug zu verlieren.

Beginnen Sie jeden Editiervorgang mit einem Click auf den Eintrag Bearbeitungsmodus umschalten den Sie entweder im Kontextmenü eines Layers, in der Attributtabelle, der Digitalisierungswerkzeugleiste oder dem Menü *Layer* finden.

Sobald der Layer im Bearbeitungsmodus ist, werden zusätzliche Werkzeuge verfügbar und Markierungen an den Eckpunkten alles Objekte erscheinen, bis Sie die *Markierungen nur für gewählte Objekte anzeigen* Option unter *Einstellungen* ► *Optionen…* ► *Digitalisierung* aktiviert haben.

Tipp

Regelmäßiges Sichern der Daten

Denken Sie daran Layeränderungen regelmäßig zu speichern. Dies überprüft auch dass Ihre Datenquelle alle Änderungen akzeptiert.

Techniken zur Bearbeitung von Geometrien

Wenn ein Geometrie-Zeichenwerkzeug (vor allem die Werkzeuge zum Hinzufügen, Teilen und Umformen von Objekten) für einen linien- oder polygonbasierten Layer aktiviert ist, können Sie die Technik zum Hinzufügen neuer Stützpunkte auswählen:

- Die Funktion :sup:`Mit Segment digitalisieren `: zeichnet ein gerades Segment, dessen Anfangs- und Endpunkte durch Linksklick definiert werden.
- Die Funktion Mit Kurve digitalisieren: zeichnet eine Kurvenlinie basierend auf drei aufeinanderfolgenden Knoten, die durch Linksklicks definiert werden (Start, Punkt entlang des Bogens, Ende). Wenn der Geometrietyp keine Kurven unterstützt, dann werden aufeinanderfolgende kleinere Segmente verwendet, um die Krümmung zu approximieren.
- Die Funktion Laufende Digitalisierung: zeichnet Linien im Freihandmodus, d.h. Knoten werden nach der Cursorbewegung in der Kartenansicht und einer *Streaming Tolerance* hinzugefügt. Die Streaming Tolerance definiert den Abstand zwischen aufeinanderfolgenden Scheitelpunkten. Zur Zeit ist die einzige unterstützte Einheit Pixel (`px`). Nur der beginnende Linksklick und der endende Rechtsklick sind in diesem Modus notwendig.
- Die Funktion Form digitalisieren: löst Werkzeuge auf der *Werkzeugleiste für Formen* aus, um ein Polygon einer regelmäßigen Form zu zeichnen.

Die gewählte Technik bleibt beim Wechsel zwischen den Digitalisierungswerkzeugen erhalten. Sie können jede der ersten drei Methoden kombinieren, während Sie die gleiche Geometrie zeichnen.

Objekte digitalisieren

Abhängig von der Art des Layers benutzen Sie eine der Schaltflächen Objekt hinzufügen, Punktobjekt hinzufügen, Linienobjekt hinzufügen oder Polygonobjekt hinzufügen in der Werkzeugleiste, um neue Objekte zum aktiven Layer hinzuzufügen.

Fügen Sie ein geometrieloses Objekt hinzu, indem Sie auf den Knopf Objekt hinzufügen klicken, danach tragen Sie im sich öffnenden Objektformular die Attributwerte ein.

Erzeugen Sie Objekte mit den Geometriewerkzeugen, indem Sie zunächst die Geometrie digitalisieren, danach geben Sie die Attributwerte ein. So digitalisieren Sie die Geometrie:

1. (Optional, da es die Standardeinstellung ist) Wählen Sie die Geometriezeichnungsmethode Digitalisieren mit Segment
2. Klicken Sie mit der linken Maustaste in den Bereich der Karte, an dem Sie den ersten Punkt Ihres neuen Objekts erzeugen möchten. Für Punktobjekte sollte das schon genügen und, falls benötigt, das Objektformular öffnen, in das Sie die Attributwerte eintragen.
3. Machen Sie bei Linien- oder Polygonobjekten weitere Klicks mit der linken Maustaste für jeden weiteren Punkt, den Sie erfassen möchten. Sie können sich auf die Optionen für das *Einrasten*, das *Einrasten auf ein eigenes Gitter* oder das Bedienfeld *Erweiterte Digitalisierung* stützen, um jeden Stützpunkt genau zu platzieren.

   Neben dem Zeichnen gerader Segmente zwischen Knoten, die Sie einzeln anklicken, können auch Linien und Polygone gezeichnet werden:

   - *Automatisch verfolgen*, wodurch die Digitalisierung beschleunigt wird. Dadurch werden aufeinanderfolgende gerade Linien zwischen den von Ihnen platzierten Scheitelpunkten erzeugt, die bestehenden Objekten folgen.
   - freihändig digitalisiert werden, indem Sie `R` drücken oder Stream Digitizing aktivieren.
   - als Kurve gezeichnet, indem Sie `Strg+Umschalt+G` drücken oder Digitalisieren mit Kurve aktivieren.

> **Bemerkung**
>
> Beim Digitalisieren von Linien- oder Polygongeometrien können Sie zwischen den Geometriezeichnungsmethoden hin- und herwechseln, so dass Sie Objekte erstellen können, die gerade Segmente, Freihandsegmente und gekrümmte Teile mischen.

4. Drücken Sie die Taste `Entfernen` oder `Löschen`, um den oder die letzten Knoten, die Sie fälschlicherweise hinzugefügt haben, rückgängig zu machen.
5. Wenn Sie alle Punkte digitalisiert haben, klicken Sie mit der rechten Maustaste auf eine beliebige Stelle, um die Digitalisierung der Geometrie dieses Objektes abzuschließen.

> **Tipp**
>
> **Das Digitalisierungsgummiband anpassen**
>
> Während des Erfassens von Polygonen kann das voreingestellte rote Band unterliegende Objekte oder Orte verstecken, um einen Punkt zu erfassen. Dies kann durch Einstellen einer geringeren Opazität (oder Alpha-Kanal) des Bands ausgebessert werden *Farbe füllen* im Menü *Einstellungen* ► *Optionen* ► *Digitalisierung*. Sie können auch die Verwendung des Gummibands vermeiden *Don't update rubber band during node editing*.

6. Wenn Sie ein Linienobjekt digitalisieren, schließt `Umschalt` + Rechtsklick die Linie automatisch.
7. Das Attributformular wir sich öffnen, in das Sie die Informationen für das neue Objekt eingeben können. Abb. 12.80 zeigt die Attributeingabe für einen neuen Fluss. Im Menü *Einstellungen* ► *Optionen* können Sie im Bereich *Digitalisieren* auch folgende Einstellungen vornehmen:
   - *Attributeingabeformular bei der Objekterstellung unterdrücken* um das Öffnen des Formulares zu vermeiden
   - oder *Letzte Attributwerteingaben wiederverwenden* um Felder beim Öffnen automatisch gefüllt zu haben und nur ändernde Werte zu korrigieren.

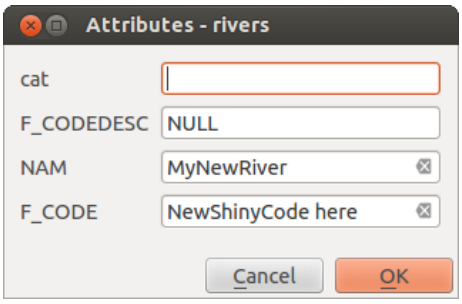


Abb. 12.80: Das Attributformular nach dem Digitalisieren eines neuen Vektorobjekts

Das Knotenwerkzeug

QGIS bietet Ihnen zwei Werkzeuge zum Bearbeiten der Stützpunkte von Objekten in Vektorlayern an:

- Knotenwerkzeug (Aktueller Layer): nur übereinanderliegende Objekte im aktiven Layer (im Bedienfeld *Layer*) werden bearbeitet
- Knotenwerkzeug (alle Layer): übereinanderliegende Objekte in allen editierbaren Layern werden bearbeitet. Damit können Sie Objekte bearbeiten ohne den aktiven Layer wechseln zu müssen oder mehrere Layer gleichzeitig (z.B. Grenzen eines Landes und seiner Regionen)

Die Knotenwerkzeuge bieten Ihnen für jeden editierbaren Vektorlayer Bearbeitungsmöglichkeiten für Stützpunkte, ähnlich wie sie in CAD-Programmen vorhanden sind. Sie können mehrere Stützpunkte auf einmal auswählen und sie verschieben, neue hinzufügen oder sie alle zusammen löschen. Die Knotenwerkzeuge unterstützen Sie auch beim topologischen Editieren. Ihre Auswahl bleibt bestehen, also wenn Sie eine Operation ausführen bleibt die Auswahl für das jeweilige Objekt und Werkzeug aktiv.

Tipp

Eine Reihe neuer Stützpunkte zeichnen

Das Stützpunkt-Werkzeug unterstützt keine *automatische Verfolgung* und ist für die Bearbeitung einzelner Stützpunktpunkte sowie das Verschieben oder Löschen einer beliebigen Auswahl mehrerer Stützpunktpunkte ausgelegt. Verwenden Sie das Werkzeug Objekte überarbeiten, wenn Sie eine Reihe neuer Stützpunktpunkte ersetzen oder einfügen müssen, insbesondere wenn Sie andere Features verfolgen möchten.

Es ist wichtig die Eigenschaften unter *Einstellungen* ► *Optionen* ► *Digitalisierung* ► *Suchradius:* auf eine Zahl größer als Null einzustellen. Andernfalls ist QGIS nicht in der Lage mitzuteilen welcher Stützpunkt bearbeitet werden soll und wird einen Warnhinweis zeigen.

Tipp

Stützpunktmarkierungen

QGIS unterstützt verschiedene Arten von Stützpunktmarkierungen: ‚Teiltransparenter Kreis', ‚Kreuz' und ‚Keine'. Um die Markierung zu ändern wählen Sie den Menüpunkt *Optionen* im Menü *Einstellungen* und klicken Sie auf den Tab *Digitalisieren*. Dort wählen Sie den passenden Eintrag.

Eine einfache Übung

Gegeben sei ein Layer im Bearbeitungsmodus. Aktivieren Sie das Knotenwerkzeug. Rote Kreise erscheinen, wenn Sie die Maus über Stützpunkte bewegen.

- **Stützpunkte auswählen**: Wählen Sie Stützpunkte auf eine der folgenden Weisen aus:
  - Klicken Sie sie einzeln an, während Sie `Umschalt` gedrückt halten
  - Klicken Sie und ziehen Sie ein Rechteck auf, das die gewünschten Stützpunkte enthält
  - Zeichnen Sie ein Polygon, das die gewünschten Stützpunkte enthält: Halten Sie die Taste `Alt` gedrückt und klicken Sie mit dem Knotenwerkzeug, um mit dem Digitalisieren des Polygons zu beginnen. Jeder folgende Klick fügt einen neuen Stützpunkt zum Polygon hinzu; es wird mit dem Gummiband dargestellt. Die Tasten `Zurück` oder `Entfernen` löschen den letzthinzugefügten Stützpunkt. Mit `Esc` brechen Sie die Auswahl über ein Polygon ab; das geht auch, indem Sie alle Stützpunkte löschen, die sie bisher erfasst haben. Mit Rechtsklick beenden Sie die Digitalisierung des Polygons und alle innerhalb gelegenen Stützpunkte sind ausgewählt.

Sobald ein Stützpunkt ausgewählt ist, wechselt seine Farbe auf blau. Fügen Sie weitere Stützpunkte zur aktuellen Auswahl hinzu, indem Sie `Umschalt` gedrückt halten, während Sie bei der Auswahl weiterhin so vorgehen, wie beschrieben. Halten Sie `Strg` gedrückt, um Stützpunkte aus der Auswahl zu entfernen.

> **Tipp**
>
> **Eine Objektauswahl beschränkt das Knotenwerkzeug**
>
> Sie können Stützpunkte über mehrere verschiedene Objekte (und Layer) auswählen. Interessieren Sie sich aber für die Stützpunkte eines bestimmten Objekts an einer Stelle mit vielen Objekten, sollten Sie zuerst dieses Objekt auswählen. Benutzen Sie danach die Stützpunktauswahl über ein Polygon oder ein Rechteck um die Sie interessierenden Stützpunkte: nun sind nur die Stützpunkte des vorher ausgewählten Objekts ausgewählt.
>
> Genauso verhält es sich, wenn Sie das Objekt im Bedienfeld *Stützpunktbearbeitung* anzeigen.

- **Stützpunkte im Stapelbetrieb auswählen**: Der Stapelbetriebsmodus wird durch drücken von `Umschalt+R` aktiviert. Wählen Sie einen ersten Stützpunkt durch einen einzelnen Klick aus und bewegen Sie den Mauszeiger **ohne zu klicken** über einen anderen Stützpunkt. Diese Aktion wählt alle Stützpunkte zwischen den beiden aus, wobei (für Polygone) der kürzeste Weg zwischen beiden benutzt wird.

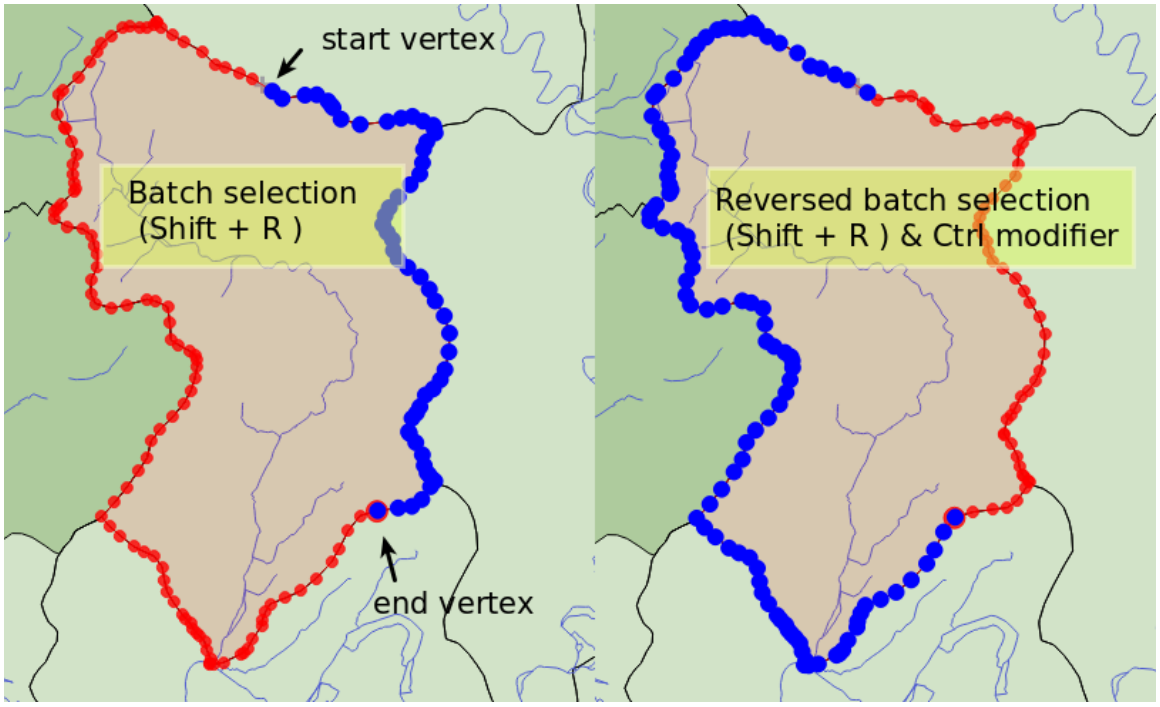


Abb. 12.81: Stützpunkte im Stapelbetrieb mit `Umschalt+R` auswählen

Wenn Sie `Strg` drücken, wird die Auswahl umgekehrt, so dass der längste Weg entlang der Objektgrenze benutzt wird. Die Stützpunktauswahl im Stapelbetrieb wird mit einem zweiten Klick abgeschlossen, oder mit `Esc` abgebrochen.

- **Stützpunkte hinzufügen**: Halten Sie `Umschalt` gedrückt und doppelklicken Sie auf die entsprechende Stelle des Segments, an dem Sie einer Linie oder einem Polygon einen neuen Stützpunkt hinzufügen wollen.

  Während Sie die Maus über ein Segment bewegen, erscheint ein virtueller Stützpunkt in der Mitte des Segments. Sie können darauf klicken, den Mauszeiger an einen anderen Ort bewegen und dort erneut klicken, um einen neuen Stützpunkt hinzuzufügen. Bei Linien erscheint der virtuelle Stützpunkt an beiden Endpunkten: klicken Sie darauf, klicken Sie weitere Male und beenden Sie die Eingabe mit einem Rechtsklick; mit dieser Methode können Sie einen vorhandenen Linienzug verlängern.

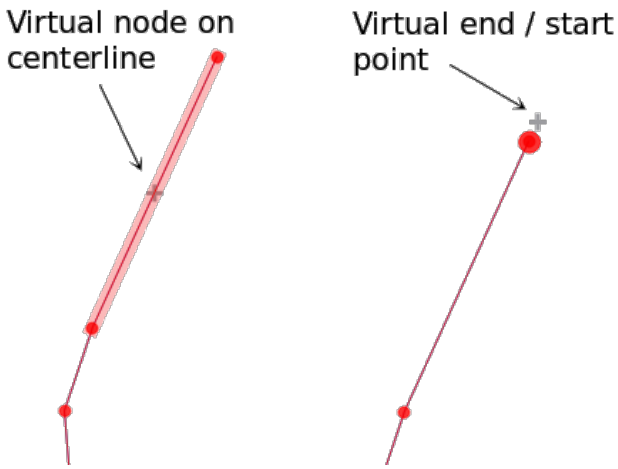


Abb. 12.82: Virtuelle Knoten um Stützpunkte hinzuzufügen

- **Stützpunkte löschen**: Wählen Sie die Stützpunkte und drücken Sie auf `Entfernen` . Das Löschen aller Stützpunkte erzeugt ein geometrieloses Objekt, sofern das mit der Datenquelle vereinbar ist. Beachten Sie, dass das nicht das gesamte Objekt löscht, sondern nur den Geometrieteil; um ein Objekt in Gänze zu löschen, müssen Sie das Werkzeug Ausgewähltes löschen benutzen.
- **Stützpunkte verschieben**: Wählen Sie zunächst alle Stützpunkte aus, die Sie verschieben wollen, klicken Sie auf einen davon und dann auf die gewünschte Zielposition. Benutzen Sie die *Einrastoptionen* und die Einschränkungen für Entfernung, Winkel sowie exakte X- und Y-Positionen im Bedienfeld *Erweiterte Digitalisierung* bevor Sie das zweite mal klicken. Alle ausgewählten Stützpunkte werden verschoben.

  Sollten Sie aber die die Option *auf Gitter einrasten* aktiviert haben, werden die ausgewählten Stützpunkte auf denjenigen Gitterpunkt eingerastet, der der neuen Position am nächsten ist. Auch nicht ausgewählte Stützpunkte werden zum nächstgelegenen Gitterpunkt verschoben. Es gibt kein einfaches Verschieben.

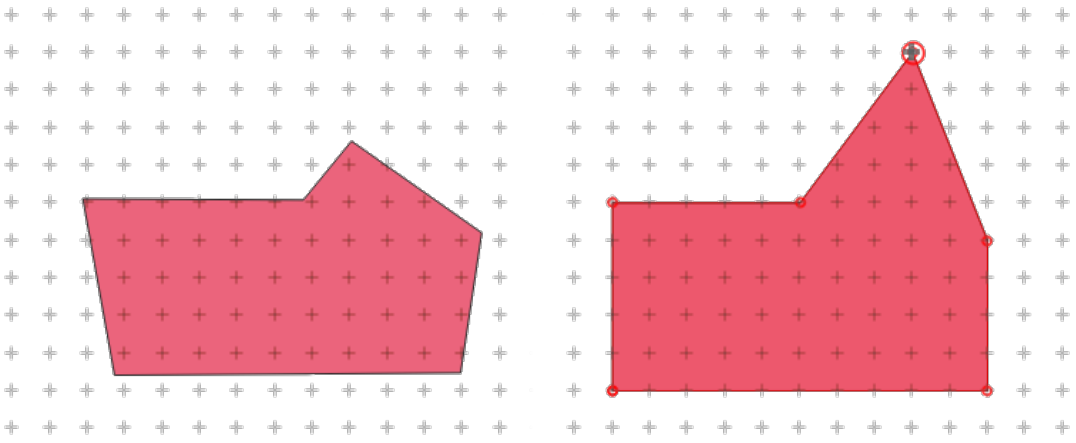


Abb. 12.83: Wird der oberste Stützpunkt verschoben, werden alle Stützpunkte auf das Gitter eingerastet

- **Nachbarsegmente in/aus Kurve umwandeln**: Wählen Sie den mittleren Scheitelpunkt des Segments, das Sie umwandeln möchten, und drücken Sie die Taste `O`. Wenn der Stützpunkt in einer Kurve lag, wird die Kurve in eine gerade Linie umgewandelt. Befand sich der Stützpunkt zwischen zwei Geraden, werden diese in eine Kurve umgewandelt. Der erste oder letzte Stützpunkt einer Linie kann nicht in eine Kurve mit Mittelscheitelpunkt umgewandelt werden. Der Layer muss mit dem Geometrietyp Kurve kompatibel sein.

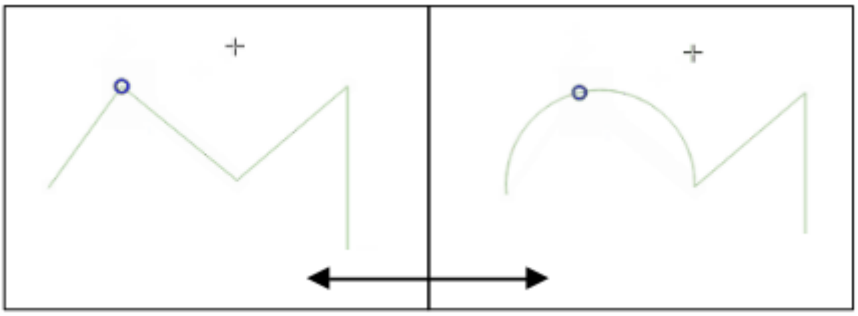


Abb. 12.84: Wechsel von Kurven zu Geraden mit dem Buchstabe ○

Jede Änderung, die Sie mit dem Knotenwerkzeug vornehmen wird als eigener Eintrag im Bedienfeld *Rücknahme* gespeichert. Denken Sie daran, dass alle Operationen sowohl topologisches Editieren als auch das Umprojezieren in Echtzeit unterstützen.

Das Bedienfeld Stützpunktbearbeitung

Wenn Sie ein Stützpunktwerkzeug aktivieren, öffnen Sie auch das Bedienfeld *Vertex Editor*. Wenn Sie mit der rechten Maustaste auf ein Objekt klicken, wird das Bedienfeld mit einer Liste aller Stützpunkte des Objekts mit ihren *x*, *y* (*z*, *m*, falls vorhanden) Koordinaten und *r* (für den Radius, im Falle einer kreisförmigen Geometrie) gefüllt. Das Objekt wird außerdem für die Bearbeitung exklusiv gemacht, was bedeutet, dass die Bearbeitung aller anderen Objekte deaktiviert ist:

- Wenn Sie eine Zeile in der Tabelle auswählen, wird der entsprechende Stützpunkt in der Kartenansicht ausgewählt und andersherum.
- Durch Klicken oder Ziehen über die Kartenansicht werden nur die Eckpunkte und Segmente dieses Objekts ausgewählt oder verschoben.
- Ändern Sie eine Koordinate in der Tabelle und die Position des Stützpunkts wird aktualisiert. Dies ist ein bequemer Weg, um die Z-Koordinate oder den M-Wert von Eckpunkten zu bearbeiten.
- Sie können auch mehrere Zeilen markieren und sie insgesamt löschen.
- Neue Stützpunkte können nur zu dem gebundenen Objekt hinzugefügt werden

Wenn Sie nicht möchten, dass das Bedienfeld *Vertex-Editor* bei jeder Interaktion mit den Scheitelpunktwerkzeugen sofort angezeigt wird (und möglicherweise andere Bedienfelder ausblendet oder die Platzierung von Bedienfeldern stört), deaktivieren Sie den Eintrag *Tabelle automatisch öffnen* im Menü ☰ Optionen am oberen Rand des Bedienfelds. Sie können das Bedienfeld dann auch schließen. Um das Bedienfeld wieder zu öffnen, müssen Sie mit der rechten Maustaste auf ein Bedienfeld oder eine Symbolleiste klicken und es/sie in der Liste auswählen oder den Eintrag *Vertex-Editor anzeigen* in der *Digitalisierungs-Symbolleiste* markieren.

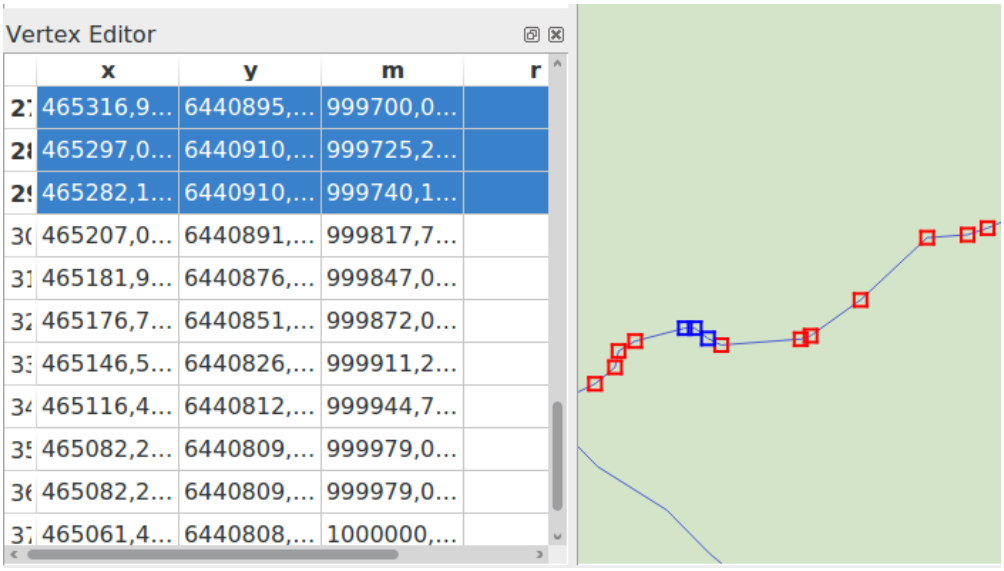


Abb. 12.85: Das Bedienfeld Stützpunktbearbeitung mit ausgewählten Knoten

Regeln für die Zuordnung von Z-Koordinaten oder M-Werten

Das Digitalisieren von 3D-Vektorobjekten oder Objekten mit M-Wert unterscheidet sich nicht wesentlich von (X,Y) 2D Layern. Die in diesem Kapitel beschriebenen Werkzeuge und Optionen sind immer noch verfügbar und helfen Ihnen, den Stützpunkt oder Punkt in einer ebenen Umgebung zu platzieren. Dann müssen Sie eventuell die Zuweisung der Z-Koordinate (oder des M-Werts) vornehmen:

- Standardmäßig weist QGIS neuen Scheitelpunkten den *Standard Z-Wert* (bzw. *Standard M-Wert*) zu, der in der Registerkarte *Einstellungen* ► *Optionen* ► *Digitalisieren* eingestellt ist. Wenn das *Erweiterte Digitalisierung Bedienfeld* verwendet wird, dann wird der Wert aus dessen *z* (bzw. *m*) Widget übernommen.
- Beim Fangen an einem Stützpunkt übernimmt der neue oder verschobene Stützpunkt den Z- oder M-Wert des gefangenen Stützpunkts.
- Wenn Sie bei eingeschalteter topologischer Bearbeitung an einem Segment einrasten, wird der neue Scheitelpunkt-Z- oder -M-Wert entlang des Segments interpoliert.
- Wenn das *z* (bzw. *m*) Widget des *Erweiterte Digitslisierung Bedienfeld* gesperrt ist, wird sein Wert auf den Stützpunkt angewandt und hat Vorrang vor jedem gefangenen Stützpunkt- oder Segment-Z- oder -M-Wert.

Um die Z- oder M-Werte eines bestehenden Objekts zu bearbeiten, können Sie das Bedienfeld *Vertex editor panel* verwenden. Um Objekte mit benutzerdefinierten Z- oder M-Werten zu erstellen, können Sie das *Erweitertes Bedienfeld für die Digitalisierung* verwenden.

Objekte ausschneiden, kopieren und einfügen

Ausgewählte Objekte können ausgeschnitten, kopiert und an andere Ebenen im aktuellen QGIS-Projekt übergeben (eingefügt) werden, wenn sich der Ziellayer auch im Editiermodus befindet, indem Sie nach Auswahl des Ziellayers auf den Knopf Bearbeitungsstatus umschalten klicken.

Tipp

Polygone in Linien und umgekehrt transformieren mit copy/paste

Kopieren Sie ein Linienobjekt und fügen Sie ihn in ein Polygonlayer ein: QGIS fügt ihn in den Ziellayer dessen Begrenzung am nächsten der Geometrie des Linienobjekts entspricht. Dies ist ein einfacher Weg, um verschiedene

Geometrien der selben Daten zu erzeugen.

Objekte können an externe Anwendungen als Text übergeben werden. Dabei werden die Objekte im CSV-Format dargestellt und die Geometriedaten im OGC Well-Know Text (WKT) Format. WKT und GeoJSON–Objekte von außerhalb können ebenso in einen Layer innerhalb von QGIS eingefügt werden.

Aber wann macht es Sinn, Objekte zum kopieren, auszuschneiden und einzufügen? Ein Beispiel ist, wenn Sie parallel an mehreren Layern arbeiten und Objekte zwischen den Layern hin- und herkopieren möchten. Ein Szenario könnte sein, dass Sie einen neuen Layer erstellen möchten, in dem aber nur einige Objekte aus einem bereits existierenden Layer verwendet werden sollen, wie etwa 5 Seen aus der Karte `lakes.shp` , die insgesamt aber tausende Seen enthält.

Als Beispiel werden wir einige Seen in einen neuen Layer kopieren:

1. Laden Sie den Layer, von dem Sie einige Objekte kopieren wollen (Quelle)
2. Laden oder erstellen Sie einen Layer, in den die kopierten Objekte eingefügt werden sollen (Ziel)
3. Schalten Sie für den Ziel Layer den Bearbeitungsstatus ein
4. Stellen Sie die Quelle aktiv, indem Sie es in der Legende anklicken
5. Objekte über Fläche oder Einzelklick wählen um Objekte aus dessen Quelllayer zu wählen
6. Klicken Sie auf das Icon Objekte kopieren
7. Stellen Sie das ‚Ziel' aktiv, indem Sie es in der Legende anklicken
8. Klicken Sie auf das Icon Objekte einfügen
9. Beenden Sie den Bearbeitungsstatus für beide Layer und speichern Sie das Ergebnis ab

Was passiert, wenn der Quell- und Ziellayer ein unterschiedliches Schema enthält (Spaltennamen und -typen unterscheiden sich)? QGIS verwendet die Einträge, die gleich sind und ignoriert den Rest. Wenn es Ihnen egal ist, ob die Attribute korrekt übernommen werden, dann ist es egal, wie Sie die Spaltennamen und -typen der Attributtabelle erstellen. Wenn auch die Attributdaten korrekt übernommen werden sollen, dann stellen Sie sicher, dass auch die Spaltennamen und -typen beider Layer zueinander passen.

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.

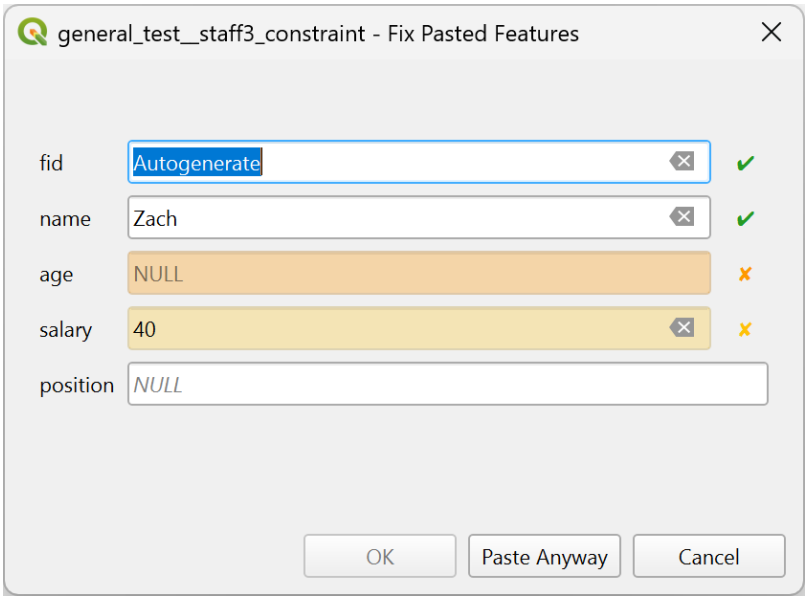


Abb. 12.86: Dialog for handling invalid attribute values when pasting a single feature

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.

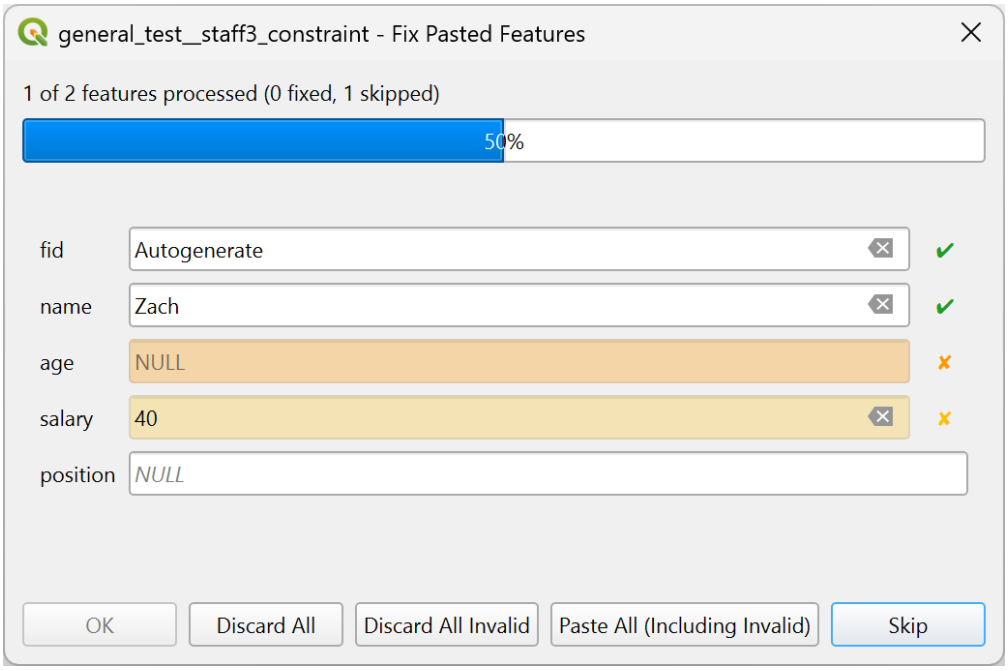


Abb. 12.87: Dialog for handling invalid attribute values when pasting multiple features

Bemerkung

Deckungsgleichheit eingefügter Objekte

Wenn Ihre Quell- und Ziellayer die gleiche Projektion verwenden dann haben die eingefügten Objekte die identische Geometrie wie der Quellayer. Wenn der Ziellayer jedoch eine andere Projektion hat dann kann QGIS nicht garantieren dass die Geometrie identisch ist. Dies ist einfach aus dem Grund so, dass sich kleine Rundungsfehler ergeben wenn zwischen Projektionen konvertiert wird.

Tipp

Zeichenketten Attribut in ein anderes kopieren

Wenn Sie in Ihrer Attributtabelle eine neue Spalte mit dem Typ „String“ erstellt haben und Werte aus einer anderen Attributspalte einfügen möchten, die eine größere Länge hat, wird die Länge der Spalte um den gleichen Betrag erweitert. Dies liegt daran, dass der GDAL Shapefile-Treiber in der Lage ist, String- und Integer-Felder automatisch zu erweitern, um die Länge der einzufügenden Daten dynamisch zu berücksichtigen.

Ausgewählte Objekte löschen

Wenn Sie ein gesamtes Objekt (Attribute und Geometrie) löschen wollen, können Sie das tun, indem Sie zunächst die Geometrie der Objekte über Fläche oder Einzelklick wählen auswählen. Die Auswahl kann auch über die Attributtabelle erfolgen. Sobald Sie die Auswahl festgelegt haben, drücken Sie `Entfernen` oder `Rücktaste` oder nutzen Sie Ausgewähltes löschen um die Objekte zu löschen. Mehrfachauswahlen können ebenso auf einmal gelöscht werden.

Das Werkzeug Ausgewählte Objekte ausschneiden kann auch benutzt werden, um Objekte zu löschen. Die Objekte werden gelöscht aber zusätzlich noch im ‚spatial clipboard‘ abgelegt. In diesem Fall könnte man dann den letzten Schritt, falls ein Fehler unterlaufen ist, wieder rückgängig machen, indem wir auf das Werkzeug Objekte einfügen drücken. Ausschneiden, kopieren und übergeben von Objekten funktioniert mit den gerade ausgewählten Objekten und können nach Bedarf kombiniert verwendet werden.

Rückgängig und Wiederholen

Mit dem Rückgängig und Wiederholen Werkzeug können Sie Vektorbearbeitungsoperationen rückgängig machen oder wiederholen. Es gibt auch ein andockbares Bedienfeld das alle Operationen im Rücknahme-/Wiederholungsverlauf zeigt (siehe *Figure_edit_undo*). Dieses Bedienfeld wird nicht standardmäßig angezeigt; es kann angezeigt werden, indem Sie mit rechts auf die Werkzeugleiste klicken und das Rücknahme/Wiederholungs-Kontrollkästchen aktivieren. Rücknahme/Wiederholung ist auch dann aktiv wenn das Bedienfeld nicht angezeigt wird.

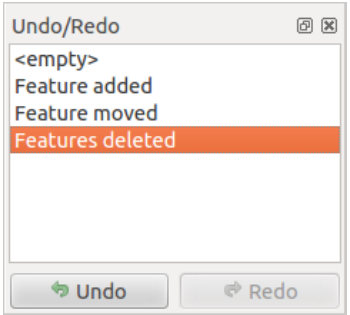


Abb. 12.88: Rückgängig und Wiederholen von Digitalisierschritten

Wenn Rückgängig oder `Strg+z` (oder `Cmd+z`) gedrückt wird, wird der Status aller Objekte und Attribute auf den Status zurückgesetzt, den sie vor der zurückgenommenen Aktion hatten. Es kann sein, dass Änderungen die keine normalen Vektorbearbeitungsoperationen sind (z.B. Änderungen, die durch ein Plugin durchgeführt wurden) nicht rückgängig gemacht werden können. Dies hängt davon ab, wie die Änderungen durchgeführt werden.

Wenn Sie Rücknahme/Wiederholung verwenden wollen klicken Sie einfach auf eine Operation in der History; alle Objekte werden dann auf den Stand vor der ausgewählten Operation zurückgesetzt.

Änderungen speichern

Wenn ein Layer im Bearbeitungsmodus ist behält QGIS alle Änderungen im Speicher. Aus diesem Grund werden diese nicht umgehend der Datenquelle oder -platte übermittelt. Wenn Sie Bearbeitungen in dem aktuellen Layer speichern wollen aber mit dem Bearbeiten fortfahren wollen ohne den Bearbeitungsmodus verlassen zu wollen können Sie den Layeränderungen speichern Knopf klicken. Wenn Sie den Editiermodus mit Bearbeitungsstatus umschalten aussschalten wollen werden Sie ebenfalls gefragrt ob Sie Ihre Änderungen speichern oder verwerfen wollen.

Wenn die Änderungen nicht gespeichert werden können (z.B. weil die Festplatte voll ist oder Attribute Werte aufweisen, die außerhalb der Wertespanne liegen), bleiben die Änderungen erstmal im QGIS Arbeitsspeicher. Dies ermöglicht es, Änderungen vorzunehmen und dann nochmals die Daten zu speichern.

Tipp

Datenintegrität

Es ist immer gut ein Backup von Ihren Daten zu machen bevor Sie mit dem Bearbeiten starten. Während die Autoren von QGIS sich bemühen die Integrität Ihrer Daten zu bewahren bieten wir keine Garantie in dieser Hinsicht.

Mehrere Layer auf einmal speichern

Mit dieser neuen Funktion können Sie mehrere Layer digitalisieren. Wählen Sie *Layeränderungen speichern* um alle Änderungen, die Sie in mehreren Layern gemacht haben, zu speichern. Sie haben auch die Möglichkeit *Verwerfen für gewählte Layer* zu benutzen so dass die Digitalisierung für alle selektierten Layer rückgängig gemacht werden kann. Wenn Sie das Bearbeiten der selektierten Layer beenden wollen kann man das einfach mit der Funktion *Abbruch für gewählte Layer* erreichen.

Die gleichen Funktionen sind für das Bearbeiten aller Layer des Projektes zugängig.

Tipp

Transaktionsgruppen benutzen, um Änderungen in mehreren Layern zu bearbeiten, zu speichern oder zurückzunehmen

Aktivieren Sie Option *Wenn möglich automatisch Transaktionsgruppen erstellen* im Menü *Projekt* ► *Einstellungen…* ► *Datenquellen* beim Arbeiten mit Layern aus der selben PostGreSQL-Datenbank, um ihr Verhalten zu synchronisieren (den Bearbeitungsstatus zusammen umschalten, Änderungen gleichzeitig speichern oder verwerfen).

12.3.5 Erweiterte Digitalisierung

Tab. 12.4: Werkzeuge in der erweiterte-Digitalisierung-Werkzeugleiste für Vektorlayer

Icon	Zweck	Icon	Zweck
	Erweiterte Digitalsierungswerkzeuge einschalten		
	Objekt(e) verschieben		Objekt(e) kopieren und verschieben
	Objekt(e) drehen		Objekt vereinfachen
	Objekt skalieren		
	Ring hinzufügen		Teil hinzufügen
	Ring füllen		Linie umkehren
	Ring löschen		Teil löschen
	Linie versetzen		Objekte überarbeiten
	Teile zerlegen		Objekte trennen
	Attribute gewählter Objekte vereinen		Gewählte Objekte verschmelzen
	Punktsymbole drehen		Punktsymbolversatz
	Objekt abschneiden/ausdehnen		

Objekt(e) verschieben

Mit dem Objekt(e) verschieben Werkzeug können Sie bestehende Objekte verschieben.

1. Wählen Sie die Objekte aus, die Sie verschieben möchten.
2. Klicken Sie in das Kartenfenster, um den Quellpunkt der Verschiebung zu markieren; Sie können das Einrasten benutzen, um einen exakten Punkt auszuwählen.

   Sie können auch die Einstellungen im Bedienfeld *advanced digitizing constraints* benutzen, um die exakten Koordinaten des Quellpunktes zu setzen. In diesem Falle:

   1. Klicken Sie zunächst auf die Schaltfläche , um das Bedeinfeld zu aktivieren.
   2. Tippen Sie `x` und tragen Sie den entsprechenden Wert für den Quellpunkt ein, den Sie benutzen wollen.

      Drücken Sie dann die Schaltfläche neben der Eingabezeile, um den Wert zu sperren.
   3. Genauso für die `y`-Koordinate.
   4. Klicken Sie ins Kartenfenster und Ihr Quellpunkt wird an den eingegebenen Koordinaten platziert.
3. Bewegen Sie den Mauszeiger über das Kartenfenster, um den Zielpunkt der Verschiebung zu markieren, wobei Sie weiterhin den Einrastmodus benutzen können oder, wie oben beschrieben, bennutzen Sie das Bedeinfeld Erweiterte Digitalisierung, das Ihnen zusätzliche `Abstand`- und ``Winkel``einschränkungen anbietet, um den Zielpunkt der Verschiebung zu platzieren.
4. Klicken Sie ins Kartenfenster: Alle Objekte werden an ihren neuen Ort verschoben.

Mit dem Werkzeug Objekt(e) kopieren und verschieben können Sie gleichermaßen eine verschobene Kopie der Objekte erzeugen.

Bemerkung

Wenn kein Objekt ausgewählt ist, während Sie das erste mal mit einem der Werkzeuge *Objekt(e) verschieben* oder *Objekt(e) kopieren und verschieben* ins Kartenfenster klicken, wird nur das Objekt unter dem Mauszeiger von der Aktion betroffen. Sollten Sie also mehrere Objekte verschieben wollen, sollten sie vorher ausgewählt worden sein.

Objekt(e) drehen

Mit dem Werkzeug Objekt drehen können Sie ein oder mehrere Objekte im Kartenfenster drehen.

1. Drücken Sie auf die Schaltfläche Objekt drehen
2. Klicken Sie auf das Objekt, das sie drehen möchten. Der Flächenschwerpunkt des Objekts wird als Drehmittelpunkt markiert und ein Eingabefeld erscheint, das den aktuellen :guilabel:`Rotation`swinkel anzeigt.
3. Klicken Sie ins Kartenfenster, wenn Sie mit der neuen Position zufrieden sind oder geben Sie den Drehwinkel händisch in das Eingabefeld ein. Sie können auch die Eingabe *Einrasten auf* ° benutzen, um den Winkel einzuschränken.
4. Wollen Sie mehrere Objekte auf einmal drehen, sollten Sie sie zuerst auswählen. Der Drehmittelpunkt ist dann der Schwerpunkt ihrer verbundenen Geometrien.

Sie können auch einen anderen Drehmittelpunkt als den Flächenschwerpunkt benutzen: klicken Sie ins Kartenfenster während Sie die Taste `Strg` drücken und der angeklickte Punkt wird der neue Drehmittelpunkt.

Wenn Sie `Umschalt` vor dem Klick auf die Karte halten, wird die Drehung in 45 Grad Schritten erfolgen, was anschließend in einem Eingabefeld bearbeitet werden kann.

Zum Abbrechen der Objektdrehung, drücken Sie die Taste `Esc` oder klicken Sie auf die Schaltfläche Objekt(e) drehen.

Objekt skalieren

Das Werkzeug Objekt skalieren ist ähnlich wie das Werkzeug Objekt drehen. Aber anstatt das ausgewählte Objekt zu drehen skaliert es dessen Geometrie. Die Änderung wird in Abhängigkeit vom Ankerpunkt ausgeführt und den Skalierungsfaktor können Sie manuell in das Fenster eingeben, das in der oberen Ecke der Karte erscheint.

Objekt vereinfachen

Mit dem Werkzeug Objekt vereinfachen können Sie die Form einer Linie oder eines Polygons interaktiv verändern, indem Sie die Anzahl der Stützpunkte verringern oder verdichten, so lange die Geometrie dabei gültig bleibt:

1. Wählen Sie das Werkzeug Objekt vereinfachen.
2. Klicken Sie auf das Objekt oder ziehen Sie ein Rechteck über die Objekte auf.
3. Ein Eingabefeld öffnet sich, in dem Sie die zu verwendende *Methode* festlegen können, d.h. ob Sie:
   - die *simplify the geometry* wollen, so dass sie weniger Stützpunkte als die Ausgangsgeometrie enthält. Die zur Verfügung stehenden Methoden sind `Nach Abstand vereinfachen`, `Durch Einrasten auf Gitter vereinfachen` oder `Durch Fläche vereinfachen (Visvalingam)`. Dann müssten Sie noch die *Toleranz* in `Layereinheiten`, `Pixel` oder `Karteneinheiten` angeben, die für die Vereinfachung benutzt werden soll. Je höher die Toleranz, desto mehr Stützpunkte werden entfernt.

- oder die *densify the geometries* wollen, indem mit der Option `Glatt` neue Stützpunkte eingefügt werden: für jeden bestehenden Stützpunkt werden zwei Stützpunkte auf jedes von ihm ausgehende Segment gesetzt, und zwar in der mit *Versatz* definierten Entfernung, wobei der Versatz dem Prozentanteil der Segmentlänge entspricht. Mit *Wiederholungen* legen Sie fest, wie oft die Platzierung wiederholt werden soll: je mehr Wiederholungen Sie einstellen, desto mehr Stützpunkte werden eingefügt und desto regelmäßiger wird das Objekt.

Wenn Sie ein Projekt schließen oder eine Editiersitzung beenden, werden Einstellungen, die Sie benutzt haben, gespeichert. Sie können also die selben Parameter wieder benutzen, wenn Sie das nächste mal ein Objekt vereinfachen.

4. Unten im Eingabefenster wird eine Zusammenfassung der Veränderungen angezeigt, die sich ergäben; dabei wird die Anzahl der Objekte und Stützpunkte aufgelistet (vor und nach der Operation sowie das Verhältnnis daraus). Zusätzlich wird im Kartenfenster die zu erwartende Geometrie über die bestehende mithilfe der Gummibandfarbe dargestellt.
5. Sobald die zu erwartende Geometrie Ihren Vorstellungen entspricht, klicken Sie *OK*, um die Änderung anzuwenden. Sollten Sie die Operation abbrechen wollen, klicken Sie auf *Abbrechen* oder klicken Sie mit rechts in das Kartenfenster.

Bemerkung

im Gegensatz zur Objektvereinfachung im Menü *Einstellungen* ► *Optionen* ► *Darstellung*, welches die Geometrie nur zur Darstellung vereinfacht, bearbeitet das Werkzeug Objekt vereinfachen die Geometrie in der Datenquelle.

Teil hinzufügen

Sie können zu einem ausgewählten Objekt einen Teil hinzufügen, das erzeugt ein Multipunkt-, Multilinie- oder Multipolygonobjekt. Der neue Teil muss außerhalb des bereits vorhandenen digitalisiert werden, der vorher ausgewählt werden sollte.

Die Schaltfläche Teil hinzufügen kann ebenfalls benutzt werden, um eine Geometrie zu einem geometrielosen Objekt hinzuzufügen. Zuerst wählen Sie das Objekt in der Attributtabelle, dann digitalisieren Sie eine neue Geometrie mit dem Werkzeug Teil hinzufügen.

Bemerkung

Reihenfolge der Stützpunkte in Polygonteilen

Im Gegensatz zu den OGC-Standards erzwingt QGIS nicht, dass die Stützpunkte der Außenbegrenzung eines Polygon-Objekts gegen den Uhrzeigersinn angeordnet sein müssen. Daher können innerhalb eines Layers beide Richtungen vorkommen. Allerdings sind die äußeren Stützpunkte aller Teile desselben Multipolygon-Objekts in derselben Richtung angeordnet.

Sie können jedoch den Algorithmus *Force right-hand-rule* verwenden, um die Objekte eines Layers so zu beschränken, dass die Stützpunkte der Außengrenzen im Uhrzeigersinn angeordnet sind.

Teil löschen

Mit dem Werkzeug Teil löschen können Teile eines Multiobjekts gelöscht werden (z.B. ein Polygon eines Multipolygonobjekts). Das Werkzeug funktioniert mit allen Multigeometrien: Punkt, Linie und Polygon. Weiterhin kann es dazu verwendet werden, die Geometrie eines Objekts vollständig zu entfernen. Um einen Teil zu löschen, klicken Sie einfach in dieses Teil.

Ring hinzufügen

Sie können Löcher in Polygonen mit dem Werkzeug Ring hinzufügen herstellen. Das bedeutet, dass Sie innerhalb eines bestehenden Polygons weitere Polygone digitalisieren. Diese erscheinen dann als ‚Loch', so dass nur der Zwischenraum des inneren und des äußeren Polygons als Ringpolygon übrig bleibt.

Um einen Ring hinzuzufügen, müssen Sie wie folgt vorgehen:

1. Wählen Sie das zu ändernden Objekte aus.
2. Aktivieren Sie das Ring hinzufügen-Werkzeug.
3. Zeichnen Sie ein Polygon innerhalb der ausgewählten Geometrie unter Verwendung der oben genannten *Techniken*. In den ausgewählten Geometrien erscheint ein Loch.
4. Wenn beim Zeichnen des Rings keine Geometrie ausgewählt ist, wird jedem Polygon, über das der Ring gezeichnet wird, ein Loch hinzugefügt.

Bemerkung

Reihenfolge der Stützpunkte in Polygon-Ringen

Im Gegensatz zu den OGC-Standards erzwingt QGIS nicht, dass die Stützpunkte der Außenbegrenzung eines Polygon-Objekts gegen den Uhrzeigersinn angeordnet sein müssen. Daher können innerhalb eines Layers beide Richtungen vorkommen. Allerdings sind die äußeren Stützpunkte aller Ringe desselben (Multi-)Polygon-Objekts in derselben Richtung angeordnet.

Sie können jedoch den Algorithmus *Force right-hand-rule* verwenden, um die Objekte eines Layers so zu beschränken, dass die Stützpunkte der Außenkonturen im Uhrzeigersinn und die Stützpunkte der Innenringe gegen den Uhrzeigersinn angeordnet sind.

Ring füllen

Das Werkzeug Ring füllen hilft Ihnen dabei, ein Polygonobjekt zu erzeugen, das komplett innerhalb eines anderen Polygons liegt und so gestaltet ist dass sich beide nirgendwo überlappen; das neue Objekt liegt innerhalb eines Lochs im bestehenden Objekt. So gehen Sie vor:

1. Wählen Sie das Werkzeug Ring füllen.
2. Zeichnen Sie ein Polygon über das vorhandene Objekt: QGIS fügt der Geometrie ein neus Loch hinzu (wie wenn Sie das Werkzeug Ring hinzufügen benutzt hätten) und erzeugt ein neues Objekt, dessen Geometrie dem Loch entspricht (wie wenn Sie *Spurverfolgung* über die innere Grenze mit dem Werkzeug Polygonobjekt hinzufügen angewendet hätten).
3. Falls das Loch im Objekt bereits exisitert, können Sie alternativ den Mauszeiger darüber positionieren und mit links klicken, während Sie die :kbd:`Umschalt`taste gedrückt halten: ein neues Objekt, das das Loch ausfüllt, wird an dieser Stelle gezeichnet.

Das Formular *Objektattribute* des neuen Objekts öffnet sich und ist mit Werten des „Eltern“objekts und/oder *fields constraints* vorausgefüllt.

Ring löschen

Das Werkzeug Ring löschen ermöglicht es, Löcher innerhalb einer existierenden Fläche zu löschen. Das Werkzeug funktioniert nur mit Polygon- und Multipolygonobjekten. Es findet keine Veränderung statt, wenn es auf den äußeren Ring eines Polygons angewendet wird.

Objekte überarbeiten

Mit dem Werkzeug Objekte überarbeiten können Sie Linien- und Polygonobjekte umformen. Bei Linien ersetzt es den Teil der Linie vom ersten bis zum letzten Schnittpunkt mit der ursprünglichen Linie.

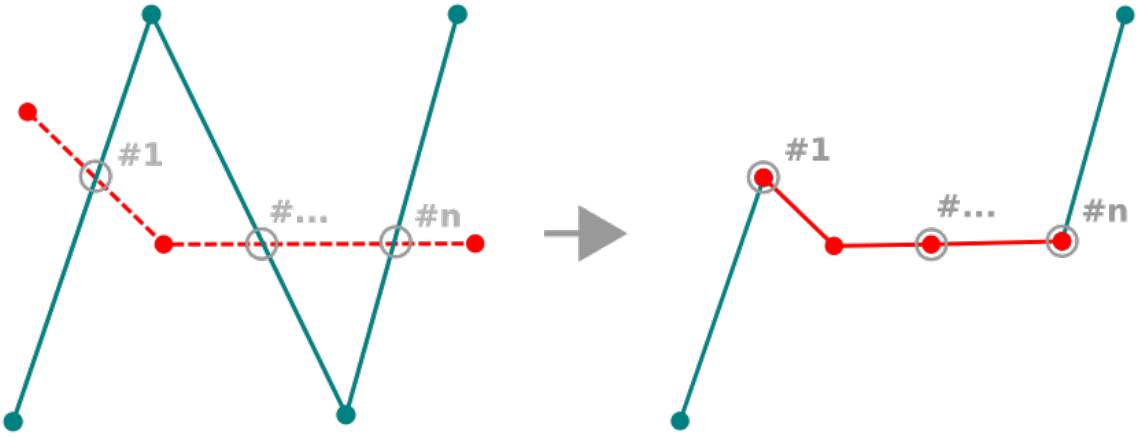


Abb. 12.89: Eine Linie umformen

Tipp

Liniengeometrien mit dem Werkzeug Objekte überarbeiten verlängern

Benutzen Sie das Werkzeug Objekte überarbeiten, um vorhandene Liniengeometrien zu verlängern: Fangen Sie den ersten oder letzten Stützpunkt der Linie und zeichnen sie eine neue. Schließen Sie die Digitalisierung ab und die neue Objektgeometrie ist die vereinigte Geometrie der beiden Linien.

Bei Polygonen wird die Polygongrenze umgeformt. Damit das funktioniert, muss die Linie des Umformwerkzeugs die Polygongrenze mindestens zwei mal kreuzen. Um diese Linie zu zeichnen, klicken Sie in das Kartenfenster, um ihre Stützpunkte zu erzeugen. Die Zeichnung schließen Sie mit einem Rechtsklick ab. Genauso wie bei Linienobjekten wird nur das Segment zwischen dem ersten und dem letzten Schnittpunkt der Linie mit der Polygongrenze betrachtet. Diejenigen Teile der Linie, die innerhalb des Polygons liegen, schneiden es dort ab, diejenigen, die außerhalb liegen erweitern es.

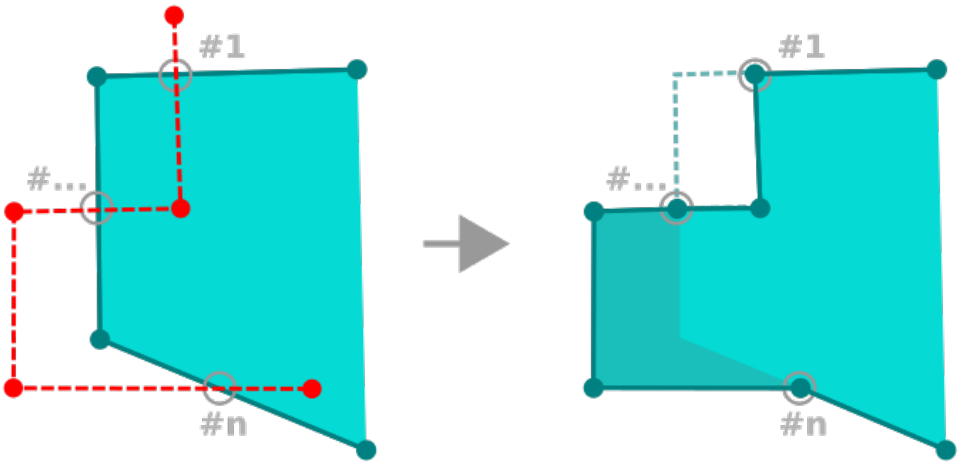


Abb. 12.90: Ein Polygon umformen

Bei Polygonen kann das Umformen manchmal zu unerwarteten Ergebnissen führen. Es ist in erster Linie dafür geeignet, kleinere Teile eines Polygons zu ersetzen und nicht für größere Überarbeitungen, außerdem ist es nicht erlaubt, dass die Umformline mehrere Ringe eines Polygons kreuzt, weil das zu einem ungültigen Polygon führen würde.

Bemerkung

Das Objekte überarbeiten Werkzeug kann die Startposition eines Polygonringes oder einer geschlossenen Linie verändern. Der Punkt, der zweimal abgebildet ist wird also nicht mehr der gleiche sein. Dies mag kein Problem für die meisten Anwendungen sein, sollte aber beachtet werden.

Linie versetzen

Das Linie versetzen Werkzeug erzeugt parallele Verschiebungen von Linien-Objekten. Das Werkzeug kann auf den bearbeiteten Layer (die Geometrien werden geändert) oder auch auf Hintergrundlayer angewendet werden (in diesem Fall werden Kopien der Linien/Ringe erstellt und dem bearbeiteten Layer hinzugefügt). Es eignet sich daher ideal für die Erstellung von Abstandslinienlayern. Ein *Benutzereingabe*-Dialogfeld wird angezeigt, in welchem der Versatzabstand und andere Einstellungen vorgenommen werden können.

Um einen Linienversatz eines Linien- oder Polygon-Objekts zu erstellen, müssen Sie zunächst in den Bearbeitungsmodus wechseln und das Werkzeug Linie versetzen aktivieren. Klicken Sie dann auf ein Objekt, um die Linie zu verschieben. Bewegen Sie die Maus und klicken Sie an die gewünschte Stelle oder geben Sie den gewünschten Abstand in das Benutzereingabefeld ein. Wenn Sie beim zweiten Klick die Taste `Strg` gedrückt halten, wird eine versetzte Kopie erstellt.

Wenn Sie von Geometrien aus Hintergrundlayern eine Kopie mit einem Versatz erstellen wollen, stellen Sie sicher, dass das Einrasten aktiviert ist, und halten Sie die Taste `Strg` gedrückt, um die Geometrie aus dem Hintergrundobjekt auszuwählen. Halten Sie die Taste `Strg` auch beim zweiten Klick gedrückt. Die Geometrien werden in den Geometrietyp des Ziellayers konvertiert.

Unter Einstellungen > Optionen (Registerkarte „Digitalisierung“, Abschnitt **Werkzeug zum Versetzen von Linien**) oder über das Einstellungen-Icon während der Bearbeitung können sie *einige Parameter* wie **Verbindungsstiel**, **Quadrantensegmente**, **Eckgrenze** und **Endstil** einstellen.

Linie umkehren

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.

So ändern Sie die Digitalisierrichtung einer Linie:

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.

Tipp

For bulk line direction changes, use the *Reverse line direction* processing algorithm which can reverse all lines in a layer or a selection at once.

Objekte trennen

Mit dem Werkzeug Objekte zerteilen teilen Sie ein Objekt in zwei oder mehr unabhängige Objekte, d.h. jede Geometrie gehört zu einer neuen Zeile in der Attributtabelle.

So teilen Sie Linien oder Polygonobjekte:

1. Wählen Sie das Werkzeug Objekte zerteilen.
2. Zeichnen Sie eine Linie über die Objekte, die Sie teilen möchten. Wenn eine Auswahl aktiv ist, werden nur ausgewählte Objekte geteilt. Dem ursprünglichen Objekt wird dann die größte Geometrie zugewiesen, die sich aus der Teilung ergibt, und für die verbleibenden Teile werden neue Objekte erstellt. Die Felder der Objekte werden gemäß den Regeln in der Datenquelle oder den *Teilungs-Richtlinie* ausgefüllt/aktualisiert.
3. Sie können dann wie gewohnt alle Attribute aller resultierenden Objekte ändern.

Tipp

Eine Polylinie mit einem Klick in neue Objekte zerteilen

Fangen und Klicken Sie mit dem Werkzeug Objekte zerteilen auf einen vorhandenen Stützpunkt einer Polylinie, um sie in zwei neue Objekte zu zerteilen.

Teile zerlegen

In QGIS ist es möglich die Teile eines Multi-Part Features zu zerlegen so dass die Anzahl der Teile sich erhöht. Zeichnen Sie einfach eine Linie über den Teil den Sie zerlegen wollen indem Sie dafür das Teile zerlegen Icon verwenden.

> **Tipp**
>
> **Eine Polylinie mit einem Klick in neue Teile zerlegen**
>
> Fangen und Klicken Sie mit dem Werkzeug Teile zerlegen auf einen vorhandenen Stützpunkt einer Polylinie, um sie in zwei neue Polylinien zu zelegen, die weiterhin zum selben Objekt gehören.

Gewählte Objekte verschmelzen

Mit dem Werkzeug Gewählte Objekte verschmelzen erzeugen Sie ein neues Objekt durch das verschmelzen vorhandener Objekte: Ihre Geometrien werden verschmolzen, um ein neues zu bilden. Wenn die Objekte keine gemeinsame Grenze haben, wird ein Multipolygon-/Multipolylinien-/Multipunktobjekt erzeugt.

1. Wählen Sie zunächst die Objekte aus, die Sie verschmelzen wollen.
2. Drücken Sie dann auf die Schaltfläche Gewählte Objekte verschmelzen.
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:
   - ändern des Wertes in der entsprechenden Zelle;
   - eine Zeile in der Tabelle auswählen und auf die Schaltfläche *Attribute des gewählten Objekts übernehmen* klicken. Damit übernehmen Sie alle Werte dieses Objekts;
   - klicken Sie auf *Attribute des Objekts mit der größten Fläche verwenden*, um die Attribute des längsten Linienobjekts, des größten Polygonobjekts oder des Multipointobjekts mit den meisten Teilen zu übernhmen;
   - auf die Schaltfläche *Alle Felder überspringen* klicken, um leere Attribute zu benutzen;
   - klicken Sie auf das kleine Dreieck im Kopf der Tabelle und wählen Sie eine der Möglichkeiten aus, die dann ausschließlich auf das jeweilige Feld angewendet wird. Sie können auch die Werte der Ausgangsobjekte aggregieren (Minimum, Maximum, Median, Summe, Anzahl, Verknüpfung…, jeweils abhängig vom Feldtyp. Sehen Sie im Abschnitt *Das Statistik-Bedienfeld* für eine Liste aller Funktionen nach).

   > **Bemerkung**
   >
   > Falls der Layer Standardwerte oder -klauseln für Felder definiert hat, werden diese als Startwert für das verschmolzene Objekt benutzt.

4. Drücken Sie auf *OK* um die Änderungen anzuwenden. Ein einzelnes (Multi-)Objekt wird im Layer angelegt und ersetzt die vorher ausgewählten Objekte.

Attribute gewählter Objekte vereinen

Mit dem Werkzeug Attribute gewählter Objekte vereinen können Sie die selben Attribute in Objekte speichern, ohne ihre Grenzen zu vereinen. Die Eingabemaske ist die selbe wie beim Werkzeug `Gewählte Objekte verschmelzen`, außer dass die ausgewählten Objekte mit ihrer Geometrie bestehen bleiben, während einige ihrer Attributwerte identisch werden.

Punktsymbole drehen

Das Werkzeug Punktsymbole drehen ermöglicht es Ihnen, den Drehwinkel eines Punktsymbols im Kartenfenster zu ändern.

1. Zuerst müssen Sie das Feld angeben, in dem der Rotationswert gespeichert werden soll. Dies geschieht, indem dem Symbol data-defined rotation property: ein Feld zugewiesen wir:
   1. Im Dialog *Layereigenschaften* ► *Symbolisierung* gehen Sie zu den Einstellungen für das Symbol.
   2. Klicken Sie auf die Schaltfläche *Datendefinierte Übersteuerung* neben der Einstellung *Drehung* (bevorzugt) in der obersten Ebene *Markierung* der Symbolebenen.
   3. Wählen Sie ein Feld Auswahl *Feldtype*. Die Werte in diesem Feld werden nun dazu benutzt, jedes Symbol eines Objekts entsprechend zu drehen.

      Sie können auch Daten im Projekt speichern anhaken, um ein Feld im *Hilfsspeicher* zu erzeugen, das die Drehungswerte aufnimmt.

> **Bemerkung**
>
> **Stellen Sie sicher, dass in allen Symbolebenen das selbe Feld eingestellt ist**
>
> Wenn Sie das datendefinierte Feld für die Drehung in der obersten Ebene der Symbolebenen einstellen, ist es automatisch in allen Symbolebenen eingestellt. Das ist die Voraussetzung für die graphische Symboldrehung mit dem Werkzeug *Punktsymbole drehen*. Falls auch nur eine Symbolebene nicht das selbe Feld für die Drehung eingestellt hat, wird das Werkzeug nicht funktionieren.

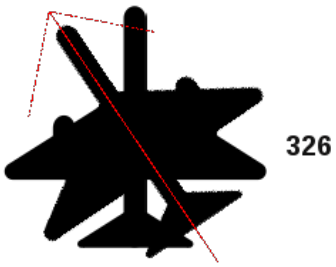


Abb. 12.91: Ein Punktsymbol drehen

2. Klicken Sie danach mit dem Werkzeug Punktsymbole drehen auf ein Punktsymbol in der Karte
3. Bewegen Sie die Maus. Ein roter Pfeil mit dem Drehwinkel wird Ihnen angezeigt (siehe Abb. 12.91). Wenn Sie die Taste `Strg` gedrückt halten, während sie die Maus bewegen, verändert sich der Drehwinkel in Schritten von 15 Grad.
4. Klicken Sie nochmals, wenn der gewünschte Drehwinkel erreicht ist. Das Symbol wird dann mit diesem neuen Drehwinkel dargestellt und das Feld entsprechend gefüllt.

Klicken Sie mit der rechten Maustaste, um die Drehung des Punktymbols abzubrechen.

Punktsymbolversatz

Das Werkzeug Punktsymbolversatz ermöglicht es Ihnen, die Darstellungsposition von Punktsymbolen im Kartenfenster interaktiv zu verändern. Es funktioniert ähnlich wie das Werkzeug Punktsymbole drehen außer dass Sie ein Feld für den datendefinierten *Versatz (X,Y)* des Symbols einstellen müssen. Dieses Feld wird dann für diejenigen Objekte mit den Versatzkoordinaten gefüllt, deren Symbol Sie im Kartenfenster hin- und herschieben.

1. Stellen Sie ein feld für die datendefinierte Übersteuerung der Symboleigenschaft *Versatz (X,Y)* ein. Sollte das Symbol aus mehreren Layern bestehen, müssen Sie das Feld für jeden einzelnen davon einstellen
2. Aktivieren Sie das Werkzeug Punktsymbolversatz
3. Klicken Sie auf einen Punkt
4. Verschieben Sie ihn auf eine neue Position
5. Klicken Sie nochmals. Das Punktsymbol wird auf die neue Position bewegt. Die Werte für den Versatz werden im verknüpften Feld gespeichert.

   Klicken Sie mit der rechten Maustaste, um die Verschiebung des Punktymbols abzubrechen.

Bemerkung

Das Werkzeug Punktsymbolversatz verschiebt nicht das Punktobjekt selbst; dafür sollten Sie das Werkzeug Knotenwerkzeug (aktueller Layer) oder das Werkzeug Objekt verschieben benutzen.

Objekt abschneiden/ausdehnen

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:

- **Trimmen**: Entfernt Teile des Liniensegments oder der Polygonbegrenzung, die über die Schnittlinie hinausgehen
- **Erweitern**: Erweitert Polygongrenzen oder Liniensegmente, so dass sie an der Schnittlinie einrasten können.

Um bestehende Geometrien zu beschneiden oder zu erweitern:

1. Wählen Sie das Werkzeug Trim/Extend
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.

Bemerkung

Snapping is automatically enabled when this tool is activated. Your original snapping settings will be restored once the tool is deactivated.

Achtung

Achten Sie bei der Verwendung des Werkzeugs Trim/Extend auf die veränderte Geometrie. Abhängig von den Eingaben kann es ungültige Geometrien erzeugen, was möglicherweise zu Fehlern beim Speichern von Layern führt.

12.3.6 Formen digitalisieren

Die Symbolleiste *Form digitalisieren* bietet eine Reihe von Werkzeugen zum Zeichnen von Linien oder Polygonen mit Objekten regelmäßiger Formen. Sie ist mit der Form digitalisieren *Geometrie-Zeichenmethode* synchronisiert, die Sie in der *Digitalisierungs-Werkzeugleiste* auswählen können. Um sie zu benutzen:

1. Symbolleiste anzeigen: *Ansicht* ► *Symbolleisten* ► *Formen digitalisieren*
2. Wählen Sie ein Werkzeug, das die Form einer Geometrie erstellt oder verändert, z.B. Linienfeature hinzufügen, Polygonfeature hinzufügen, Teil hinzufügen, Ring hinzufügen, Reshape Features, …
3. Die Schaltfläche Digitalisieren mit Segment auf der *Digitalisier-Werkzeugleiste* ist aktiviert. Beim ersten Mal müssen Sie eventuell auf die Schaltfläche Form digitalisieren umschalten, um die Werkzeuge in der *Werkzeugleiste für Formen* zu aktivieren.
4. Wählen Sie ein Form-Digitalisierungswerkzeug und zeichnen Sie.

Kreisförmige Linie nach Radius

Die Schaltfläche Circular string by radius ermöglicht das Hinzufügen von Linien- oder Polygonobjekten mit einer kreisförmigen Geometrie, wenn zwei Knoten auf der Kurve und ein Radius gegeben sind:

1. Klicken Sie zweimal mit der linken Maustaste, um die beiden Punkte auf der Geometrie zu platzieren.
2. Ein *Radius*-Widget in der oberen rechten Ecke der Kartenansicht zeigt den aktuellen Radius an (entspricht dem Abstand zwischen den Punkten). Bearbeiten Sie dieses Feld auf den gewünschten Wert.
3. Eine Übersicht der Bögen, die diesen Bedingungen entsprechen, wird angezeigt, während Sie den Mauszeiger bewegen. Klicken Sie mit der rechten Maustaste, wenn der erwartete Bogen angezeigt wird, um ihn zu bestätigen.
4. Fügen Sie einen neuen Punkt hinzu, um einen weiteren Bogen zu formen.

Bemerkung

Gebogene Geometrien sind als solche nur in kompatiblen Datenanbieter gespeichert

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.

Kreise zeichnen

Um Kreise zu zeichnen, gibt es einen Satz von Werkzeugen. Sie werden im folgenden beschrieben.

Kreise werden in Kreisbögen umgewandelt. Falls das die jeweilige Datenquelle zulässt, werden sie als gebogene Geometrien gespeichert. Falls nicht, wird QGIS die Kreisbögen in Segmente aufteilen. Dies wird im Abschnitt in *Kreisförmige Linie nach Radius* beschrieben.

- Kreis aus 2 Punkten: Die beiden Punkte bestimmen den Durchmesser und die Ausrichtung des Kreises. (Linksklick, Rechtsklick)
- Kreis aus 3 Punkten: Zeichnet einen Kreis aus drei bekannten Punkten auf dem Kreis. (Linksklick, Linksklick, Rechtsklick)
- Kreis durch einen Mittelpunkt und einen anderen Punkt: Zeichnet einen Kreis mit einem gegebenen Mittelpunkt und einem Punkt auf dem Kreis (Linksklick, Rechtsklick). In Verbindung mit dem *Das Bedienfeld Erweiterte Digitalisierung* kann dieses Werkzeug zu einem „Kreis aus Mittelpunkt und Radius hinzufügen“-Werkzeug werden, indem der Abstandswert nach dem ersten Klick gesetzt und fixiert wird.
- Kreis aus 3 Tangenten: Zeichnet einen Kreis, der tangential zu drei Segmenten verläuft. **Beachten Sie, dass Sie das Einrasten an Segmenten aktivieren müssen** (Siehe Einrasten_toleranz). Klicken Sie auf ein Segment, um eine Tangente hinzuzufügen. Wenn zwei Tangenten parallel sind, werden die Koordinaten des Klicks auf die erste parallele Tangente verwendet, um die Position des Kreises zu bestimmen. Wenn drei Tangenten parallel sind, erscheint eine Fehlermeldung und die Eingabe wird gelöscht. (Linksklick, Linksklick, Rechtsklick)
- Kreis aus 2 Tangenten und einem Punkt: Ähnlich wie Kreis aus 3 Tangenten, nur dass Sie zwei Tangenten auswählen, einen Radius eingeben und den gewünschten Mittelpunkt wählen müssen.

Ellipsen zeichnen

Um Ellipsen zu zeichnen, gibt es einen Satz von Werkzeugen. Sie werden im folgenden beschrieben.

Ellipsen können nicht in bogenförmige Geometrien umgewandelt werden, so dass sie immer segmentiert werden.

- Ellipse aus Zentrum und zwei Punkten: Zeichnet eine Ellipse mit gegebenem Zentrum, Hauptachse und Nebenachse. (Linksklick, Linksklick, Rechtsklick)
- Ellipse aus Mittelpunkt und einem Punkt: Zeichnet eine Ellipse in einen Begrenzungsrahmen mit dem Mittelpunkt und einem Eckpunkt. (Linksklick, Rechtsklick)
- Ellipse aus Umfang: Zeichnet eine Ellipse in einen Begrenzungsrahmen mit zwei gegenüberliegenden Ecken. (Linksklick, Rechtsklick)
- Ellipse aus Brennpunkten: Zeichnet eine Ellipse durch 2 Punkte als Brennpunkte und einen Punkt auf der Ellipse. (Linksklick, Linksklick, Rechtsklick)

Rechtecke zeichnen

Ihnen stehen eine Reihe von Werkzeugen zum Zeichnen von Rechtecken zur Verfügung, sie werden nachfolgend beschrieben.

- Rechteck aus Zentrum und einem Punkt hinzufügen: Zeichnet ein Rechteck, dass über seinen Mittelpunkt und eine Ecke definiert wird. (Linksklick, Rechtsklick)
- Rechteck nach Ausdehnung hinzufügen: Zeichnet ein Rechteck, das über zwei gegenüberliegende Ecken definiert wird. (Linksklick, Rechtsklick)
- Rechteck über drei Punkte hinzufügen (Abstand vom 2. und 3. Punkt): Zeichnet ein orientiertes Rechteck aus drei Punkten. Der erste und der zweite Punkt definieren Länge und Winkel der ersten Seiote. Der dritte Punkt legt die Länge der anderen Seite fest. Sie können das Bedienfeld *Das Bedienfeld Erweiterte Digitalisierung* benutzen, um die Länge der Seiten festzulegen. (Linksklick, Linksklick, Rechtsklick)
- Rechteck über drei Punkte hinzufügen (Abstand vom projizierten Punkt auf Segment und 1. und 2. Punkt): Die Funktion ist genau so wie im vorstehenden Werkzeug, allerdings wird die Länge der zweiten Seite über die Projektion des dritten Punktes auf die erste Seite berechnet. (Linksklick, Linksklick, Rechtsklick)

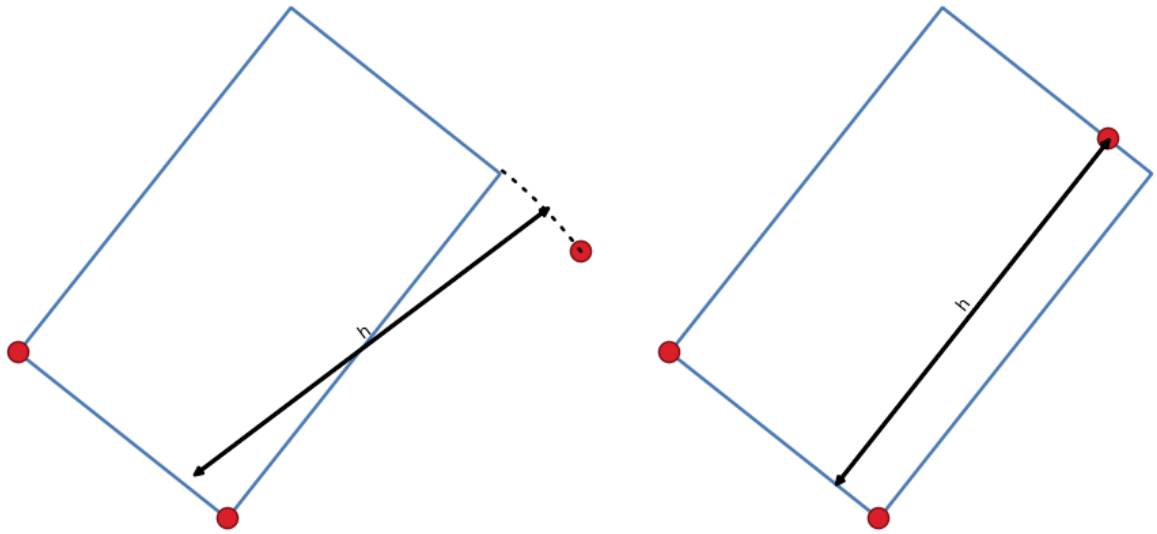


Abb. 12.92: Ein Rechteck über drei Punkte hinzufügen, mittels Abstand (rechts) und projiziertem Punkt (links)

Regelmäßige Polygone zeichnen

Ihnen stehen eine Reihe von Werkzeugen zum Zeichnen von regelmäßigen Polygonen zur Verfügung, sie werden nachfolgend beschrieben. Klicken Sie mit links, um den ersten Punkt zu setzen. In dem danach erscheinenden Eingabefeld können Sie die Anzahl der Seiten festlegen. Klicken Sie mit rechts, um das regelmäßige Polygon abzuschließen.

- Regelmäßiges Polygon aus zwei Punkten hinzufügen: Zeichnet ein regelmäßiges Polygon, bei dem Länge und Winkel der ersten Seite über zwei Punkte definiert werden.
- Regelmäßiges Polygon aus Zentrum und einem Punkt hinzufügen: Zeichnet ein regeläßiges Polygon über seinen Mittelpunkt. Der zweite Punkt bestimmt Winkel und Abstand einer Seite zu diesem Mittelpunkt.
- Regelmäßiges Polygon aus Zentrum und einer Ecke hinzufügen: Die selbe Funktion wie das vorhergehende Werkzeug, jedoch bestimmt der zweite Punkt Winkel und Abstand zu einem Eckpunkt des Polygons.

12.3.7 Das Bedienfeld Erweiterte Digitalisierung

Bei der Erfassung, Umformung und Teilung neuer oder bestehender Geometrien haben Sie auch die Möglichkeit, das Bedienfeld Erweiterte Digitalisierung zu verwenden. Sie können Linien exakt parallel oder senkrecht zu einem bestimmten Winkel digitalisieren oder Linien an bestimmte Winkel binden. Außerdem können Sie Ihre neue Geometrie durch die Eingabe von X- und Y-Koordinaten sowie Z für 3D-Objekte oder M-Werte genau definieren.

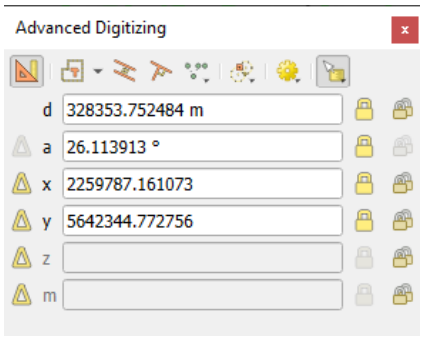


Abb. 12.93: Das Bedienfeld Erweiterte Digitalisierung

Das Bedienfeld *Erweiterte Digitalisierung* kann entweder mit einem Rechtsklick auf die Symbolleiste, über das Menü *Ansicht* ► *Bedienfelder* ► oder durch Drücken von `Strg+4` geöffnet werden. Sobald das Bedienfeld sichtbar ist, klicken Sie auf die Schaltfläche Erweiterte Digitalisierungswerkzeuge aktivieren, um den Werkzeugsatz zu aktivieren.

Bemerkung

Die Werkzeuge sind nicht aktiviert, wenn die Kartenansicht geographische Koordinaten zeigt.

Die Werkzeuge zielen darauf ab, Koordinaten, Längen und Winkel zu sperren, während Sie den Mauszeiger beim Digitalisieren über das Kartenfenster bewegen.

Sie können auch Festsetzungen mit relativer oder absoluter Referenz vornehmen. Relative Referenz bedeutet, dass die Festsetzungen für den folgenden Stützpunkt relativ zum vorhergehenden Stützpunkt oder Segment sind.

Die Werkzeuge

Oben im Bedienfeld *Erweiterte Digitalisierung* finden Sie folgende Schaltflächen:

- Erweiterte Digitalisierungswerkzeuge einschalten
- Konstruktionsmodus: ermöglicht es Ihnen, Ihre Klickpunkte zu erfassen, um sie als Referenzpunkte für das Sperren der Entfernung, des Winkels oder der Relativwerte für X, Y, Z oder M zu benutzen. Mehr dazu unter *Konstruktionsmodus*.
- Parallel ermöglicht es Ihnen, eine Linie parallel zu einer vorhandenen zu zeichnen (mehr dazu unter *Parallele und senkrechte Linien*)
- Senkrecht ermöglicht es Ihnen, eine Linie senkrecht zu einer vorhandenen zu zeichnen (mehr unter *Parallele und senkrechte Linien*)
- Konstruktionswerkzeuge bietet eine Reihe von Optionen, die die Platzierung der Stützpunkte auf der Grundlage der extrapolierten Koordinaten der vorhandenen Elemente einschränken:
  - *Line Extension*: Bewegen Sie den Mauszeiger über ein Segment und Sie erhalten eine lila gepunktete Linie, die das Segment über die Kartenansicht verlängert. Sie können den Stützpunkt überall auf dieser virtuellen Linie einrasten lassen.

  - *X/Y Punkt*: Bewegen Sie den Mauszeiger über einen Scheitelpunkt und Sie erhalten eine lila gepunktete Linie entlang seiner X- oder Y-Koordinate in der Kartenansicht. Sie können den Scheitelpunkt an einer beliebigen Stelle auf dieser virtuellen Linie einrasten. Es ist sogar möglich, den Mauszeiger über zwei verschiedene Eckpunkte zu bewegen, wobei für beide virtuelle Koordinatenlinien erzeugt werden, an deren Schnittpunkt man dann einrasten kann.

- 2-Kreispunktüberschneidungen: Ermöglicht die Digitalisierung eines Punktes oder Stützpunktes am Schnittpunkt zweier Kreise. (Weitere Informationen unter *2-Kreispunktüberschneidung*).

- Fangen an gemeinsamen Winkeln: zeigt beim Bewegen des Cursors eine virtuelle Linie an, an der Sie einrasten können, um den nächsten Stützpunkt hinzuzufügen. Die Fanglinie wird durch den zuletzt hinzugefügten Stützpunkt und einen (absoluten oder relativen zum vorherigen Segment) Winkel aus einer voreingestellten Liste definiert (in Schritten von 0,1°, 0,5°, 1°, 5°, 10°, 15°, 18°, 22,5°, 30°, 45° oder 90°). Wählen Sie *Nicht an gemeinsamen Winkeln fangen*, um dieses Objekt zu deaktivieren.

  Fangen an Objekten kann zusammen mit dem Fangen an gemeinsamen Winkeln für eine genaue Digitalisierung verwendet werden. Für eine feinere Steuerung, wie das zu fangende Zielelement beibehalten werden soll, können Sie unter dem Eintrag *Fangpriorität* angeben, ob das Fangen an Objekten Vorrang vor gemeinsamen Winkeln haben soll und umgekehrt. Sie können während des Digitalisierungsvorgangs von einer Methode zur anderen wechseln, sodass keine der Fangoptionen in der Zwischenzeit deaktiviert wird. Drücken Sie `N` (oder `Umschalt+N`) während eines Digitalisierungsvorgangs, um die Liste der Winkel zu durchlaufen.

- Floater-Einstellungen: wenn der Punkt *Floater anzeigen* markiert ist, folgt ein Kontextmenü mit Digitalisierungsinformationen dem Cursor während der Digitalisierung. Die Werte können über das Bedienfeld mit den Shortcuts 1 aufgerufen, bearbeitet und nach der Bestätigung (Drücken von `Eingabe`) mit Locked gesperrt werden. Die Art der anzuzeigenden Informationen kann im unteren Teil des Menüs ausgewählt werden:
  - *Entfernung anzeigen*
  - *Winkel anzeigen*
  - *XY-Koordinaten anzeigen*
  - *Z-Wert anzeigen*
  - *M-Wert anzeigen*
  - *Kurs/Azimut anzeigen*
  - *Gemeinsamen Einrastwinkel anzeigen*

Unterhalb der Symbolleiste finden Sie eine Reihe von Textfeldern, deren Wert standardmäßig die Position oder Bewegung des Cursors in der Kartenansicht widerspiegelt. Wenn Sie diese Werte bearbeiten, können Sie die Position der zu bearbeitenden Elemente einschränken:

- *d* für den Abstand von einer Referenzposition, in der Regel dem zuletzt bearbeiteten Stützpunkt
- *a* für den Winkel (absolut oder relativ) von einer Bezugsposition, in der Regel dem zuletzt bearbeiteten Segment
- *x* für die X-Koordinate des Zeigers
- *y* für die Y-Koordinate des Zeigers
- *z* für den Standard-Z-Wert oder die Z-Koordinate des Stützpunkt oder Segments unter dem Zeiger
- *m* für den Standard-M-Wert oder den M-Wert des Stützpunkt oder Segments unter dem Zeiger

Tastenkürzel

Um die Arbeit mit dem Bedienfeld Erweiterte Digitalisierung zu beschleunigen, gibt es eine Reihe von Tastenkürzeln:

Tab. 12.5: Tastenkürzel für die Werkzeuge des Bedienfelds Erweiterte Digitalisierung

Taste	nur Taste	Strg+ or Alt+	Shift+
D	Abstand einstellen	Abstand beibehalten	
A	Winkel einstellen	Winkel beibehalten	Winkel relativ zum letzten Segment an-/ausschalten
X	X-Koordinate einstellen	X-Koordinate beibehalten	X relativ zum letzten Stützpunkt an-/ausschalten
Y	Y-Koordinate einstellen	Y-Koordinate beibehalten	Y relativ zum letzten Stützpunkt an-/ausschalten
Z	Z-Koordinate einstellen	Z-Koordinate beibehalten	Z relativ zum letzten Stützpunkt an-/ausschalten
M	M-Wert einstellen	M-Wert beibehalten	M relativ zum letzten Stützpunkt an-/ausschalten
C	Konstruktionsmodus an-/ausschalten		
P	Zwischen senkrechtem und parallelem Modus umschalten		

Bemerkung

Z-Koordinate und M-Wert sind nur verfügbar, wenn die Geometriedimension des Layers dies hergibt.

Mit absoluter Referenz digitalsieren

Sobald Sie eine neue Geometrie von Grund auf neu zeichnen, kann es hilfreich sein, den ersten Stützpunkt an festgelegten Koordinaten zu setzen.

Wenn Sie z.B. ein neues Objekt zu einem Polygonlayer hinzufügen, klicken Sie aud die Schaltfläche . Sie können die genauen Koordinaten angeben, an denen Sie die Digitalisierung beginnen möchten, d.h.:

1. klicken Sie auf das Eingabefeld *x* (oder benutzen das Tastenkürzel X),
2. geben Sie die X-Koordinate ein und drücken Sie Eingabe oder klicken Sie auf die Schaltfläche rechts davon, um die Mausposition auf der X-Achse im Kartenfenster zu beizubehalten,
3. klicken Sie auf das Eingabefeld *y* (oder benutzen das Tastenkürzel Y),
4. geben Sie die Y-Koordinate ein und drücken Sie Eingabe oder klicken Sie auf die Schaltfläche rechts davon, um die Mausposition auf der Y-Achse im Kartenfenster zu beizubehalten.
5. Wenn der Layer Z-Koordinaten oder M-Werte hat, wird das entsprechende *z* oder *m*-Widget aktiviert und zeigt den Standardwert an, der unter *Einstellungen* ► *Optionen…* auf dem Reiter *Digitalisierung* eingestellt ist.
    1. Klicken Sie auf das Textfeld *z* oder *m* (oder verwenden Sie das Tastaturkürzel Z oder M).
    2. Geben Sie den gewünschten Koordinatenwert ein und drücken Sie Enter oder klicken Sie auf die Schaltfläche rechts daneben, um den Wert im Widget zu sperren.

    Bemerkung

    Lesen Sie *Regeln für die Zuordnung von Z-Koordinaten oder M-Werten* für Details darüber, wie Z-Koordinaten und M-Werte automatisch aus bestehenden Objekten bestimmt werden.

6. Zwei blaue gepunktete Linien und ein grünes Kreuz kennzeichnen die genauen Koordinaten, die Sie eingegeben haben. Klicken Sie auf die Kartenansicht, um einen Scheitelpunkt an der Position des grünen Kreuzes hinzuzufügen.

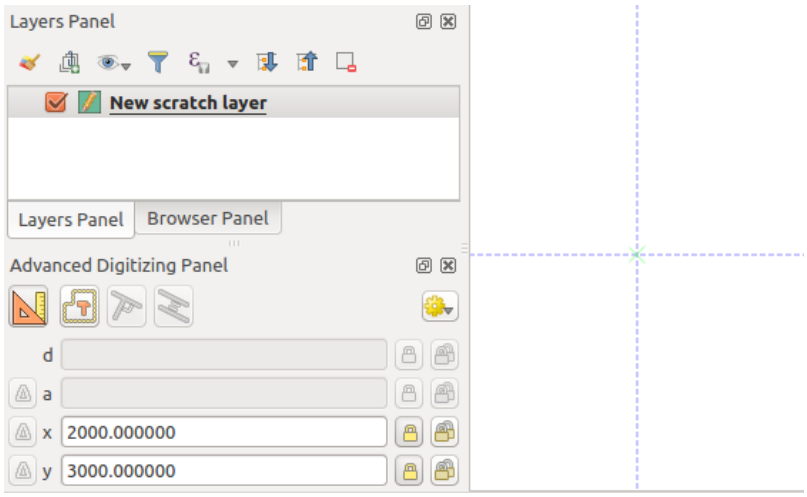


Abb. 12.94: Eine Zeichnung mit vorgegebenen Koordinaten beginnen

7. Sie können wie oben fortfahren und einen neuen Koordinatensatz für den nächsten Scheitelpunkt hinzufügen oder zu einem anderen *Digitalisierungsmodus* wechseln (z.B. Segment, Kurve oder Stream).
8. Falls Sie ein Segment bestimmter Länge zeichnen möchten:
   1. Klicken Sie auf das Eingabefeld *d (Entfernung)* (Tastenkürzel `D` , Entfernung = engl. distance)
   2. Tragen Sie die Entfernung ein (in Karteneinheiten)
   3. Drücken Sie `Eingabe` oder klicken Sie auf die Schaltfläche rechts, um den Mauszeiger im Kartenfenster auf die Länge des Segments zu sperren (also die Länge beizubehalten). Um den letztgesetzten Stützpunkt wird ein Kreis gezeichnet, dessen Radius der eingegebenen Entfernung entspricht. Ein Kreuz auf der Kreislinie zeigt Ihnen die Position des nächsten Stützpunktes an, falls Sie nun klicken.

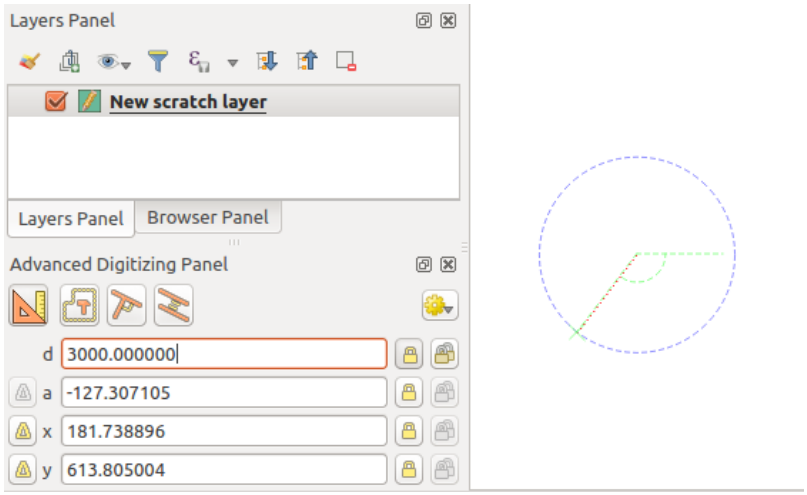


Abb. 12.95: Segment mit einer vorgegebenen Länge

9. Sie können auch die Position des Stützpunktes beschränken, indem Sie den Winkel des Segments setzen wie oben beschrieben:
   1. Klicken Sie auf das Eingabefeld *a (Winkel)* (Tastenkürzel `A` , Winkel = engl. angle)

2. Geben Sie den Winkel (in Grad) ein

3. Drücken Sie `Eingabe` oder klicken Sie auf die Schaltfläche rechts, um den Winkel zu sperren. Durch den letztgesetzten Stützpunkt wird eine Linie gezeichnet, die um den eingegebenen Winkel gedreht ist. Ein Kreuz auf dieser Linie zeigt Ihnen die Position des nächsten Stützpunktes an, falls Sie nun klicken.

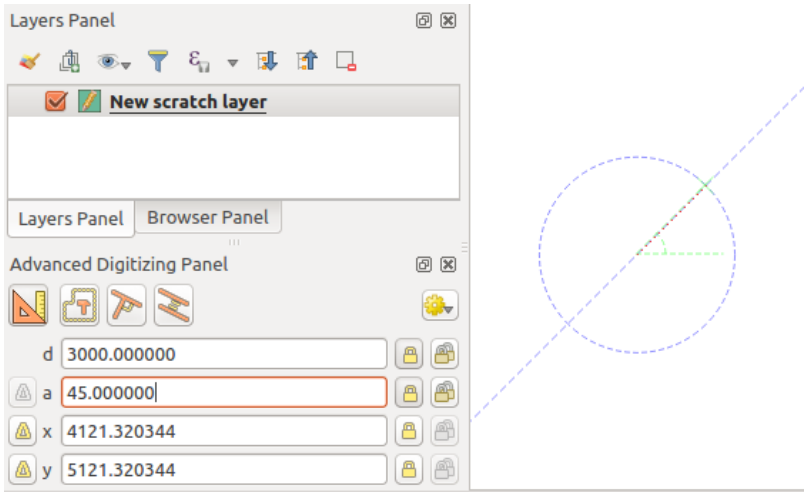


Abb. 12.96: Segment mit vorgegebenem Winkel

Hinweis

Drücken Sie `Strg+<key>` oder `Alt+<key>`, um die die Eingabe zu sperren den Wert automatisch bearbeitbar zu machen. Nun müssen Sie nur noch `Eingabe` drücken. Zusammen mit der Anzeige über Karten umschalten kann das Digitalisieren über die Tastatur sehr zeitsparend sein.

Mit relativer Referenz digitalsieren

Anstatt absolute Werte für Winkel oder Koordinaten können Sie Werte relativ zum zuletzt eingegebenen Stützpunkt oder Segment benutzen.

Bei Winkeln klicken Sie auf die Schaltflläche links der Eingabezeile *a* (oder drücken Sie `Umschalt+A`) um Winkeleingabe relativ zum zuletzt eingegebenen Segment ein- bzw. auszuschalten. Ist diese Option eingeschaltet, werden Winkel zwischen dem zuletzt eingegebenen Segment und der Position des Mauszeigers gemessen.

Klicken Sie auf die Schaltflächen links der Eingabefelder *x*, *y*, *z* oder *m* (oder drücken Sie `Umschalt+<key>`), um die Koordinateneingabe relativ zum zuletzt gesetzten Stützpunkt ein- oder auszuschalten. Sind diese Optionen eingeschaltet, werden die Koordinaten so berechnet, als sei der letztgesetzte Stützpunkt der Koordinatenursprung.

Laufendes Beibehalten

Sowohl beim Digitalisieren mit absoluter als auch mit relativer Referenz können Sie den Winkel, die Entfernung, X, Y, Z und M Einschränkungen laufend beibehalten, indem auf die Schaltflächen *laufend beibehalten* klicken. Damit können Sie mehrere Punkte oder Stützpunkte digitalisieren und dabei die selben Einschränkungen benutzen.

Konstruktionsmodus

Sie können den *Konstruktionsmodus* an- und abschalten, indem Sie auf die Schaltfläche Konstruktionsmodus klicken oder mit Tastenkürzel `C` . Während Sie sich im Konstruktionsmodus befinden, werden durch Klicks in die Kartenansicht keine neuen Stützpunkte hinzugefügt; stattdessen werden die Klickpositionen erfasst, so dass Sie sei als Referenzpunkte benutzen können, um die relativen Werte der Entfernung, des Winkels oder von X, Y, Z, M zu sperren.

Neben der Schaltfläche Konstruktionsmodus finden Sie ein Dropdown-Menü, in dem Sie folgende Optionen auswählen können:

- *Konstruktionsleitlinien aufzeichnen*: Alle Konstruktionsschritte werden als gestrichelte Linien dargestellt. Jeder von Ihnen ausgeführte Schritt wird visuell dargestellt, sodass Sie den Konstruktionsprozess nachvollziehen können. Die Hilfslinien werden angezeigt, solange *Erweiterte Digitalisierung* aktiv bleibt.
- *Konstruktionsleitlinien anzeigen*: Ermöglicht es Ihnen, Konstruktionsleitlinien in der Kartenansicht ein- oder auszublenden. Wenn diese Option aktiviert ist, werden alle aktiven Hilfslinien angezeigt, was eine bessere räumliche Orientierung und Präzision während des Digitalisierungsprozesses ermöglicht.
- *Auf sichtbare Konstruktionsleitlinien einrasten*: Die Führungen sind einrastbar, sodass Sie neue Konstruktionsschritte an jeder beliebigen Stelle entlang der vorhandenen Konstruktionsleitlinien beginnen können.
- Wählen Sie *Konstruktionsleitlinien löschen*, um alle Hilfslinien aus der Kartenansicht zu entfernen.

Der Konstruktionsmodus kann z.B. benutzt werden, um einen Punkt in einer genau festgelegten Entfernung von einem bestehenden Punkt zu zeichnen.

Haben Sie einen Punkt in der Kartenansicht und der Einrastmodus ist richtig eingeschaltet, können Sie sehr einfach andere Punkte in einer vorgegebenen Entfernung und im vorgegebenen Winkel dazu zeichnen. Zusätzlich zur Schaltfläche müssen Sie den *Konstruktionsmodus* durch Klick auf die Schaltfläche Konstruktionsmodus oder Tastenkürzel `C` aktivieren.

Klicken Sie in die Nähe des Punktes, von dem aus Sie den Abstand berechnen wollen und klicken Sie auf die Eingabe *d* (Tastenkürzel `D`). Geben Sie den gewünschten Abstand ein und drücken Sie `Eingabe`, um die Mausposition im Kartenfenster beizubehalten:

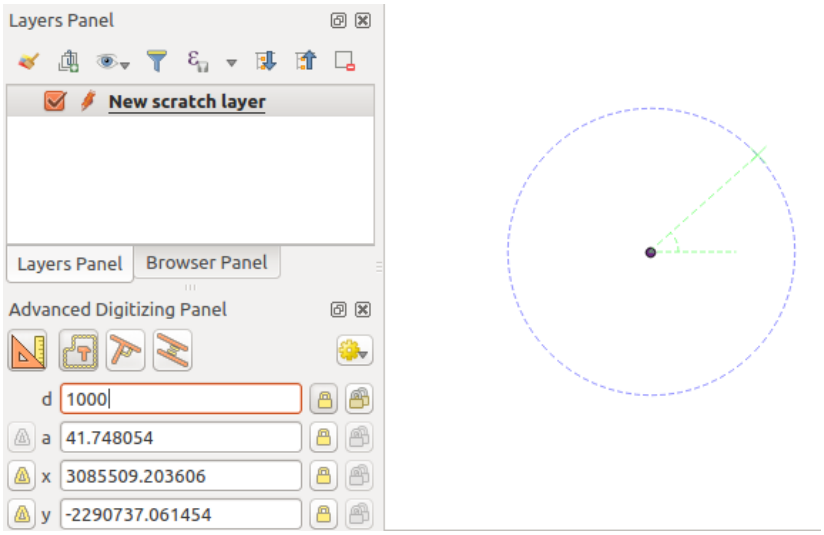


Abb. 12.97: Abstand von einem Punkt

Bevor Sie den neuen Punkt eingeben, drücken Sie `C`, um den Konstruktionsmodus zu verlassen. Jetzt können Sie ins Kartenfenster klicken und der Punkt wird mit dem eingegebenen Abstand platziert.

Sie können zusätzlich die Winkeleinstellung benutzen, um beispielsweise einen zweiten Punkt in der selben Entfernung von Ihrem Originalpunkt aber in einem bestimmten Winkel zu Ihrem eben neu gesetzten Punkt zu erzeugen. Klicken Sie auf die Schaltfläche Konstruktionsmodus oder Tastenkürzel `C` , um in den Konstruktionsmodus zu gelangen. Klicken Sie auf den eben neu gesetzten Punkt und dann auf den anderen, um ein Richtungssegment zu erhalten. Klicken Sie dann in die Texteingabe *d* (Tastenkürzel `D`), geben Sie die gewünschte Entfernung ein und drücken Sie `Eingabe`. Klicken Sie auf die Texteingabe *a* (Tastenkürzel `A`), geben Sie den gewünschten Winkel ein und drücken Sie `Eingabe`. Die Mausposition wird in Entfernung und Winkel gesperrt.

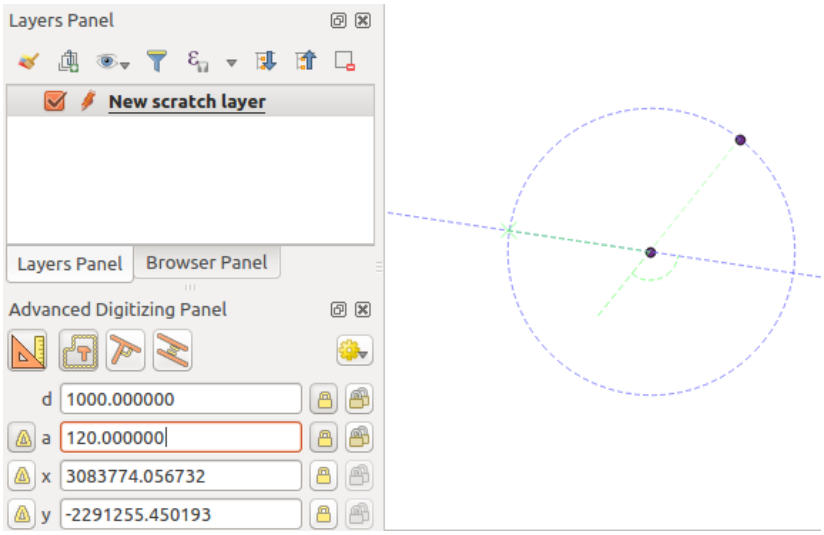


Abb. 12.98: Abstand von und Winkel zu einem Punkt

Bevor Sie den neuen Punkt hinzufügen, drücken Sie `C`, um den Konstruktionsmodus zu verlassen. Jetzt können Sie in das Kartenfenster klicken und der Punkt wird in dem eingegebenen Abstand und Winkel erzeugt. Sie können den Vorgang wiederholen, um mehrere Punkte zu erzeugen.

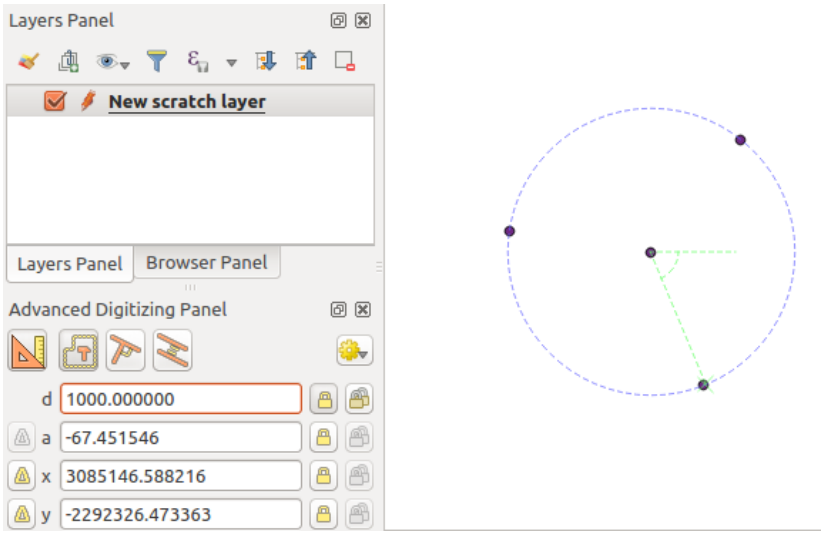


Abb. 12.99: Punkte in festgelegtem Abstand und Winkel

Parallele und senkrechte Linien

Alle bisher beschriebenen Werkzeuge können Sie mit den Werkzeugen Senkrecht und Parallel zusammen benutzen. Mit diese beiden Werkzeuge zeichnen Sie Segmente exakt rechtwinklig oder parallel zu anderen Segmenten. Das entsprechende Segment kann zu einem anderen Layer, zu einem anderen Objekt im selben Layer oder auch zu dem Objekt, an dem Sie gerade arbeiten, gehören (letzteres benötigt *Selbsteinrasten*).

So zeichnen Sie ein *senkrechtes* Segment:

1. Fügen Sie zunächst einen der beiden Stützpunkte des Segments hinzu.
2. Zum Aktivieren klicken Sie auf die Schaltfläche Senkrecht (Tastenkürzel `P` , senkrecht engl. perpendicular).
3. Klicken Sie dann auf das Segment, auf dem Ihr neues senkrecht stehen soll.
4. Eine gepunktete Hilfslinie senkrecht zu diesem Segment durch den zuletzt gesetzten Stützpunkt erscheint. Die Winkeleinstellung ist gesperrt, wodurch der nächste Stützpunkt auf dieser Linie liegen muss und ein Kreutz zeigt Ihnen die zukünftige Position an. Klicken Sie, um einen neuen Stützpunkt zu setzen.

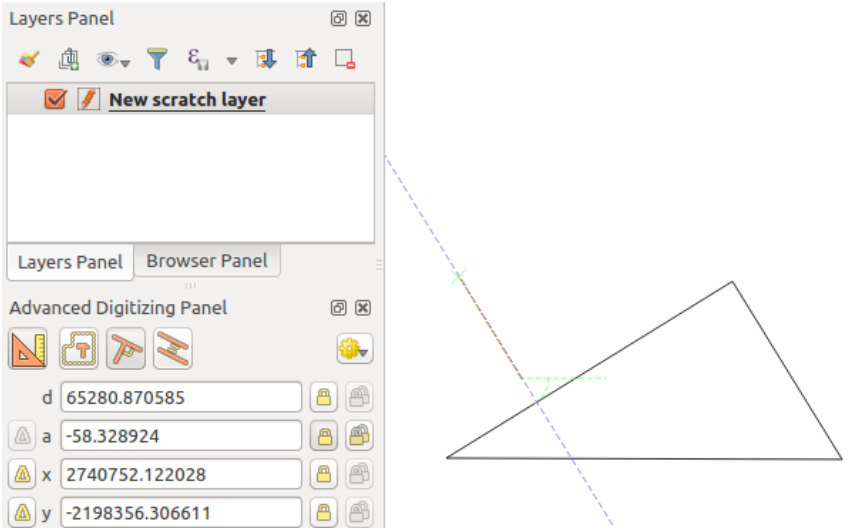


Abb. 12.100: Senkrechtes Digitalisieren

Ein *paralleles* Segment zeichnen Sie genau so mit der Ausnahme, dass Sie auf die Schaltfläche Parallel (Tastenkürzel zweimal `P`).

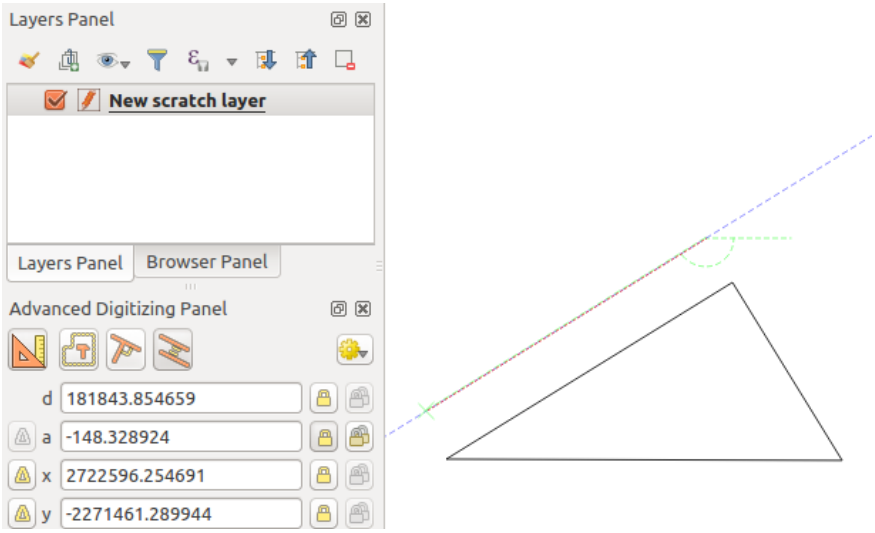


Abb. 12.101: Paralleles Digitalisieren

Beide Werkzeuge stellen den richtigen Winkel für paralleles und senkrechtes Arbeiten fest und sperren ihn, während Sie digitalisieren. Entsperren Sie den Winkel, um die Benutzung zwischendurch abzubrechen.

2-Kreispunktüberschneidung

Um einen Stützpunkt am Schnittpunkt zweier Kreise hinzuzufügen, gehen Sie wie folgt vor:

1. Klicken Sie auf das 2-Kreispunktüberschneidung-Icon.
2. Es öffnet sich ein Dialogfeld, in dem Sie die Parameter für *Kreis # 1* und *Kreis # 2* festlegen können.
3. Klicken Sie auf die Kartenansicht, und das Tool berechnet automatisch die Koordinaten *X* und *Y* für die Mittelpunkte beider Kreise.
4. Geben Sie für jeden Kreis den Abstand *d* ein.
5. Das Tool berechnet und zeigt die beiden Schnittpunkte der Kreise an.
6. Klicken Sie auf einen der Schnittpunkte, um den neuen Stützpunkt hinzuzufügen.

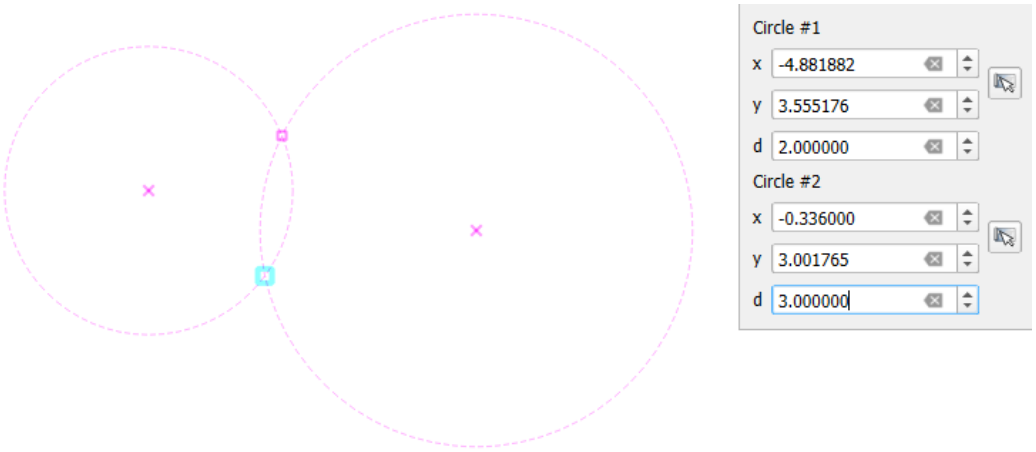


Abb. 12.102: 2-Kreispunktüberschneidung

12.3.8 Mit Verarbeitung Objekte innerhalb eines Layers verändern

Das Menü *Verarbeitung* biete eine große Zahl von Werkzeugen zur Analyse und Erzeugung neuer Objekte auf der Basis der Eigenschaften vorhandener Objekte oder ihrer Beziehung zu anderen Objekten (innerhalb oder außerhalb des selben Layers). Während üblicherweise als Ergebnis neue Layer erzeugt werden, erlauben es manche Algorithmen, den Eingabelayer selbst zu verändern. Das ist praktisch, um mehrfache Objektänderungen mittels fortschrittlicher und komplexer Methoden zu automatisieren.

So ändern Sie Objekte insitu:

1. Wählen Sie den Layer, den Sie bearbeiten möchten, im Bedienfeld *Layer* aus.
2. Wählen Sie die entsprechenden Objekte aus. Diesen Schritt können Sie überspringen, dann werden die Änderungen auf den ganzen Layer angewendet.
3. Drücken Sie die Schaltfläche Objekt insitu ändern oben in *Verarbeitungswerkzeugen*. Die Liste der Algorithmen ist gefiltert, es werden nur diejenigen angezeigt, die mit der insitu-Verarbeitung funktionieren, d.h.:
   - Sie arbeiten an der Objektquelle und nicht auf Ebene des Layers.
   - Sie ändern die Struktur des Layers nicht, z.B. Felder anlegen oder entfernen.
   - Sie ändern den Geometrietyp nicht, z.B. von Linie zu Punkt.

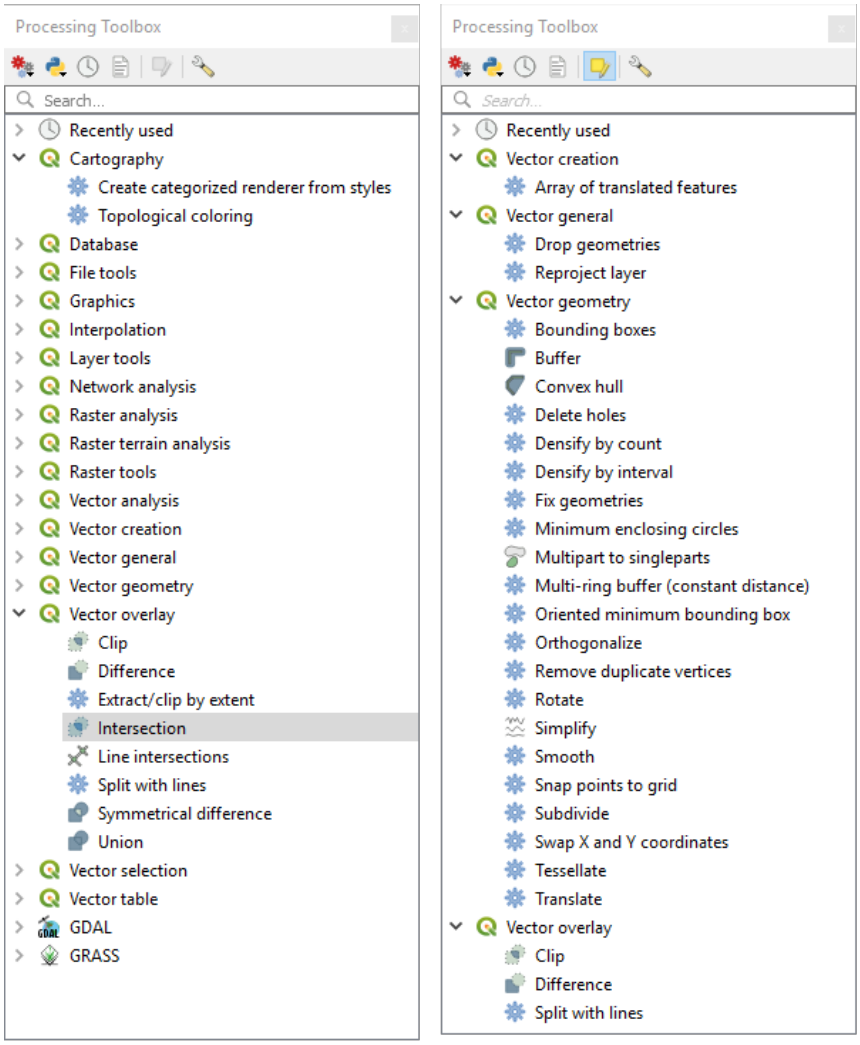


Abb. 12.103: Verarbeitungswerkzeuge: alle (links) und insitu-Änderung (rechts)

4. Wählen Sie den Algorithmus, den Sie ausführen wollen und doppelklicken Sie darauf.

   Bemerkung

   Falls der Algorithmus keine weiteren Nutzereingaben erfordert (abgesehen von Eingabelayer und Ausgabelayer), wird er sofort ausgeführt, ohne dass sich das Eingabefenster öffnet.

   1. Falls der Algorithmus andere Nutzereingaben als Ein- und Ausgabelayer erfordert, öffnet sich sein Eingabefenster. Geben Sie alle Pflichtangaben ein.
   2. Klicken Sie auf *Gewählte Objekte ändern* oder *Alle Objekte ändern* je nachdem, ob des eine aktive Auswahl gibt.

   Die Änderungen werden auf den Layer angewendet und im Editierspeicher festgehalten: tatsächlich wird der Bearbeitungsstatus des Layers angeschaltet, wie Sie am Symbol neben dem Layernamen erkennen können.

5. Drücken Sie wie üblich Layeränderungen speichern, um die Änderungen im Layer zu speichern. Sie können stattdessen auch auf Rückgängig klicken, um alle Änderungen rückgängig zu machen.

12.4 Verbinden und Bearbeiten von Daten auf verschiedenen Layern

Die Fähigkeit, Daten aus verschiedenen Layern zu verbinden, ist eine der Aufgaben einer GIS-Software. Eine solche Verbindung kann auf der räumlichen Beziehung zwischen den Objekten oder auf ihren gemeinsamen Attributen beruhen. QGIS bietet Werkzeuge, um jede dieser Verbindungen zu handhaben, wie zum Beispiel:

- Verarbeitungsalgorithmen, die als Ergebnis der Verbindung einen neuen Layer erzeugen können, nämlich *Join attributes by location*, *Join attributes by nearest*, *Join attributes by field value*, …
- SQL-Abfragen zur Erstellung eines neuen Layers aus dem *DB Manager* oder als *virtueller Layer*
- *Verknüpfungseigenschaften* oder *Beziehungseinstellungen*, die Attribute von Objekten in einem bestimmten Layer vorübergehend mit denen von Objekten in einem anderen Layer auf der Grundlage eines oder mehrerer übereinstimmender Attribute erweitern.

  Verknüpfungen und Beziehungen sind technische Konzepte, die aus Datenbanken übernommen wurden, um die in Tabellen gespeicherten Daten durch die Kombination ihrer Inhalte optimal zu nutzen. Die Idee ist, dass Objekte (Zeilen) aus verschiedenen Layern (Tabellen) miteinander verbunden werden können. Die Anzahl der Zeilen, die miteinander übereinstimmen, kann einen beliebigen Wert haben (null, eins, viele).

12.4.1 Verknüpfung von Objekten zwischen zwei Layern

Verknüpfungen in QGIS erlauben es, Objekte des aktuellen Layers mit Objekten aus einem anderen geladenen Layer zu verknüpfen. Dabei spielt es keine Rolle, ob sie räumlich Daten enthalten und um welche Art von Geometrie es sich handelt. Die Verknüpfung basiert auf einem Attribut, das von den Layern in einer Eins-zu-Eins-Beziehung gemeinsam genutzt wird.

Um eine Verknüpfung auf einem Layer (im folgenden als `Ziellayer` bezeichnet) zu erstellen:

1. Gehen Sie in den *Eigenschaften* ► des Layers auf den Reiter *Verknüpfungen*
2. Klicken Sie auf die Schaltfläche Neue Verknüpfung hinzufügen. Der Dialog *Vektorverknüpfung hinzufügen* erscheint.
3. Wählen Sie unter *Layer verknüpfen*, den Layer aus, den Sie mit dem Ziellayer verbinden wollen
4. Geben Sie das *Verknüpfungsfeld* (aus `Layer verknüpfen`) und das *Zielfeld* (aus dem `Ziellayer`) an. Diese Felder werden für die Suche nach übereinstimmenden Objekten in beiden Layern verwendet und sollten daher gemeinsame Werte haben.
5. Drücken Sie *OK* und eine Zusammenfassung der ausgewählten Parameter wird auf dem Reiter *Verknüpfungen* hinzugefügt.

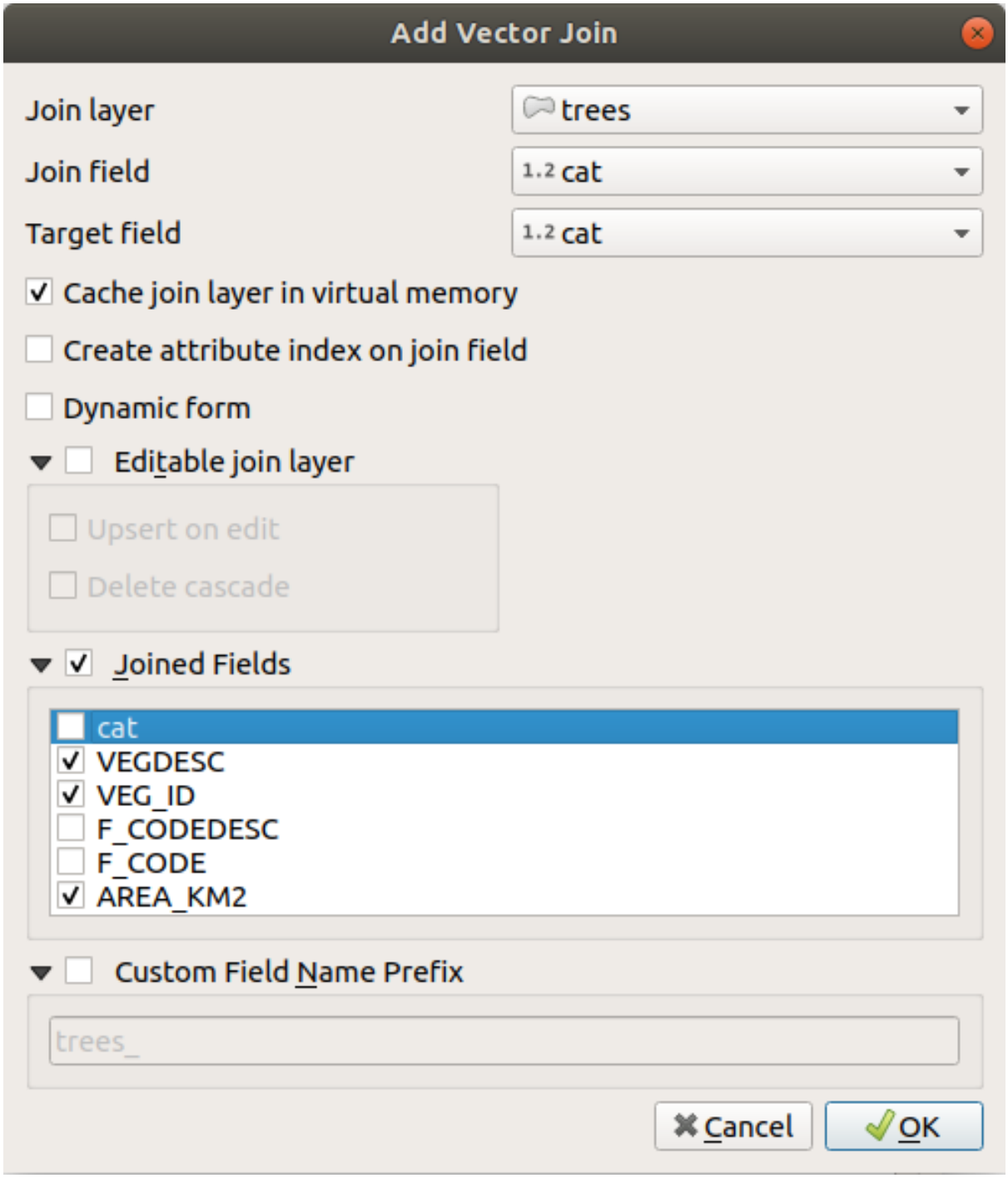


Abb. 12.104: Verknüpfe eine Attributtabelle zu einem vorhandenen Vektorlayer

Mit den obigen Schritten wird eine Verknüpfung erstellt, bei der **alle** Attribute des **ersten übereinstimmenden Objekts** im Layer der Verknüpfung zum Objekt des Ziellayers hinzugefügt werden. Die folgende Logik wird für die Paarung von Objekten während eines Verknüpfungsprozesses verwendet:

- Alle Objekte im Ziellayer werden zurückgegeben, unabhängig davon, ob sie eine Übereinstimmung aufweisen
- Wenn das Zielfeld doppelte Werte enthält, wird diesen Objekten dasselbe Objekt aus dem verknüpften Layer zugewiesen.
- Wenn das Verknüpfungsfeld doppelte übereinstimmende Werte enthält, wird nur das erste abgefragte Objekt ausgewählt.

Bemerkung

Die Verknüpfungen in QGIS basieren auf einem Einzelfeldabgleich, so dass Sie in den meisten Fällen sicherstellen möchten, dass die Werte in den abgleichbaren Feldern eindeutig sind.

QGIS bietet einige weitere Optionen, um die Verknüpfung zu optimieren:

- *Verknüpfung im Speicher cachen*: ermöglicht das Zwischenspeichern von Werten im Speicher (ohne Geometrien) aus dem verbundenen Layer, um die Suchvorgänge zu beschleunigen.
- *Attributindex auf Verknüpfungsfeld erzeugen* um Suchvorgänge zu beschleunigen
- *Dynamische Form*: hilft, Verknüpfungsfelder laufend zu synchronisieren, entsprechend dem *Zielfeld*. Auf diese Weise werden auch die Beschränkungen („constrains“) für Verknüpfungsfelder korrekt aktualisiert. Beachten Sie, dass diese Funktion standardmäßig deaktiviert ist, da sie bei vielen Objekten oder zahlreichen Verknüpfungen sehr zeitaufwändig sein kann.
- Wenn der Ziellayer bearbeitbar ist, werden in der Attributtabelle neben den Feldern einige Symbole angezeigt, um über ihren Status zu informieren:
  - : Der Verknüpfungslayer ist nicht als bearbeitbar konfiguriert. Wenn Sie Verknüpfungs-Objekte in der Zielattributtabelle bearbeiten möchten, müssen Sie die Option *Änderbarer verknüpfter Layer* aktivieren.
  - : Der Verknüpfungslayer ist so konfiguriert, dass er bearbeitet werden kann, aber sein aktueller Status ist nur lesbar.
  - : Der Layer ist bearbeitbar, aber die Synchronisierungsmechanismen sind nicht aktiviert. Wenn Sie ein Objekt automatisch in den Join Layer einfügen wollen, wenn ein Objekt im Ziel Layer erstellt wird, dann müssen Sie die Option *Einfügen/Aktualisieren bei Änderung* aktivieren. Umgekehrt kann die Option *Kaskadierendes Löschen* aktiviert werden, wenn Sie Verknüpfungs-Objekte automatisch löschen wollen.
- *Verknüpfte Felder*: Anstatt alle Felder des verknüpften Layers hinzuzufügen, können Sie eine Teilmenge angeben.
- *Benutzerfeldnamenpräfix* für verbundene Felder, um Namenskollisionen zu vermeiden

12.4.2 Festlegung von Beziehungen zwischen mehreren Layern

Im Gegensatz zu Verknüpfungen, die eine Eins-zu-Eins-Verbindung zwischen Objekten auf zwei Layern definieren, können Sie mit Beziehungen Verbindungen zwischen mehreren Objekten auf zwei oder mehr Layern herstellen. Beziehungen sind Einstellungen auf Projektebene und werden unter *Projekt* ► *Eigenschaften…* auf dem Reiter *Beziehungen* festgelegt. Von dort aus können Sie:

- *Beziehung hinzufügen* deren Typ sein kann:
  - *eins zu viele (1:n)*
  - *viele zu viele (n:m)*
  - *polymorphe Beziehung* die Sie mit den entsprechenden Werkzeugen im Dropdown-Menü der Aktion hinzufügen oder bearbeiten können.

Bemerkung

Es gibt noch keine einfache Möglichkeit, eine nicht polymorphe Beziehung zu bearbeiten, nachdem sie erstellt wurde. Nur der Name kann mit einem Doppelklick bearbeitet werden. Für alle anderen Parameter einer solchen Beziehung müssen Sie diese entfernen und neu erstellen.

- *Beziehungen laden*: QGIS ist in der Lage, bestehende Beziehungen aus unterstützten Datenbankformaten (PostgreSQL, GeoPackage, ESRI File Geodatabase, …) zu ermitteln. Dies kann eine gute Möglichkeit sein, die Definition von Beziehungen zu erleichtern.
- *Beziehung löschen*

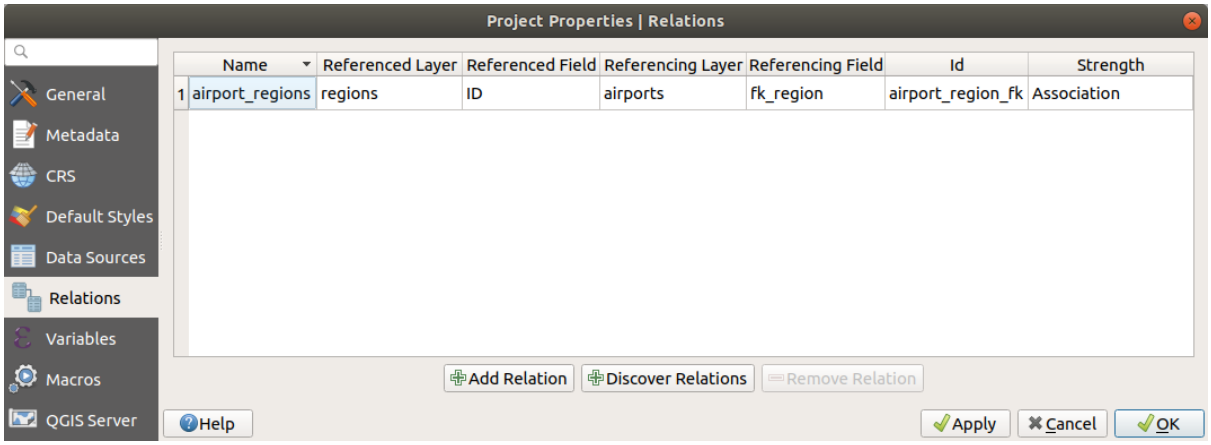


Abb. 12.105: Der Beziehungen Reiter

Eins-zu-viele-Beziehungen (1:n)

Als Beispiel haben Sie einen Layer mit allen Regionen Alaskas (Polygon), der einige Attribute über seinen Namen und Regionstyp sowie eine eindeutige ID (die als Primärschlüssel dient) enthält.

Dann erhalten Sie einen weiteren Layer oder eine Tabelle mit Informationen über Flughäfen, die sich in den Regionen befinden und die Sie ebenfalls verfolgen möchten. Wenn Sie diese zum Layer „Regionen“ hinzufügen möchten, müssen Sie eine Eins-zu-viele-Beziehung mit Fremdschlüsseln erstellen, da es in den meisten Regionen mehrere Flughäfen gibt.

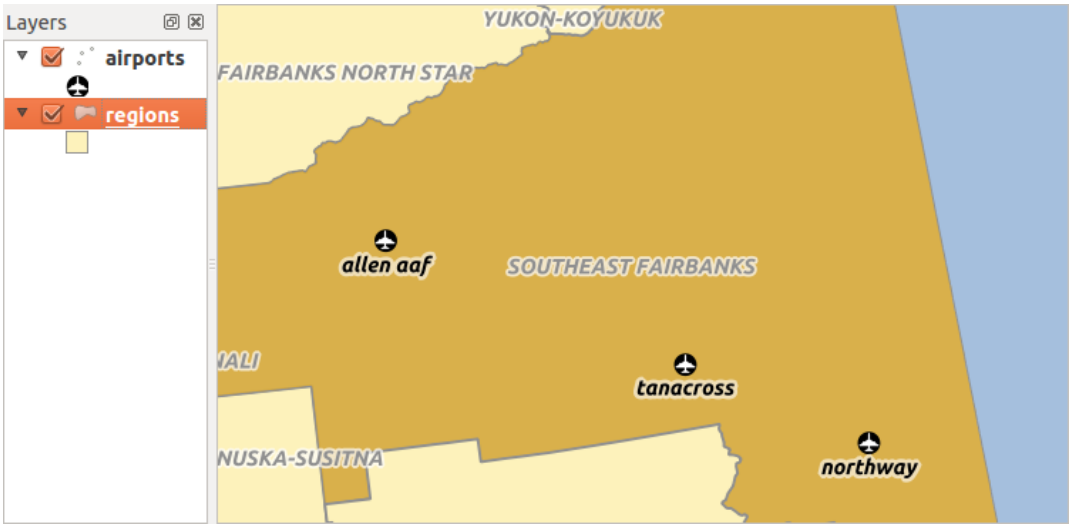


Abb. 12.106: Die Region Alaska mit Flughäfen

Layer und Schlüssel

QGIS macht keinen Unterschied zwischen einer Tabelle und einem Vektorlayer. Grundsätzlich ist ein Vektorlayer eine Tabelle mit einer Geometrie. Sie können also Ihre Tabelle als Vektorlayer hinzufügen. Um die 1:n-Beziehung zu demonstrieren, können Sie die Layer `regions` und `airports` aus den Beispieldatensatz laden. In der Praxis gehört jeder Flughafen zu genau einer Region, während jede Region eine beliebige Anzahl von Flughäfen haben kann (eine typische 1:n-Beziehung).

Zusätzlich zu den Attributen, die die Flughäfen beschreiben, hat der Layer aiports ein weiteres Feld `fk_region`, das als Fremdschlüssel fungiert (wenn Sie eine Datenbank haben, werden Sie wahrscheinlich eine Einschränkung für dieses Feld definieren wollen). Dieses `fk_region`-Feld enthält immer eine ID einer Region. Es kann als Zeiger auf die Region gesehen werden, zu der es gehört.

Alles, was Sie tun müssen, ist, QGIS die Beziehung zwischen den Layern mitzuteilen, damit Sie ein benutzerdefiniertes Formular für die Bearbeitung entwerfen können, und QGIS kümmert sich um die Einrichtung. Es ist auch möglich, verschiedenen Datenquellen in Beziehung zu setzen, Sie können also auch z.B. ein Shapefile und eine CSV-Datei verwenden.

Definieren von 1:n-Beziehungen

Als erstes müssen wir QGIS über die Beziehungen zwischen den Layern informieren. Dies geschieht unter *Projekt* ► *Eigenschaften…*. Öffnen Sie den Reiter *Beziehungen* und klicken Sie auf *Beziehung hinzufügen*.

- **Name** wird als Titel verwendet. Es sollte eine vom Menschen lesbare Zeichenfolge sein, die beschreibt, wofür die Beziehung verwendet wird. In diesem Fall nennen wir sie einfach **airport_relation**.
- **Referenziert (Elter)**, auch als Eltern-Layer bezeichnet, ist der Layer mit dem Primärschlüssel, auf den verwiesen wird, hier also der Layer `regions`. Sie müssen den Primärschlüssel des referenzierten Layers definieren, in diesem Fall ist es `ID`.
- Der **Referenzierend (Kind)**, der auch als Kind-Layer bezeichnet wird, ist der Layer, auf dem sich das Fremdschlüsselfeld befindet. In unserem Fall ist dies der Layer `airports`. Für diesen Layer müssen Sie ein referenzierendes Feld hinzufügen, das auf den anderen Layer verweist, also `fk_region`.

> **Bemerkung**
>
> Manchmal benötigen Sie mehr als ein einzelnes Feld, um Objekte in einem Layer eindeutig zu identifizieren. Das Erstellen einer Beziehung mit einem solchen Layer erfordert einen **zusammengesetzten Schlüssel**, d. h. mehr als ein einzelnes Paar übereinstimmender Felder. Verwenden Sie die Schaltfläche Neues Feldpaar als Teil eines zusammengesetzten Fremdschlüssels hinzufügen, um so viele Paare wie nötig hinzuzufügen.

- **Id** wird für interne Zwecke verwendet und muss eindeutig sein. Sie benötigen sie möglicherweise, um *Benutzerdefinierte Formulare* zu erstellen. Wenn Sie sie leer lassen, wird eine für Sie generiert, aber Sie können selbst eine zuweisen, um eine einfacher zu handhabende zu erhalten
- **Beziehungsstärke** legt die Stärke der Beziehung zwischen dem übergeordneten und dem untergeordneten Layer fest. Der Standardtyp *Assoziation* bedeutet, dass der übergeordnete Layer *einfach* mit dem untergeordneten Layer verknüpft ist, während der Typ *Zusammenstellung* es ermöglicht, beim Duplizieren des übergeordneten Layers auch die untergeordneten Objekte zu duplizieren und beim Löschen eines Objekts auch die untergeordneten Objekte zu löschen, was zu einer Kaskade über alle Ebenen führt (d.h. Kinder von Kindern von… werden ebenfalls gelöscht).

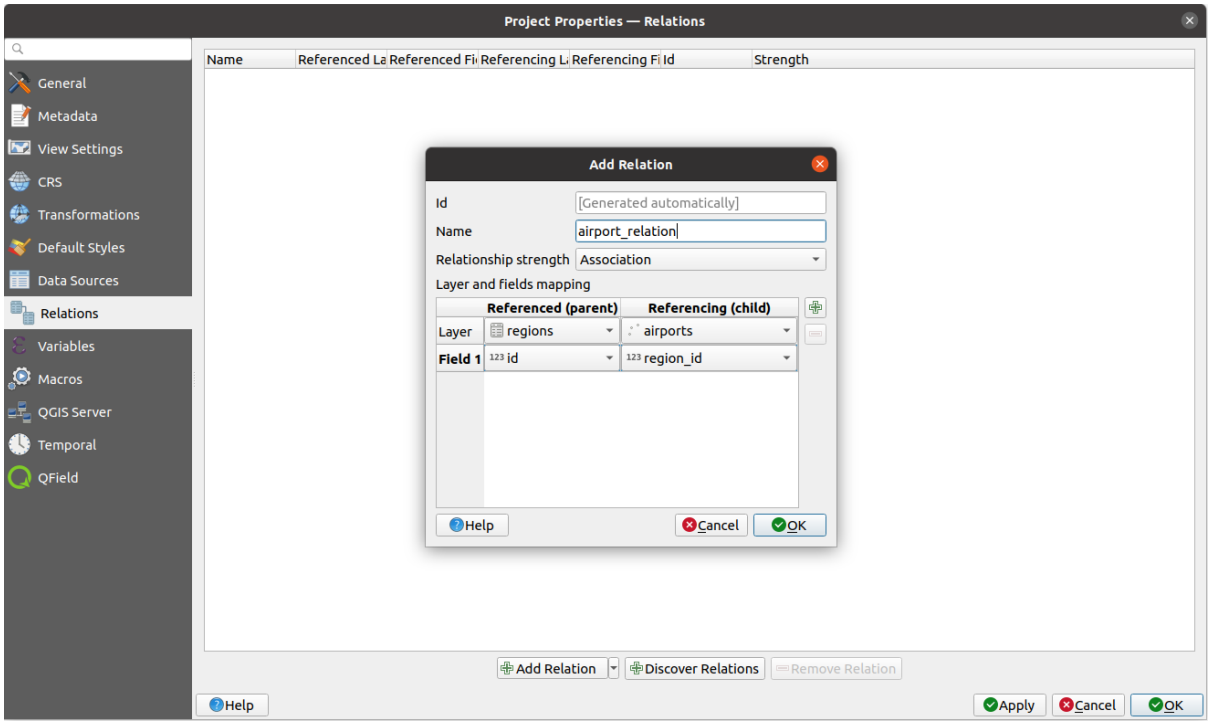


Abb. 12.107: Hinzufügen einer Beziehung zwischen den Layern regions und airports

Auf der Registerkarte *Beziehungen* können Sie auch auf die Schaltfläche *Beziehungen laden* klicken, um die verfügbaren Beziehungen von den Providern der geladenen Layer abzurufen. Dies ist möglich für Layer, die in Datenprovidern wie PostgreSQL oder SpatiaLite gespeichert sind.

Formulare für 1:n-Beziehungen

Da QGIS nun über die Beziehung Bescheid weiß, werden in den entsprechenden Dialogen weitere Informationen angezeigt. Wenn z.B. die Standardmethode zur Erstellung des Objektformulars (automatisch generiert) nicht geändert wurde, wird einfach ein neuer Bereich in diesem Formular einfügt. Wählen Sie also den Layer regions in der Legende aus und verwenden das Objekte-abfragen-Werkzeug, um auf eine Region zu klicken, die mehrere Flughäfen enthält. Je nach Ihren Einstellungen wird das Formular direkt geöffnet oder Sie müssen es im Identifizierungsdialog unter Aktionen auswählen.

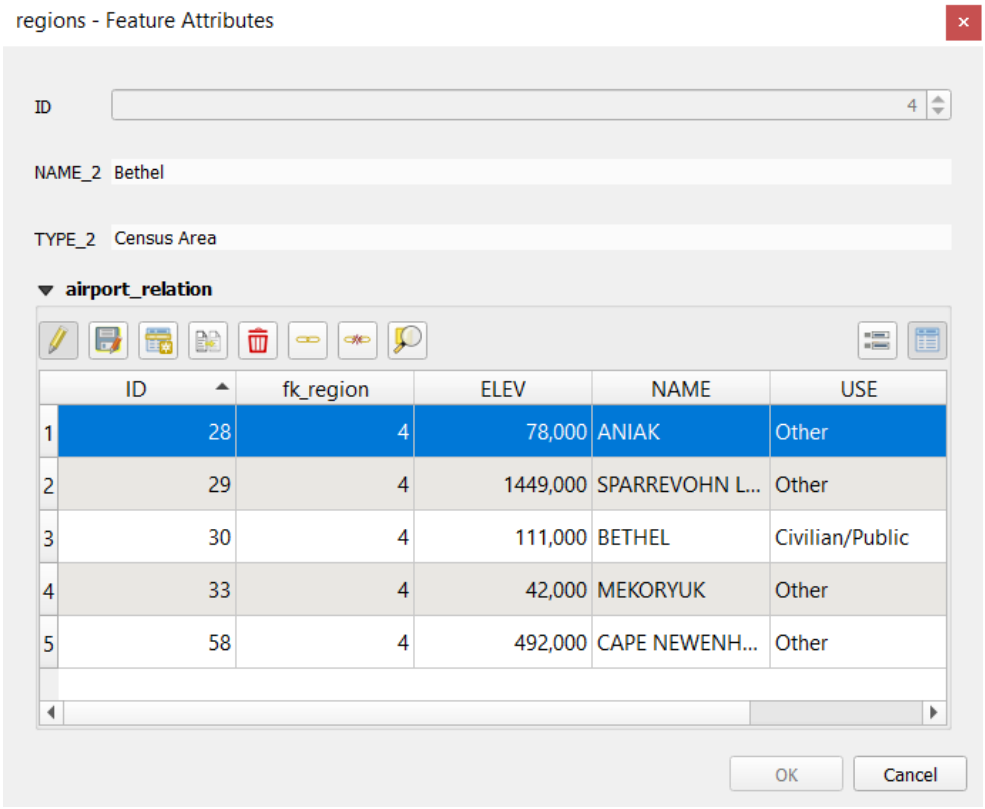


Abb. 12.108: Identifikationsergebnis-Dialog von einem Objekt aus dem Layer regions mit Beziehung zu airports

Wie Sie sehen können, werden die Flughäfen, die dieser Region zugeordnet sind, in einer Tabelle angezeigt. Außerdem gibt es einige Schaltflächen. Schauen wir sie uns kurz an:

- Mit dem Knopf kann in den Bearbeitungsmodus umgeschaltet werden. Seien Sie sich bewusst, dass der Bearbeitungsmodus des ‚airports' Layers umschaltet wird, obwohl wir uns im Objektformular eines Objektes aus dem ‚regions' Layer befinden. Die Tabelle jedoch stellt Objekte des ‚aiports' Layer dar.
- Mit der Schaltfläche werden alle Bearbeitungen im untergeordneten Layer (airports) gespeichert.
- Mit der Schaltfläche können Sie die Flughafengeometrie in der Kartenansicht digitalisieren und das neue Objekt standardmäßig der aktuellen Region zuordnen. Beachten Sie, dass sich das Symbol je nach Geometrietyp ändert.
- Die Schaltfläche fügt einen neuen Datensatz zur Attributtabelle des airports-Layers hinzu und weist das neue Objekt standardmäßig der aktuellen Region zu. Die Geometrie kann später mit dem Digitalisierwerkzeug *Teil hinzufügen* gezeichnet werden.
- Mit der Schaltfläche können Sie ein oder mehrere untergeordnete Objekte innerhalb des untergeordneten Layers kopieren und einfügen. Sie können später einem anderen übergeordneten Objekt zugewiesen oder ihre Attribute geändert werden.
- Die Schaltfläche löscht den/die ausgewählten Flughafen endgültig.
- Das Symbol öffnet ein neues Dialogfeld, in dem Sie einen beliebigen bestehenden Flughafen auswählen können, der dann der aktuellen Region zugewiesen wird. Dies kann praktisch sein, wenn Sie den Flughafen versehentlich in der falschen Region angelegt haben.
- Das Symbol hebt die Verknüpfung des/der ausgewählten Flughafens/Flughäfen mit der aktuellen Region auf, so dass sie nicht mehr zugewiesen sind (der Fremdschlüssel wird auf NULL gesetzt).

- Mit der Schaltfläche können Sie die Karte auf die ausgewählten untergeordneten Objekte zoomen.
- Die beiden Schaltflächen und auf der rechten Seite schalten zwischen der *Tabellenansicht und der Formularansicht* der entsprechenden untergeordneten Objekte um.

Wenn Sie den Drag and Drop Designer 1 für das Objekt Regionen verwenden, können Sie auswählen, welche Werkzeuge verfügbar sind. Sie können sogar entscheiden, ob ein neues Formular geöffnet werden soll, wenn ein neues Objekt hinzugefügt wird, indem Sie die Option *Formular beim Hinzufügen von Objekts verbergen* verwenden. Seien Sie sich jedoch bewusst, dass diese Option voraussetzt, dass nicht-Null-Attribute automatisch ein gültiger Standardwert zugewiesen wird, damit sie den Vorgaben entsprechen.

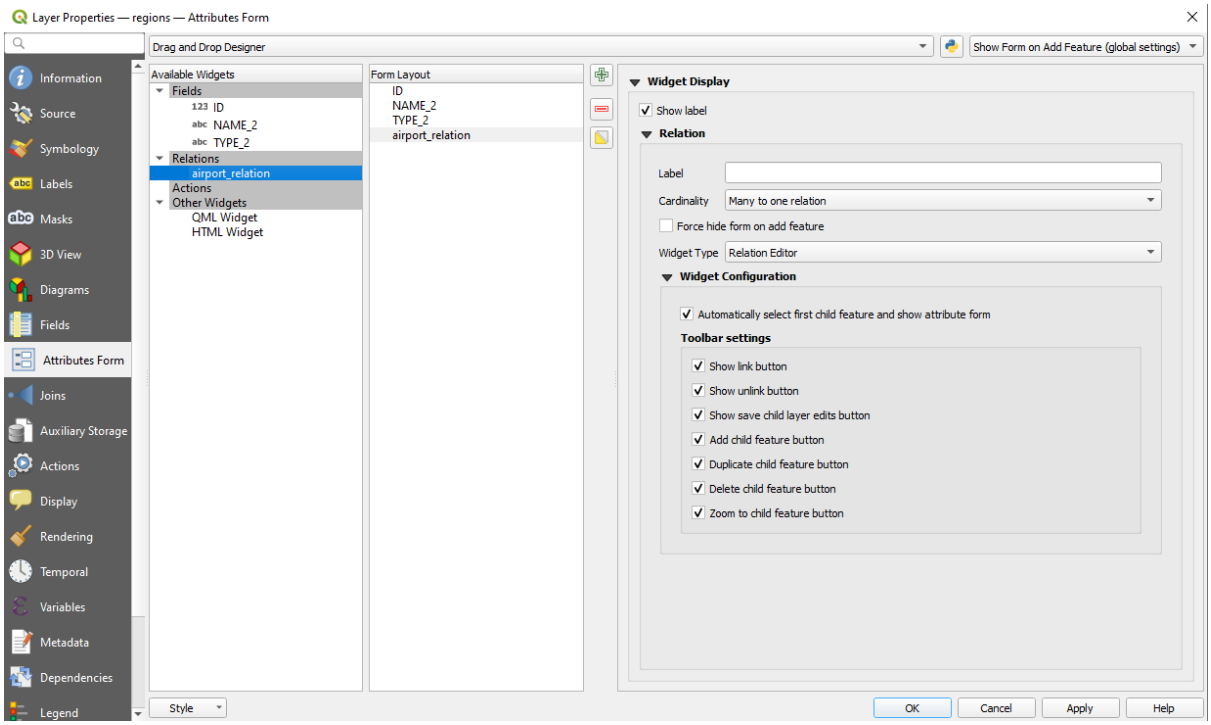


Abb. 12.109: Drag and Drop Designer für die Konfiguration der regions-airports-Beziehung

Im obigen Beispiel hat der referenzierende Layer Geometrien (es handelt sich also nicht nur um eine Tabelle), sodass die obigen Schritte einen Eintrag in der Layer-Attribut-Tabelle erzeugen, der kein entsprechendes geometrisches Objekt hat. So fügen Sie die Geometrie hinzu:

1. Wählen Sie *Attributtabelle öffnen* für den referenzierenden Layer.
2. Wählen Sie den Datensatz aus, der zuvor im Objektformular des referenzierten Layers hinzugefügt wurde.
3. Verwenden Sie das Digitalisierungswerkzeug Teil hinzufügen, um eine Geometrie an den ausgewählten Datensatz der Attributtabelle anzuhängen.

Wenn Sie mit der Flughafentabelle arbeiten, wird das Widget Relation Reference automatisch für das Feld `fk_region` (das Feld, mit dem die Relation erstellt wurde) eingerichtet, siehe *Relation Reference widget*.

Im Formular des airports-Layers sehen Sie rechts neben dem Feld `fk_region` die Schaltfläche : Wenn Sie auf die Schaltfläche klicken, wird das Formular des regions-Layers geöffnet. Mit diesem Widget können Sie einfach und schnell die Formulare der verknüpften übergeordneten Objekte öffnen.

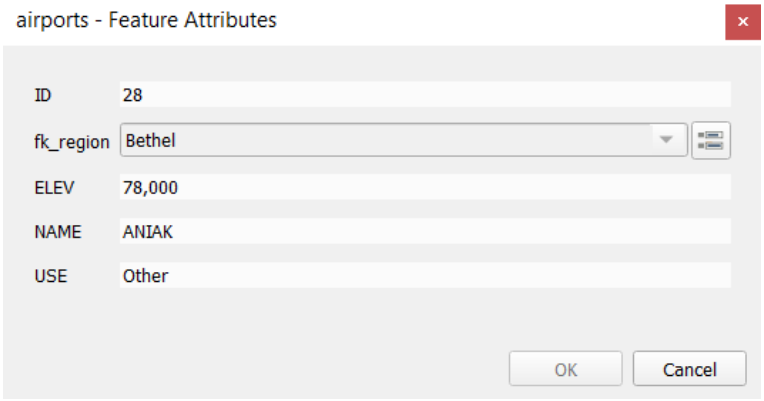
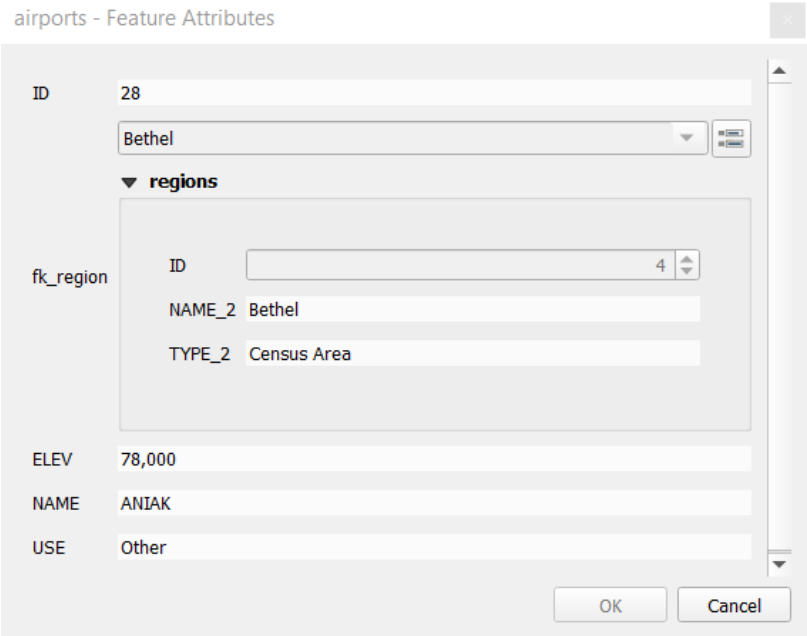


Abb. 12.110: Objekte abfragen Dialog von ‚airport' mit Beziehung zu ‚regions'

Das Relation Reference Widget hat auch eine Option, um das Formular des übergeordneten Layers in den untergeordneten Layer einzubetten. Sie ist im Menü *Eigenschaften* ► *Attributformular* des airports-Layers verfügbar: Wählen Sie das Feld `fk_region` und aktivieren Sie die Option `Eingebettetes Formular anzeigen`.

Wenn Sie sich jetzt den Objektdialog ansehen, werden Sie sehen, dass das Formular für die Region in das Formular für die Flughäfen eingebettet ist und sogar eine Combobox enthält, mit der Sie den aktuellen Flughafen einer anderen Region zuweisen können.



Wenn Sie den Editiermodus des airports-Layers umschalten, hat das Feld `fk_region` außerdem eine Autovervollständigungsfunktion: während der Eingabe sehen Sie alle Werte des Feldes `id` des regions-Layers.

Hier ist es möglich, ein Polygon für den regions-Layer mit dem Knopf zu digitalisieren, wenn Sie die Option `Hinzufügen neuer Objekte erlauben` im *Eigenschaften* ► *Attributformular* Menü des airports-Layers gewählt haben.

Der untergeordnete Layer kann auch im Werkzeug *Objekte nach Wert auswählen* verwendet werden, um Objekte des übergeordneten Layers anhand der Attribute der untergeordneten Layer auszuwählen.

In Abb. 12.111 werden alle Regionen ausgewählt, in denen die mittlere Höhe der Flughäfen mehr als 500 Meter über dem Meeresspiegel beträgt.

Sie werden feststellen, dass im Formular viele verschiedene Aggregationsfunktionen zur Verfügung stehen.

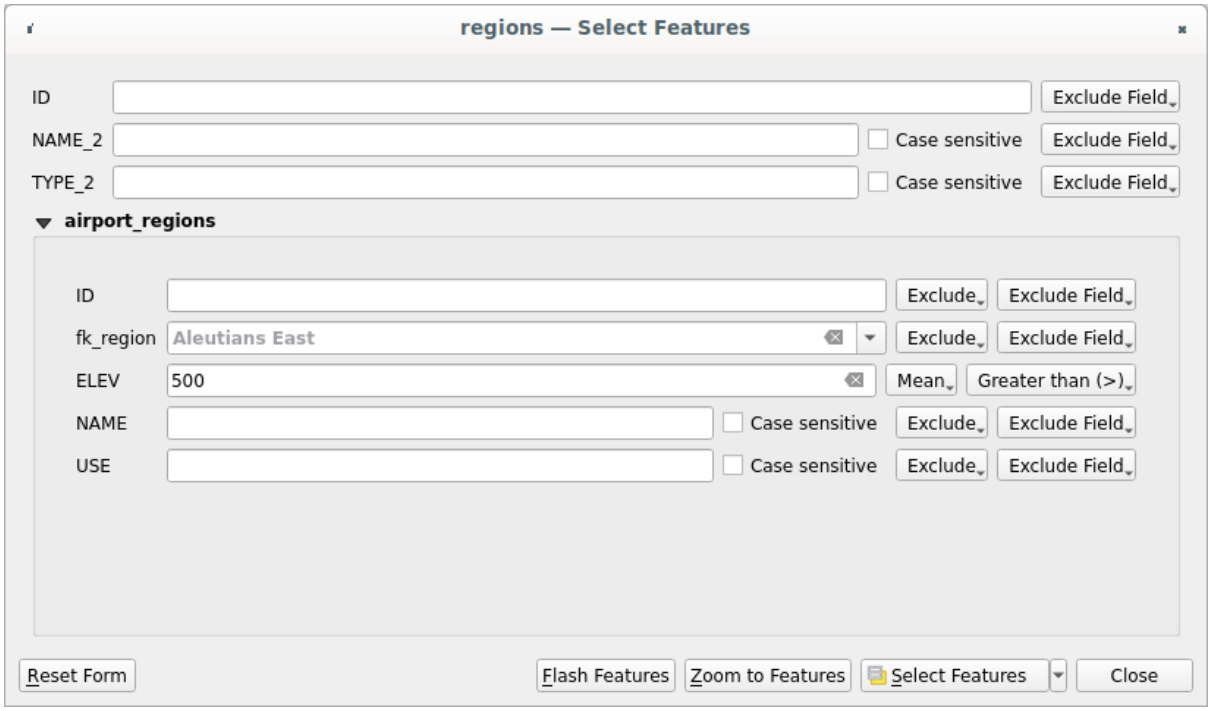


Abb. 12.111: Übergeordnete Objekte mit den Werten der untergeordneten Objekte auswählen

Viele-zu-viele-Beziehungen (n:m)

n:m-Beziehungen sind viele-zu-viele-Beziehungen zwischen zwei Tabellen. Zum Beispiel die Layer `airports` und `airlines`: an ein Flughafen landen mehrere Fluggesellschaften und eine Fluggesellschaft fliegt mehrere Flughäfen an.

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)
);
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
CREATE INDEX airlines_geom_idx ON locations.airlines USING gist (geom);

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
 );
```

Anstelle von PostgreSQL können Sie auch GeoPackage verwenden. In diesem Fall können die drei Tabellen manuell über die *Datenbank* ► *DB Verwaltung…* erstellt werden. In GeoPackage gibt es keine Schemata, sodass das Präfix *locations* nicht benötigt wird.

Fremdschlüssel-Beschränkungen (constrains) in der Tabelle `airports_airlines` können nicht mit *Tabelle* ► *Tabelle erstellen…* oder *Tabelle* ► *Tabelle bearbeiten…* erstellt werden, daher sollten sie über *Datenbank* ► *SQL-Fenster…* erstellt werden. GeoPackage unterstützt keine *ADD CONSTRAINT* Anweisungen, daher sollte die Tabelle `airports_airlines` in zwei Schritten erstellt werden:

1. Richten Sie die Tabelle nur mit dem Feld `id` ein mit *Tabelle* ► *Tabelle erstellen…*
2. Geben Sie mit *Datenbank* ► *SQL-Fenster…* diesen SQL-Code ein und führen Sie ihn aus:

```
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;
```

Dann sollten Sie in QGIS zwei *eins-zu-viele-Beziehungen* einrichten, wie oben erklärt:

- eine Beziehung zwischen der Tabelle `airlines` und der Pivot-Tabelle;
- und eine zweite zwischen der Tabelle `airports` und der Pivot-Tabelle.

Eine einfachere Methode (nur für PostgreSQL) ist die Verwendung des *Discover Beziehungen laden* in *Projekt* ► *Eigenschaften* ► *Beziehungen*. QGIS wird automatisch alle Beziehungen in Ihrer Datenbank lesen und Sie müssen nur die zwei auswählen, die Sie benötigen. Denken Sie daran, die drei Tabellen zuerst in das QGIS-Projekt zu laden.

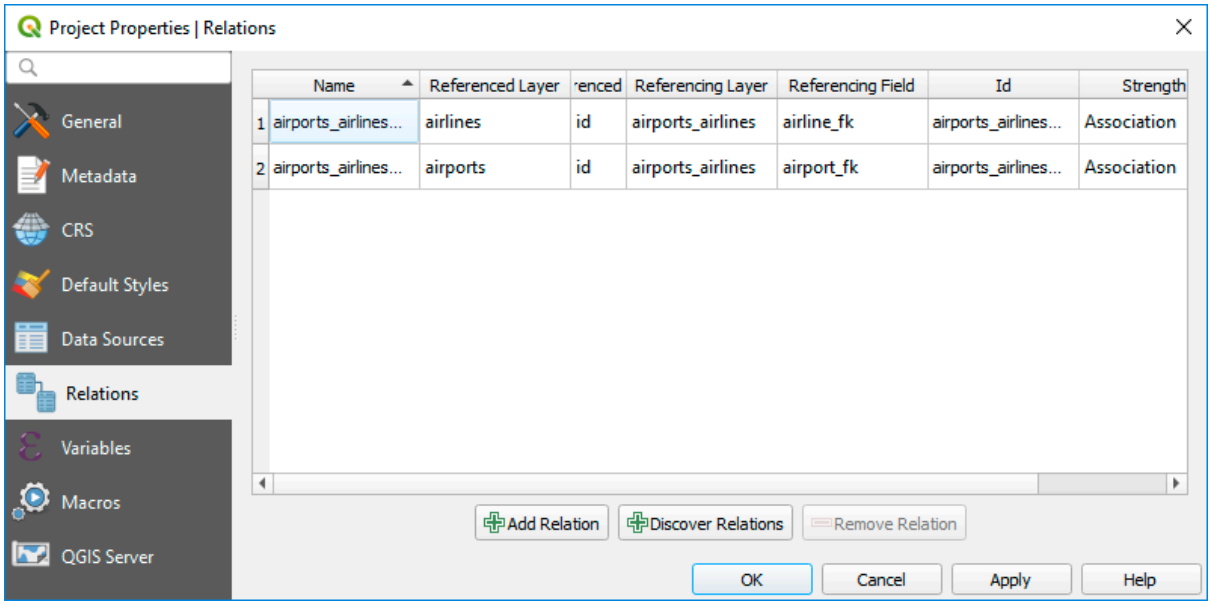


Abb. 12.112: Beziehungen laden

Falls Sie einen `airport` oder eine `airline` entfernen wollen, wird QGIS den/die zugehörigen Datensatz/-sätze in der Tabelle `airports_airlines` nicht entfernen. Diese Aufgabe wird von der Datenbank übernommen, wenn man die richtigen *Einschränkungen* bei der Erstellung der Pivot-Tabelle angibt, wie in diesem Beispiel.

Bemerkung

Kombinierte n:m-Beziehung mit automatischer Transaktionsgruppe

Sie sollten den Transaktionsmodus in *Projekteigenschaften* ► *Datenquellen* ► aktivieren, wenn Sie in einem solchen Kontext arbeiten. QGIS sollte in der Lage sein, Zeilen in allen Tabellen (Fluggesellschaften, Flughäfen und den Pivot-Tabellen) hinzuzufügen oder zu aktualisieren.

Schließlich müssen wir unter *Layereigenschaften* ► *Attributformular* für die Layer `airports` und `airlines` die richtige Kardinalität auswählen. Für den ersten Layer sollten wir die Option **airlines (id)** und für den zweiten die Option **airports (id)** wählen.

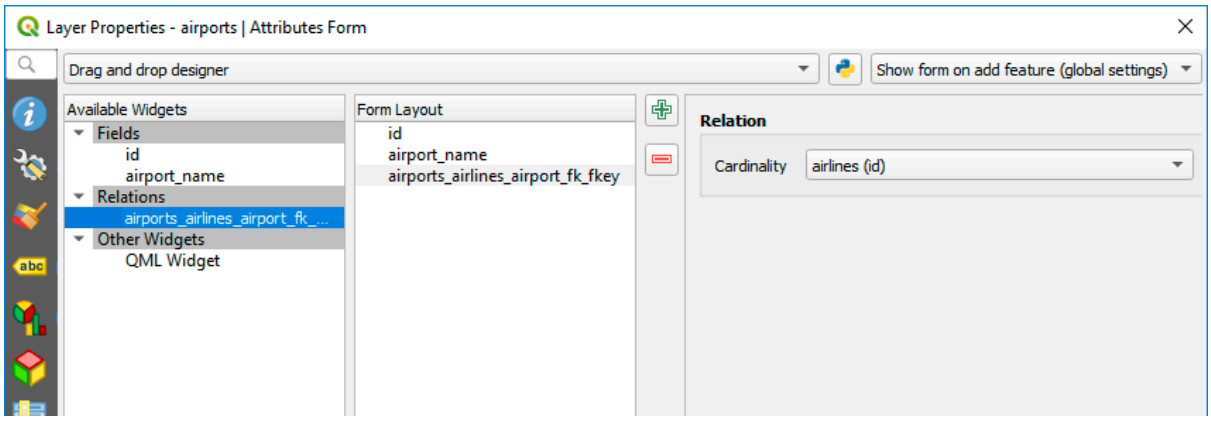


Abb. 12.113: Kardinalität der Beziehung festlegen

Jetzt können Sie einen Flughafen mit einer Fluggesellschaft (oder eine Fluggesellschaft mit einem Flughafen) verknüpfen, indem Sie *Kind-Objekt hinzufügen* oder *Vorhandenes Kind-Objekt verknüpfen* in den Unterformularen verwenden. Ein Datensatz wird automatisch in die Tabelle `airports_airlines` eingefügt.

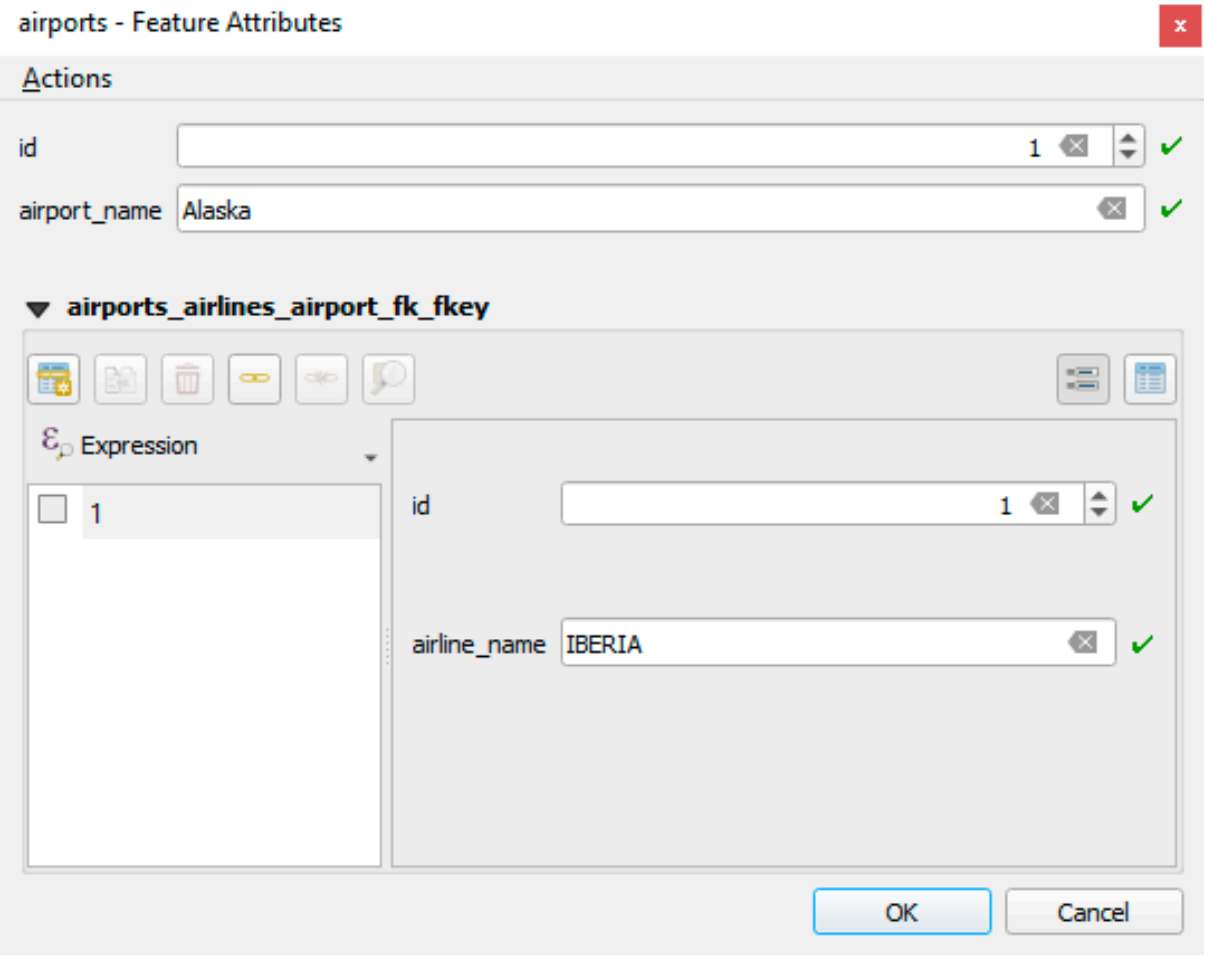


Abb. 12.114: n:m-Beziehung zwischen airports und airlines

Bemerkung

Verwendung der Kardinalität **Viele-zu-eins-Beziehung**

Manchmal ist es nicht wünschenswert, die Pivot-Tabelle in einer n:m-Beziehung auszublenden. Hauptsächlich deshalb, weil es in der Beziehung Attribute gibt, die nur dann Werte haben können, wenn eine Beziehung hergestellt ist. Wenn Ihre Tabellen ein Geometriefeld haben, könnte es interessant sein, die Option *Auf der Karte auswählen* zu aktivieren (*Layereigenschaften* ► *Attributformular* ► *Verfügbare Widgets* ► *Felder*) für die Fremdschlüsselfelder in der Pivot-Tabelle.

Bemerkung

Pivot-Tabelle Primärschlüssel

Vermeiden Sie die Verwendung mehrerer Felder im Primärschlüssel einer Pivot-Tabelle. QGIS geht von einem einzigen Primärschlüssel aus, so dass eine Einschränkung wie `constraint airports_airlines_pkey primary key (airport_fk, airline_fk)` nicht funktionieren wird.

Polymorphe Beziehungen

Der Zweck

Polymorphe Beziehungen sind ein Spezialfall von 1:n-Beziehungen, bei denen ein einziger referenzierender Layer die Objekte für mehrere referenzierte Layer enthält. Dies unterscheidet sich von normalen Beziehungen, die für jeden referenzierten Layer einen anderen referenzierenden Layer erfordern. Ein einziger referenzierender Layer wird durch Hinzufügen einer zusätzlichen Spalte `layer_field` im referenzierenden Layer erreicht, die Informationen zur Identifizierung des referenzierten Layers enthält. In seiner einfachsten Form fügt der referenzierende Layer einfach den Layernamen des referenzierten Layers in dieses Feld ein.

Genauer gesagt ist eine polymorphe Beziehung eine Reihe normaler Beziehungen mit demselben referenzierenden Layer, wobei der referenzierte Layer dynamisch definiert ist. Die polymorphe Einstellung des Layers wird durch die Verwendung eines Ausdrucks gelöst, der mit einigen Eigenschaften des referenzierten Layers übereinstimmen muss, z. B. Tabellenname, Layer-ID, Layer-Name.

Stellen Sie sich vor, wir gehen in den Park und wollen Bilder von verschiedenen Arten von Pflanzen `plants` und Tieren `animals` machen, die wir dort sehen. Jeder Pflanze oder jedem Tier sind mehrere Bilder zugeordnet. Wenn wir also die normalen 1:n-Beziehung verwenden, um Bilder zu speichern, brauchen wir zwei separate Tabellen, `animal_images` und `plant_images`. Das mag für zwei Tabellen kein Problem sein, aber stellen Sie sich vor, wir wollen separate Bilder für Pilze, Vögel usw. aufnehmen.

Polymorphe Beziehungen lösen dieses Problem, da alle referenzierenden Objekte in derselben Tabelle „documents“ gespeichert werden. Für jedes Objekt wird der referenzierte Layer im Feld „referenced_layer“ und die referenzierte Objekt-ID im Feld „referenced_id“ gespeichert.

Eine polymorphe Beziehung definieren

Zunächst müssen Sie QGIS die polymorphen Beziehungen zwischen den Layern mitteilen. Dies geschieht unter *Projekt* ► *Eigenschaften*…. Öffnen Sie die Registerkarte *Beziehungen* und klicken Sie auf den kleinen Abwärtspfeil neben der Schaltfläche *Beziehung hinzufügen*, sodass Sie die Option *Polymorphe Beziehung hinzufügen* aus dem neu erscheinenden Dropdown auswählen können.

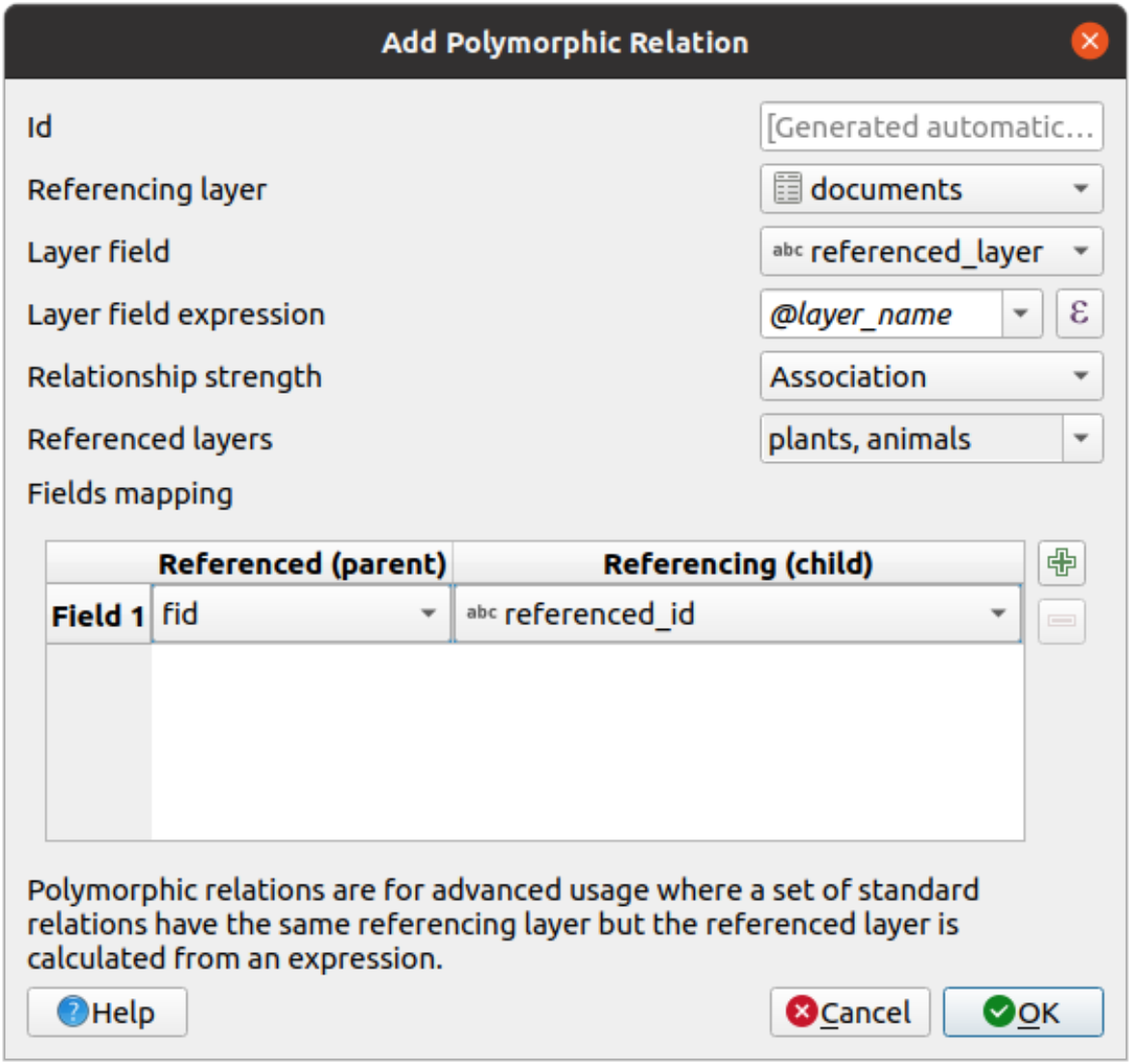


Abb. 12.115: Hinzufügen einer polymorphen Beziehung mit dem Layer `documents` als Referenz und den Layern `animals` und `plants` als referenzierte Layer.

- **Id** wird für interne Zwecke verwendet und muss eindeutig sein. Sie benötigen sie möglicherweise, um *Benutzerdefinierte Formulare* zu erstellen. Wenn Sie sie leer lassen, wird eine für Sie generiert, aber Sie können selbst eine zuweisen, um eine einfacher zu handhabende zu erhalten
- Die **Referenzierungslayer (Child)**, auch als untergeordnete Layer bezeichnet, sind diejenigen, die das Fremdschlüsselfeld enthalten. In unserem Fall ist dies der Layer „documents“ (Dokumente). Für diesen Layer müssen Sie ein Referenzierungsfeld hinzufügen, das auf den anderen Layer verweist, also „referenced_id“ (referenzierte ID).

> **Bemerkung**
>
> Manchmal benötigen Sie mehr als ein einzelnes Feld, um Objekte in einem Layer eindeutig zu identifizieren. Das Erstellen einer Beziehung mit einem solchen Layer erfordert einen **zusammengesetzten Schlüssel**, d. h. mehr als ein einzelnes Paar übereinstimmender Felder. Verwenden Sie die Schaltfläche Neues Feldpaar als Teil eines zusammengesetzten Fremdschlüssels hinzufügen, um so viele Paare wie nötig hinzuzufügen.

- **Layerfeld** ist das Feld in der Referenztabelle, in dem das Ergebnis des ausgewerteten Layer-Ausdrucks gespeichert wird, d. h. die Referenztabelle, zu der dieses Objekt gehört. In unserem Beispiel wäre dies das

Feld `referenced_layer`.

- **Layerfeldausdruck** ergibt einen eindeutigen Bezeichner des Layers. Dies kann der Layer-Name `@layer_name`, die Layer-ID `@layer_id`, der Tabellenname des Layers `decode_uri(@layer, 'table')` oder etwas anderes sein, das einen Layer eindeutig identifiziert.
- **Beziehungsstärke** legt die Stärke der erzeugten Beziehungen zwischen dem übergeordneten und dem untergeordneten Layer fest. Der Standardtyp *Assoziation* bedeutet, dass der übergeordnete Layer *einfach* mit dem untergeordneten Layer verknüpft ist, während der Typ *Zusammenstellung* es ermöglicht, beim Duplizieren des übergeordneten Layers auch die untergeordneten Objekte zu duplizieren und beim Löschen eines Objekts auch die untergeordneten Objekte zu löschen, was zu einer Kaskade über alle Ebenen führt (d.h. Kinder von Kindern von… werden ebenfalls gelöscht).
- **Referenzierte Layer**, die auch als übergeordnete Layer betrachtet werden, sind diejenigen, auf die der Primärschlüssel zeigt, in diesem Fall wären das die Layer `plants` und `animals`. Sie müssen den Primärschlüssel der referenzierten Layer aus dem Dropdown-Menü definieren, hier ist es `fid`. Beachten Sie, dass die Definition eines gültigen Primärschlüssels voraussetzt, dass alle referenzierten Layer ein Feld mit diesem Namen haben. Wenn es kein solches Feld gibt, können Sie keine polymorphe Beziehung speichern.

Einmal hinzugefügt, kann die polymorphe Beziehung über den Menüeintrag *Polymorphe Beziehung bearbeiten* bearbeitet werden.

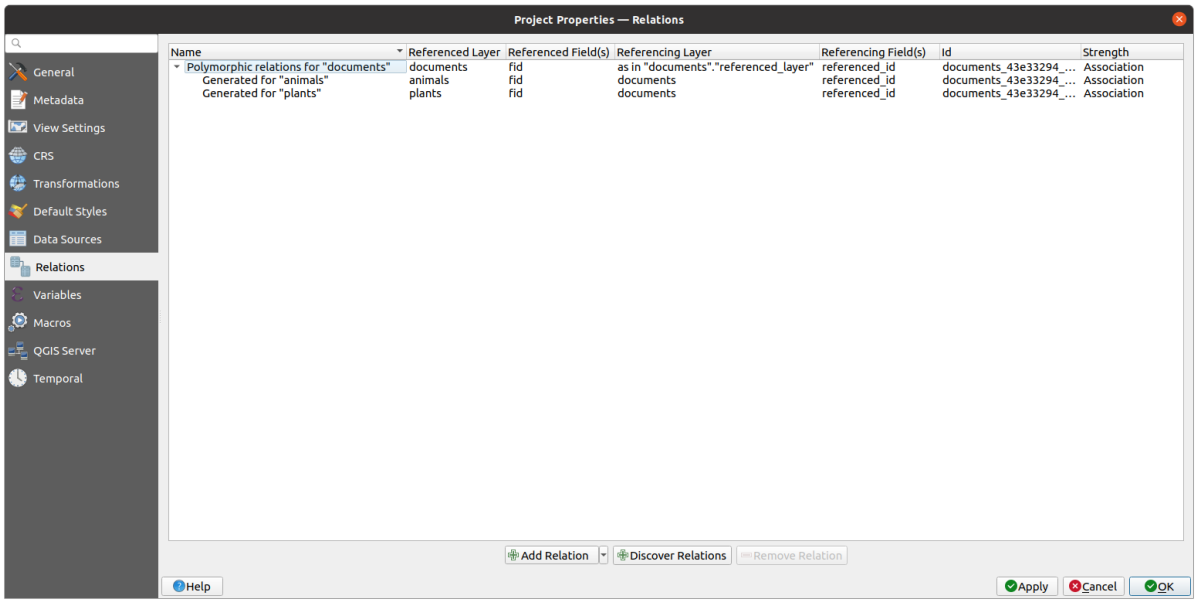


Abb. 12.116: Vorschau auf die neu erstellte polymorphe Beziehung und ihre untergeordneten Beziehungen für Tiere und Pflanzen.

Das oben genannte Beispiel verwendet das folgende Datenbankschema:

```
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
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
(
   fid serial NOT NULL,
   geom geometry(Point, 4326) NOT NULL,
   plant_species text NOT NULL,
   CONSTRAINT plants_pkey PRIMARY KEY (fid)
);

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.

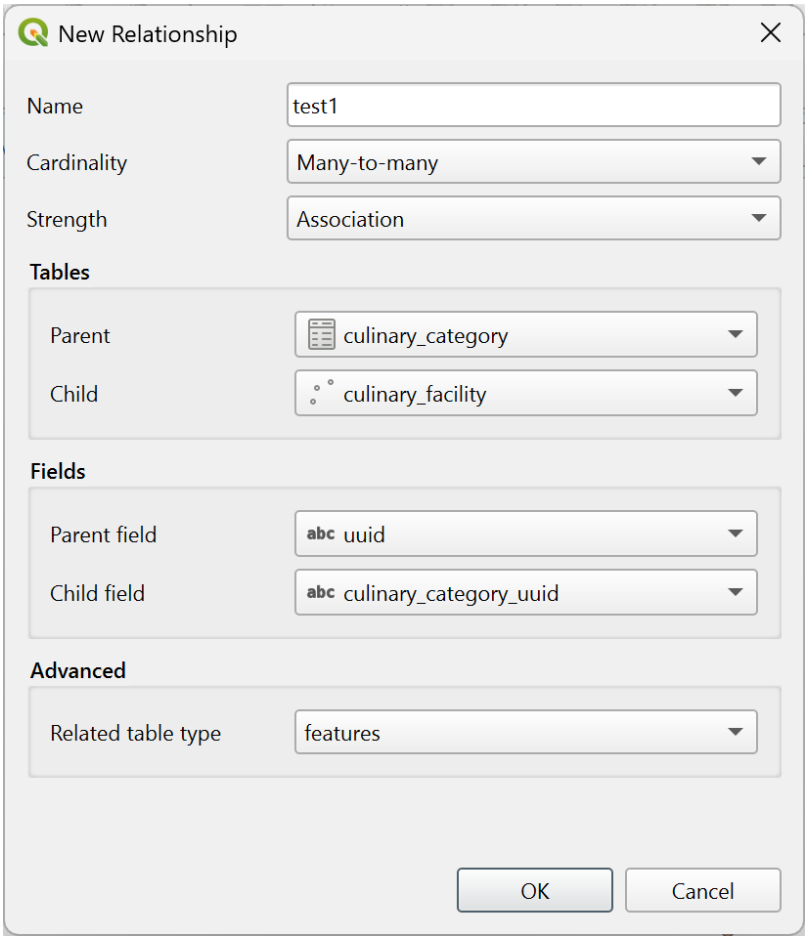


Abb. 12.117: Managing dataset-stored relationships in the Browser Panel

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.

Abb. 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).

Bemerkung

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*.

Bemerkung

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.

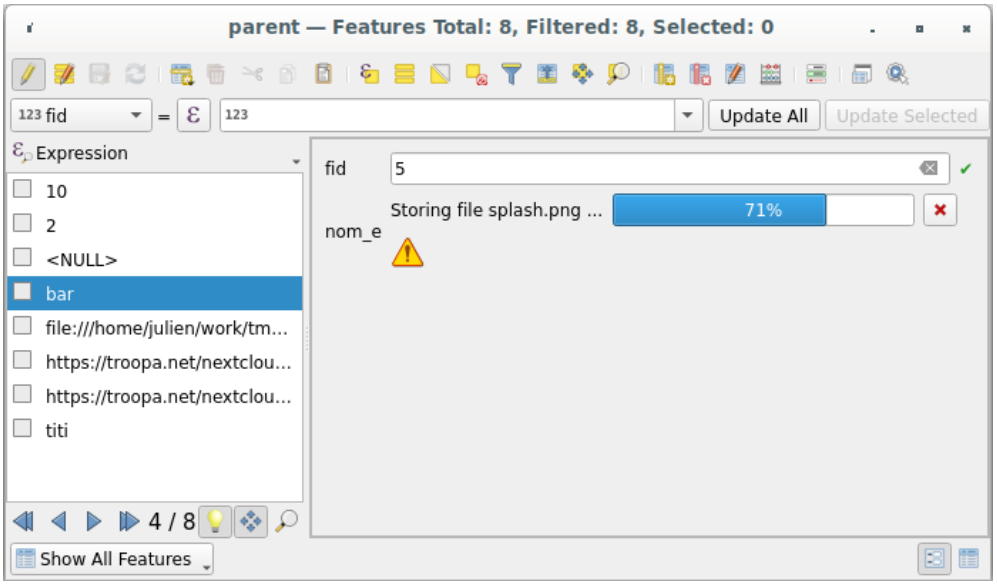


Abb. 12.119: Storing a file to a WebDAV external storage

Bemerkung

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 *Das Protokoll-Bedienfeld*.

KAPITEL 13

Arbeiten mit Rasterdaten

13.1 Der Eigenschaften-Dialog von Rasterlayern

Rasterdaten bestehen aus Pixeln (oder Zellen), und jedes Pixel hat einen Wert. Sie werden in der Regel zum Speichern verschiedener Datentypen verwendet, darunter:

- Bildmaterial, z. B. Satellitenbilder, digitale Luftaufnahmen, gescannte Karten
- Höhendaten, wie digitale Höhenmodelle (DEM), digitale Geländemodelle (DTM)
- Andere Arten von Daten wie Bodenbedeckung, Bodentypen, Niederschlag und viele andere.

Raster data can be stored in several supported formats, including GeoTIFF, ERDAS Imagine, ArcInfo ASCII GRID, PostgreSQL Raster and others. See more at *Öffnen von Daten*.

Um die Eigenschaften eines Raster-Layers anzuzeigen und einzustellen, doppelklicken Sie auf den Namen des Layers in der Kartenlegende oder klicken Sie mit der rechten Maustaste auf den Namen des Layers und wählen Sie *Eigenschaften* aus dem Kontextmenü. Daraufhin wird das Dialogfeld *Layereigenschaften* für Rasterlayer geöffnet.

Es gibt mehrere Reiter in diesem Dialog:

Information	*Quelle*	*Symbology* ([1])
Transparency ([1])	*Labels* ([1])	*Histogram* ([1])
Darstellung	*Zeitlich*	*Pyramiden*
Höhe	*Metadata*	*Legend*
Anzeige	*Attribute Tables*	*QGIS Server*
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.

> **Tipp**
>
> **Live-Update-Darstellung**
>
> Das *Layergestaltung-Bedienfeld* bietet Ihnen einige der wichtigesten Reiter des Layereigenschaften-Dialogs und kann ein gutes Werkzeug sein, um die Konfiguration der Layereigenschaften schnell anzupassen und Ihre Änderungen auf der Kartenansicht anzuzeigen.

> **Bemerkung**
>
> Da die Eigenschaften (Symbolisierung, Beschriftung, Aktionen, …) von eingebetteten Layern (siehe *Layer aus externen Projekten einbinden*) aus der ursprünglichen Projektdatei gezogen werden, und um Änderungen zu vermeiden, die dieses Verhalten unterbrechen könnten, ist der Layer-Eigenschaften-Dialog für diese Layer nicht verfügbar.

13.1.1 Information

Die Registerkarte *Information* ist schreibgeschützt und stellt einen interessanten Ort dar, um schnell zusammengefasste Informationen und Metadaten für den aktuellen Layer zu erhalten. Die bereitgestellten Informationen sind:

- allgemeine Angaben wie Name im Projekt, Quellpfad, Liste der Hilfsdateien, letzte Speicherzeit und -größe, der verwendete Datenanbieter
- Benutzerdefinierte Eigenschaften, die verwendet werden, um zusätzliche Informationen über den Layer im aktiven Projekt zu speichern. Zu den standardmäßigen benutzerdefinierten Eigenschaften gehört *Identifikation/Format*, was Einfluss darauf hat, wie die Ergebnisse der Verwendung des Werkzeugs Objekte abfragen über einen Raster-Layer formatiert werden. Weitere Eigenschaften können mit PyQGIS erstellt und verwaltet werden, insbesondere mit der Methode setCustomProperty().
- abhängig von der Daten-Quelle des Layers: Umfang, Breite und Höhe, Datentyp, GDAL-Treiber, Bandstatistiken
- das Koordinatenbezugssystem: Name, Einheiten, Methode, Genauigkeit, Referenz (d.h., ob es statisch oder dynamisch ist)
- Auslesen der Layereigenschaften: Datentyp, Ausdehnung, Breite/Höhe, Komprimierung, Pixelgröße, Bandenstatistiken, Anzahl der Spalten, Zeilen und No-Data-Werte des Rasters…
- aus den *Metadaten*: Zugriff, Ausdehnung, Links, Kontakte, Protokoll…

13.1.2 Quelle

Auf dem Reiter *Quelle* werden grundlegende Informationen über das ausgewählte Raster angezeigt, darunter:

- der *Layername*, der im *Layer*-Bedienfeld angezeigt wird;
- das *Koordinatenreferenzsystem*: Zeigt das *Koordinatenreferenzsystem (KBS)* des Layers an. Sie können das KBS des Layers ändern, indem Sie ein kürzlich verwendetes KBS aus der Dropdown-Liste auswählen oder auf die Schaltfläche KBS wählen klicken (siehe *Auswahl des Koordinatenbezugssystems*). Verwenden Sie dieses Verfahren nur, wenn das KBS falsch oder nicht angegeben ist. Wenn Sie Ihre Daten neu projizieren möchten, verwenden Sie einen Verarbeitungsalgorithmus zur Reprojektion oder *speichern Sie die Daten in einer neuen Datei*.
- 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.

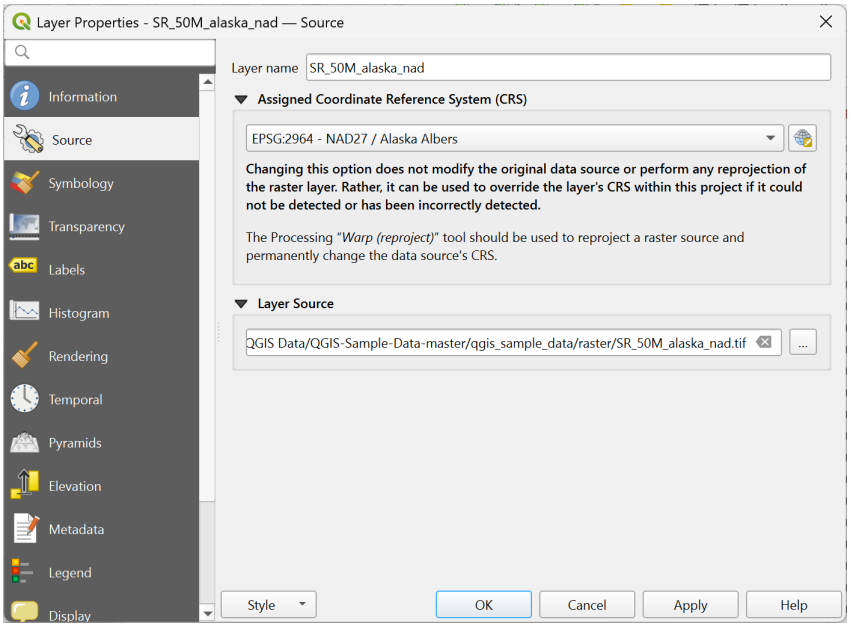


Abb. 13.1: Rasterlayereigenschaften - Quelle-Reiter

13.1.3 Symbolisierung

Der Reiter Symbolisierung für Rasterlayer besteht aus drei verschiedenen Abschnitten:

- Das *Kanaldarstellung*, in dem Sie die zu verwendende Darstellungsart festlegen können
- Die *Layerdarstellung* zur Anwendung von Effekten auf den Layer
- Die *Abtastung* zur Optimierung der Darstellung auf der Karte

Kanaldarstellung

QGIS bietet viele verschiedene *Darstellungsarten*. Die Wahl der Darstellung hängt vom Datentyp und der Information ab, die Sie hervorheben möchten.

1. *Multikanalfarbe* - wenn die Datei mehrere Bänder enthält (z. B. ein Satellitenbild mit mehreren Bändern).
2. *Paletten-/Eindeutige Werte* - für einzelne Banddateien, die mit einer indizierten Palette geliefert werden (z. B. eine digitale topografische Karte), oder für die allgemeine Verwendung von Paletten zum Darstellen von Rasterlayern.
3. *Einkanalgraustufen* - (ein Band) des Bildes wird grau gerendert. QGIS wählt diese Darstellung, wenn die Datei weder mehrfarbig noch palettiert ist (z.B. bei einer schattierten Reliefkarte).
4. *Einkanalpseudofarbe* - diese Darstellung kann für Dateien mit einer kontinuierlichen Palette oder Farbkarte (z. B. eine Höhenkarte) verwendet werden).
5. *Einzelfarbe* - Die Raster-Layer wird mit einer einzigen Farbe gerendert.

6. *Schummerung* - Erzeugt einen Schattenwurf aus einem Band.
7. *Konturen* - Erzeugt on-the-fly Konturen für ein Rasterband.

Multikanalfarbe

Bei der Multikanalfarbe werden drei ausgewählte Bereiche des Bildes als rote, grüne und blaue Komponente des Farbbildes verwendet. QGIS ermittelt automatisch *Min* und *Max* Werte für jedes Band des Rasters und skaliert die Farbgebung entsprechend. Sie können die Wertebereiche im Abschnitt *Min/Max-Werteinstellungen* anpassen.

Eine *Kontrastverbesserung*-Methode kann auf die Werte angewendet werden: ‚Keine Erweiterung', ‚Strecken auf MinMax', ‚Auf MinMax strecken und zuschneiden' und ‚Zuschneiden auf MinMax'.

Bemerkung

Kontrastverbesserung

Beim Hinzufügen von GRASS-Rastern wird die Option *Kontrastverstärkung* immer automatisch auf *Strecken auf Min/Max* gesetzt, auch wenn in den allgemeinen QGIS-Optionen ein anderer Wert eingestellt ist.

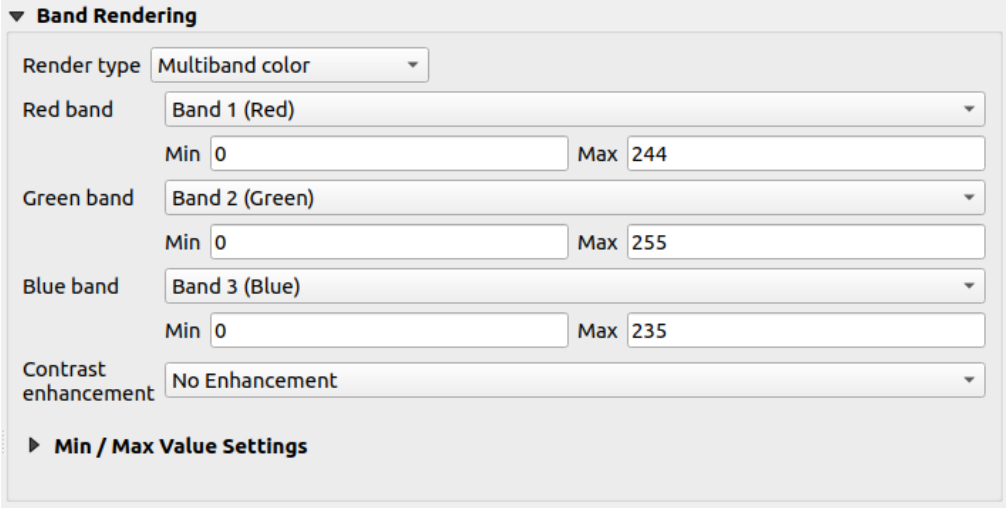


Abb. 13.2: Raster Symbolisierung - Multikanalfarbe-Darstellung

Tipp

Einen einzelnen Kanal eines Mehrkanal-Rasterlayers anzeigen

Wenn Sie ein einzelnes Band eines Multiband-Bildes (z. B. Rot) anzeigen möchten, könnten Sie denken, dass Sie die Bänder Grün und Blau auf *Nicht gesetzt* einstellen würden. Die bevorzugte Vorgehensweise ist jedoch, den Bildtyp auf *Einzelkanalgraustufe* zu setzen und dann Rot als das zu verwendende *Grauband* auszuwählen.

Paletten-/Eindeutige Werte

Dies ist die Standard-Darstellungs-Option für Einzelband-Dateien, die eine Farbtabelle enthalten, in der jedem Pixelwert eine bestimmte Farbe zugewiesen ist. In diesem Fall wird die Palette automatisch gerendert.

Sie kann für alle Arten von Rasterbändern verwendet werden, wobei jedem einzelnen Rasterwert eine Farbe zugewiesen wird.

Wenn Sie eine Farbe ändern möchten, doppelklicken Sie einfach auf die Farbe und das Dialogfeld *Farbe wählen* erscheint.

Es ist auch möglich, den Farben Beschriftungen zuzuweisen. Die Beschriftung wird dann in der Legende des Rasterlayers angezeigt.

Wenn Sie mit der rechten Maustaste auf ausgewählte Zeilen in der Farbtabelle klicken, wird ein Kontextmenü angezeigt:

- *Farbe ändern…* für die Auswahl
- *Deckkraft ändern…* für die Auswahl
- *Beschriftung ändern…* für die Auswahl

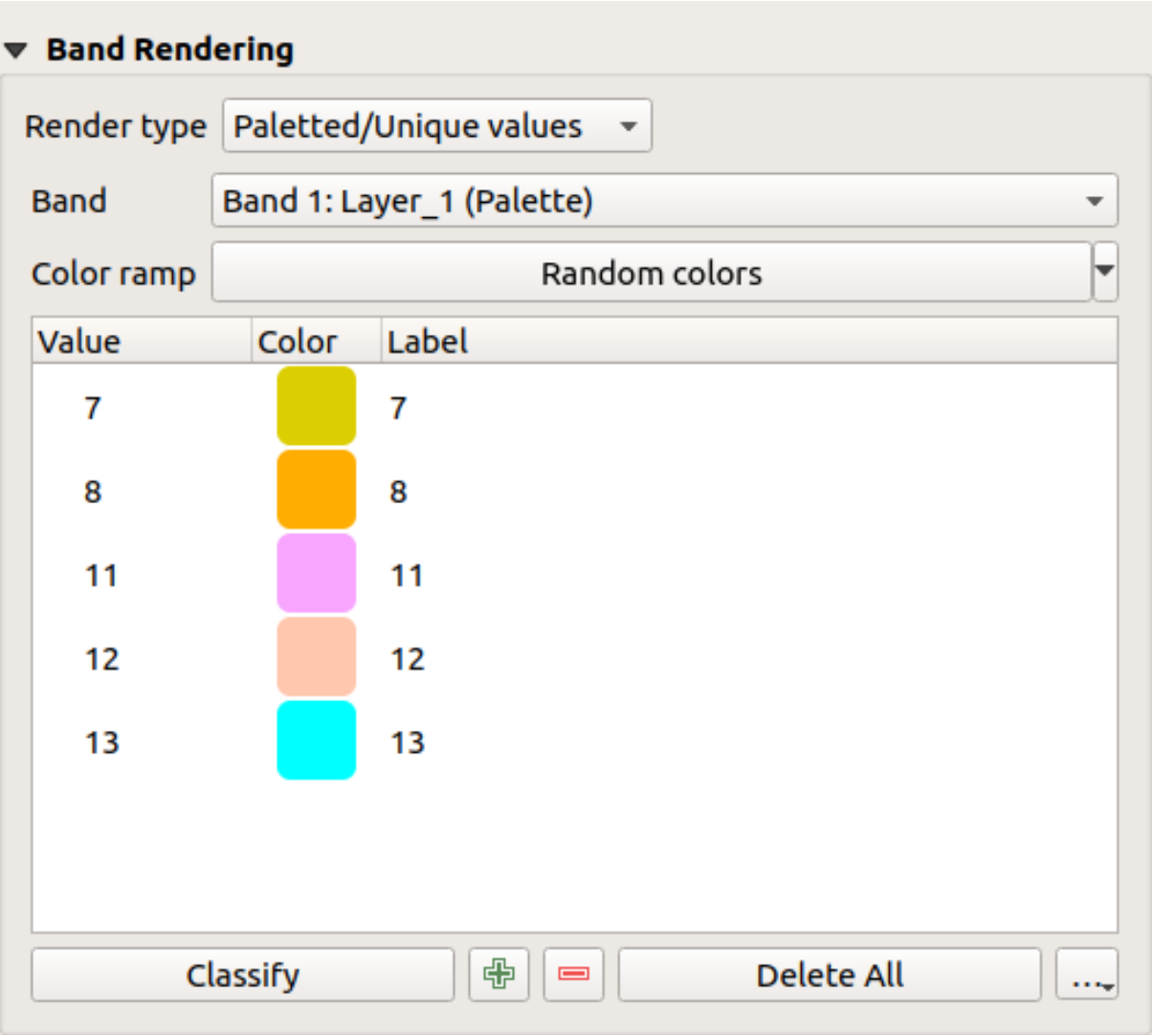


Abb. 13.3: Raster Symbolisierung - Paletten-/Eindeutige-Werte-Darstellung

Das Pulldown-Menü, das sich beim Klicken auf die Schaltfläche … (Fortgeschrittene Optionen) unterhalb der Farbkarte

auf der rechten Seite öffnet, bietet die Möglichkeit, Farbkarten zu laden (*Farbabbildung aus Datei laden…*) und zu exportieren (*Farbabbildung in Datei exportieren…*) sowie Klassen zu laden (*Klassen aus Layer laden*).

Einkanalgraustufen

Mit dieser Darstellung können Sie einen Layer mit nur einem Band mit einem *Farbverlauf* darstellen: ‚Schwarz nach Weiß' oder ‚Weiß nach Schwarz'. Sie können den Wertebereich für die Farbe (*Min* und *Max*) in den *Min/Max-Werteinstellungen* ändern.

Eine *Kontrastverbesserung*-Methode kann auf die Werte angewendet werden: ‚Keine Erweiterung', ‚Strecken auf MinMax', ‚Auf MinMax strecken und zuschneiden' und ‚Zuschneiden auf MinMax'.

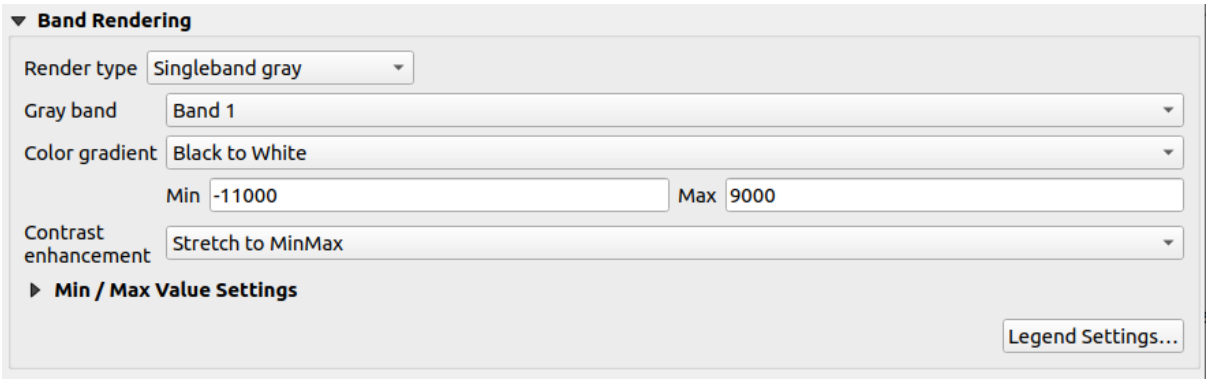


Abb. 13.4: Raster Symbolisierung - Einkanalgraustufen-Darstellung

Den Pixeln wird eine Farbe zugewiesen, die auf dem ausgewählten Farbverlauf basiert, und die Legende des Layers (im Bedienfeld *Layer* und in der Layout *Legende*) wird mit einer kontinuierlichen Farbrampe angezeigt. Drücken Sie *Legendeneinstellungen…*, wenn Sie die Einstellungen ändern möchten. Mehr Details unter *Anpassen der Rasterlegende*.

Einkanalpseudofarbe

Dies ist eine Darstellungsoption für Einzelband-Dateien, die eine kontinuierliche Palette enthalten. Sie können auch Farbkarten für ein Band eines Multibandrasters erstellen.

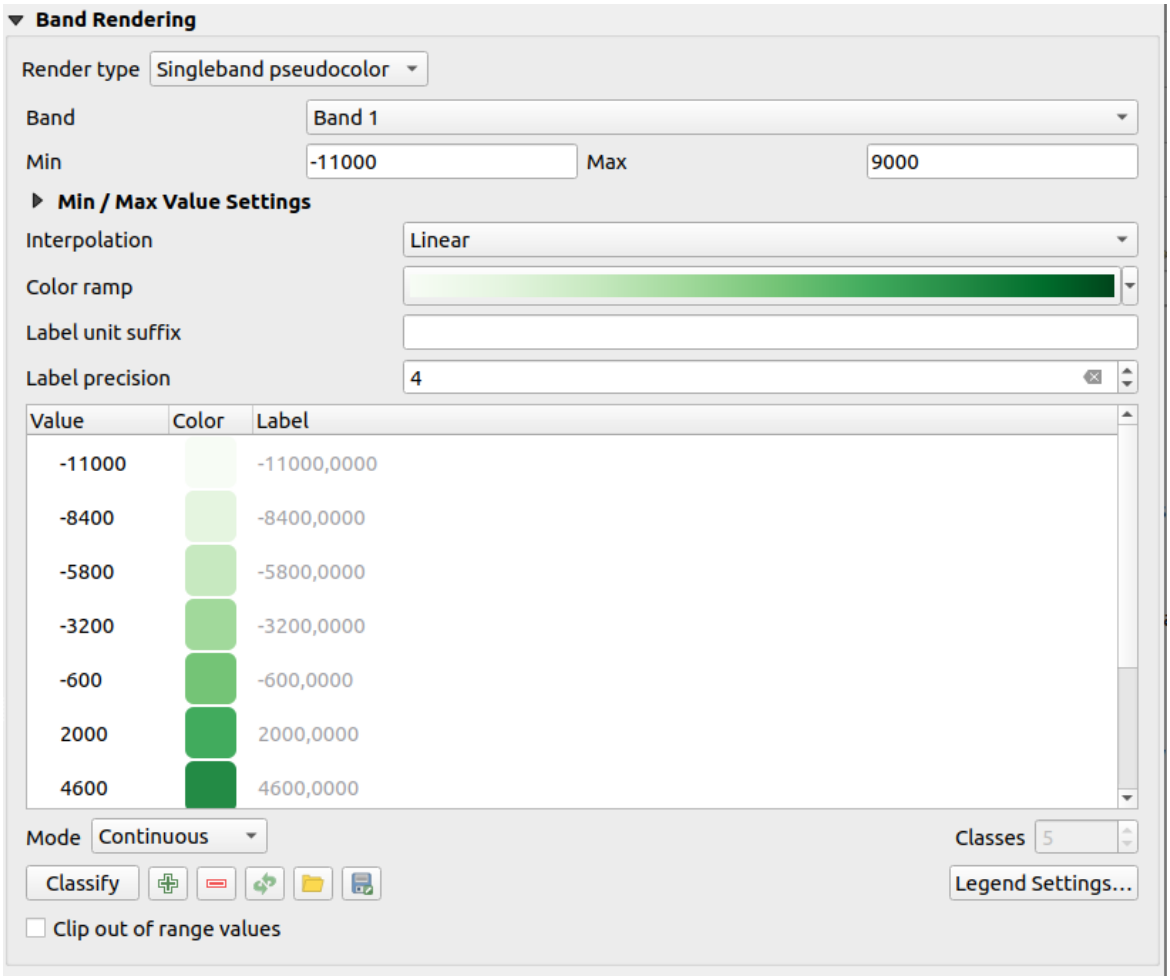


Abb. 13.5: Raster Symbolisierung - Einkanalpseudofarbe-Darstellung

Mit einem *Kanal* des Layers und einem *Wertebereich* können Sie nun interpolieren und Pixeln innerhalb von Klassen eine Darstellungsfarbe zuweisen. Mehr unter *Klassifizierung von Farbverlauf-Shader*.

Den Pixeln wird eine Farbe zugewiesen, die auf der ausgewählten Farbrampe basiert, und die Legende des Layers (im Bedienfeld *Layer* und in der Layout *Legende*) wird mit einer kontinuierlichen Farbrampe angezeigt. Drücken Sie *Legendeneinstellungen…*, wenn Sie die Einstellungen ändern oder stattdessen eine Legende mit getrennten Klassen (und Farben) verwenden möchten. Mehr Details unter *Anpassen der Rasterlegende*.

Einzelfarbe

Mit diesem Renderer können Sie einen Raster-Layer mit einer *Einzelfarbe* rendern. Dieser Renderertyp ist nützlich, wenn Sie einen Raster-Layer einheitlich darstellen möchten, ohne dass es zu Farbabweichungen aufgrund von Pixelwerten kommt.

Der Einzelfarben-Renderer kann sowohl mit Einband- als auch mit Mehrband-Raster-Layern verwendet werden. Bei Verwendung mit Mehrband-Rastern können Sie auswählen, auf welches Band die Einzelfarbe angewendet werden soll, wodurch dieses bestimmte Band effektiv einheitlich über die gesamte Raster-Layer angezeigt wird.

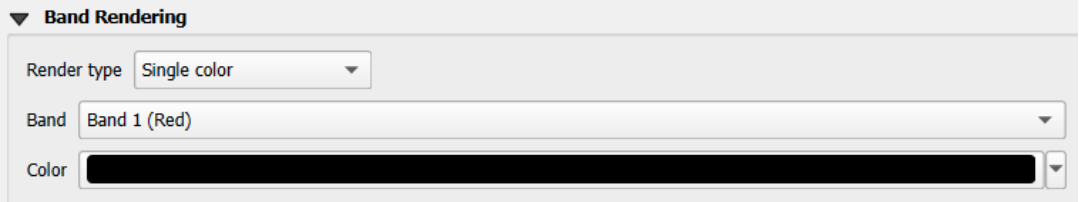


Abb. 13.6: Rastersymbolisierung – Einfarbige Darstellung

Schummerung

Darstellung eines Bandes des Rasterlayers mit Schummerung.

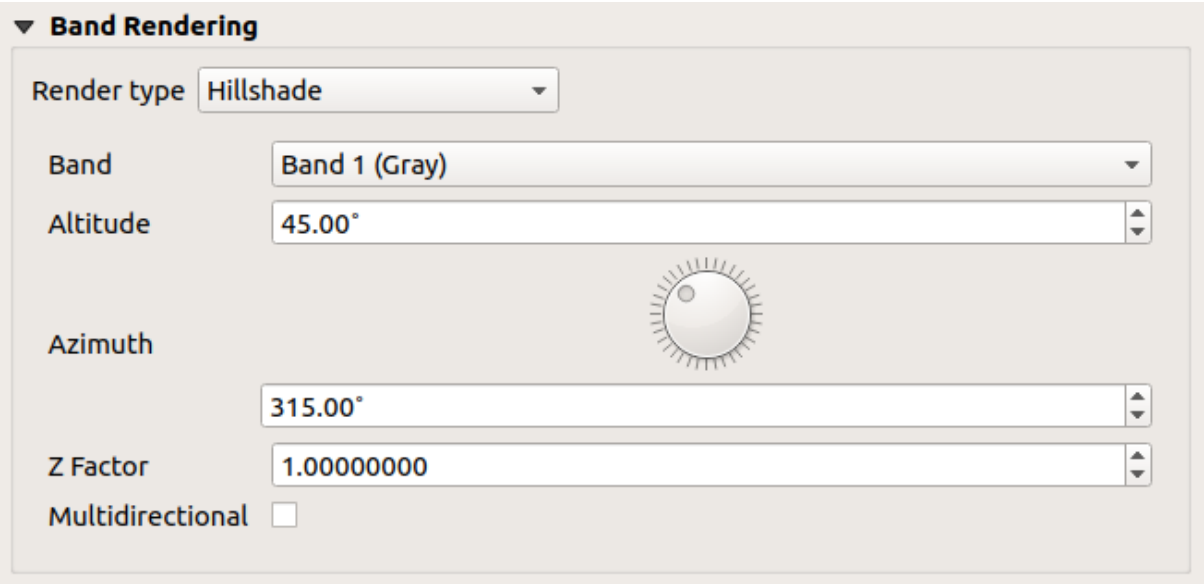


Abb. 13.7: Raster Symbolisierung - Schummerung-Darstellung

Optionen:

- *Kanal*: der Kanal des Rasterlayers, der benutzt werden soll.
- *Höhenlage*: Der Neigungswinkel der Lichtquelle (Standardwert ist `45°`).
- *Azimut*: Der Azimut der Lichtquelle (Standard ist `315°`).
- *Z-Faktor*: Skalierungsfaktor für die Werte des Rasterbandes (Standard ist `1`).
- *Mehrere Richtung*: Legt fest, ob multidirektionale Schummerung verwendet werden soll (Standard ist `off`).

Konturen

Dieser Renderer zeichnet Konturlinien, die on-the-fly aus dem Quellrasterband berechnet werden.

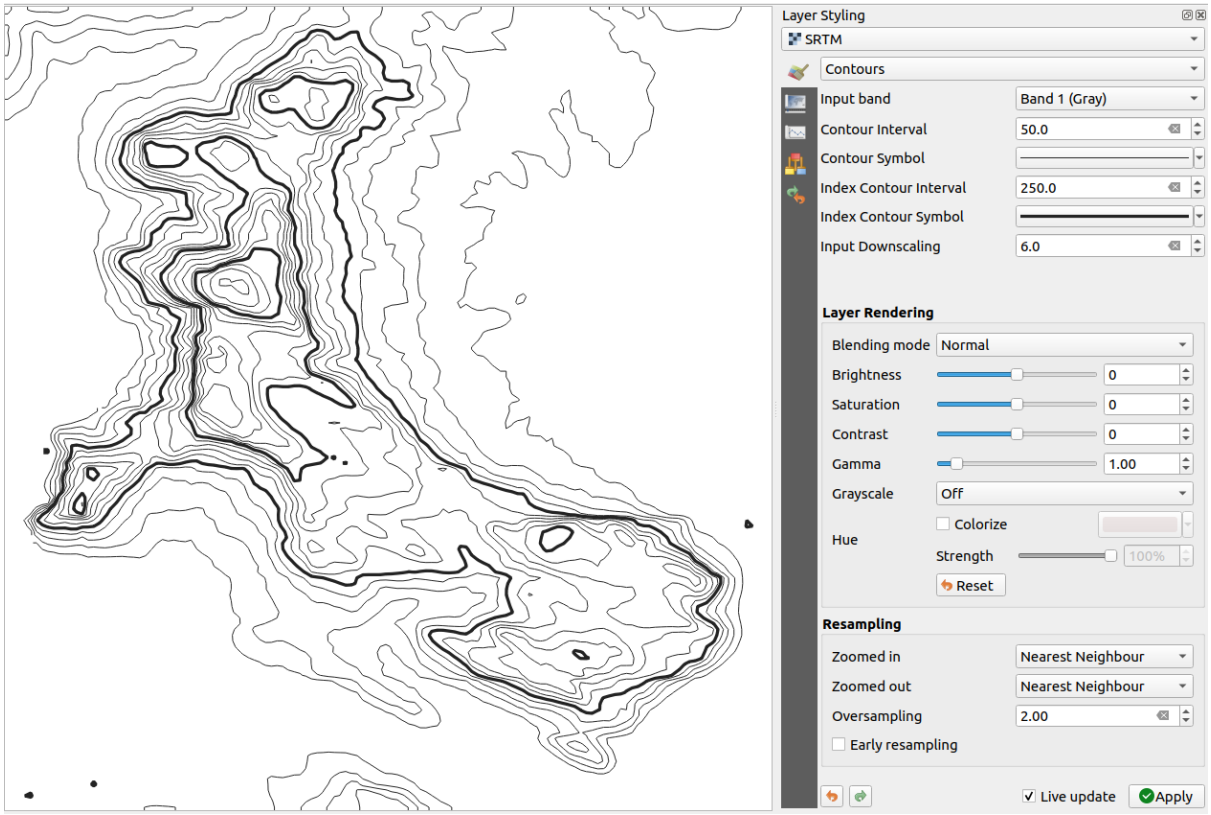


Abb. 13.8: Raster Symbolisierung - Konturen-Darstellung

Optionen:

- *Eingabekanal*: das zu verwendende Rasterband.
- *Konturintervall*: der Abstand zwischen zwei aufeinanderfolgenden Konturlinien
- *Kontursymbol*: das *Symbol* mit dem die Konturlinien dargestellt werden sollen.
- *Indexkonturintervall*: der Abstand zwischen zwei aufeinanderfolgenden **Indexkonturen**, d. h. Linien, die zur leichteren Identifizierung in einer besonderen Weise dargestellt werden, indem sie in der Regel stärker gedruckt werden als andere Konturlinien und im Allgemeinen mit einem Wert entlang ihres Verlaufs gekennzeichnet werden.
- guilabel:*Indexkontursymbol*: das Symbol, das auf die Indexkonturlinien angewendet wird
- *Eingabeniederskalieren*: Gibt an, um wie viel der Renderer die Anfrage an den Datenanbieter herunterskaliert (Standard ist `4.0`).

  Wenn Sie zum Beispiel Höhenlinien auf einem Eingangsrasterblock mit der gleichen Größe wie der Ausgangsrasterblock erzeugen, würden die erzeugten Linien zu viele Details enthalten. Dieser Detailreichtum kann durch den „Downscale“-Faktor reduziert werden, indem eine geringere Auflösung des Ausgangsrasters verlangt wird. Für einen Rasterblock 1000x500 mit Downscale 10 fordert der Renderer ein Raster 100x50 vom Provider an. Ein höherer Downscale-Faktor vereinfacht die Konturlinien (auf Kosten eines gewissen Detailverlusts).

Einstellung der Minimal- und Maximalwerte

Standardmäßig gibt QGIS die *Min* und *Max* Werte des Kanals/der Kanäle des Rasters aus. Einige sehr niedrige und/oder hohe Werte können sich negativ auf die Darstellung des Rasters auswirken. Der Beich *Min/Max-Werteinstellungen* hilft Ihnen, das Rendering zu kontrollieren.

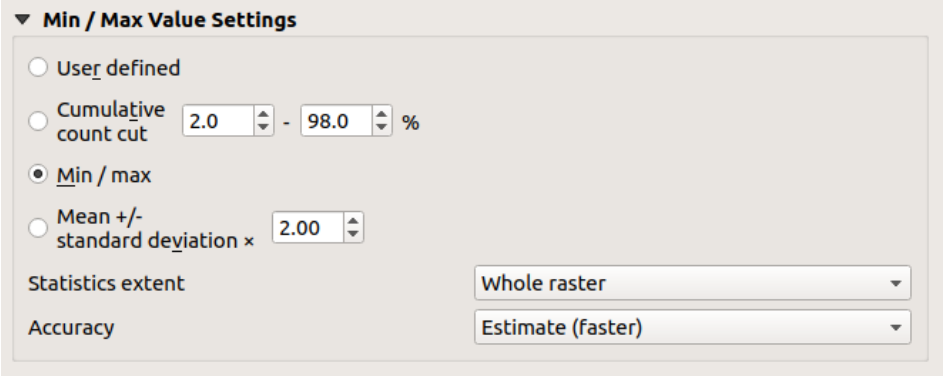


Abb. 13.9: Raster Symbolisierung - Min-/Max-Werteinstellung

Folgende Optionen sind verfügbar:

- *Benutzerdefiniert*: Die Standardwerte *Min* und *Max* des/der Bandes/Bänder können außer Kraft gesetzt werden
- *Kumulativer Zählschnitt*: Entfernt Ausreißer. Der Standardbereich der Werte ist `2%` bis `98%`, kann aber manuell angepasst werden.
- *Min / Max*: Verwendet den gesamten Wertebereich des Bildbandes.
- *Mittelwert +/- Standardabweichung x*: Erzeugt eine Farbtabelle, die nur Werte innerhalb der Standardabweichung oder innerhalb mehrerer Standardabweichungen berücksichtigt. Dies ist nützlich, wenn Sie eine oder zwei Zellen mit ungewöhnlich hohen Werten in einem Raster Layer haben, die sich negativ auf das Rendering des Rasters auswirken.

Die Berechnung der Minimal- und Maximalwerte der Banden erfolgt auf der Grundlage des:

- *Statistikausdehnung*: es kann *Ganzes Raster*, *Aktueller Ausschnitt* oder *Karte aktualisiert* sein. *Karte aktualisiert* bedeutet, dass sich die für das Rendering verwendeten Min/Max-Werte mit der Kartenansicht ändern (dynamische Dehnung).
- *Genauigkeit*, die entweder *Schätzen (schneller)* oder *Genau (langsamer)* sein kann.

Bemerkung

Bei einigen Einstellungen müssen Sie möglicherweise die Schaltfläche *Anwenden* des Layereigenschaften-Dialogs drücken, um die tatsächlichen Min- und Max-Werte in den Widgets anzuzeigen.

Klassifizierung von Farbverlauf-Shader

Diese Methode kann verwendet werden, um skalare Datensätze (Raster- oder Netzkonturen) auf der Grundlage ihrer Werte zu klassifizieren und darzustellen. Ausgehend von einem Farbverlauf und einer Anzahl von Klassen erzeugt sie Zwischenfarbkarteneinträge für Klassengrenzen. Jeder Farbe wird ein Wert zugeordnet, der aus einem Wertebereich und entsprechend einem Klassifizierungsmodus interpoliert wird. Den skalaren Datensatzelementen wird dann ihre Farbe auf der Grundlage ihrer Klasse zugewiesen.

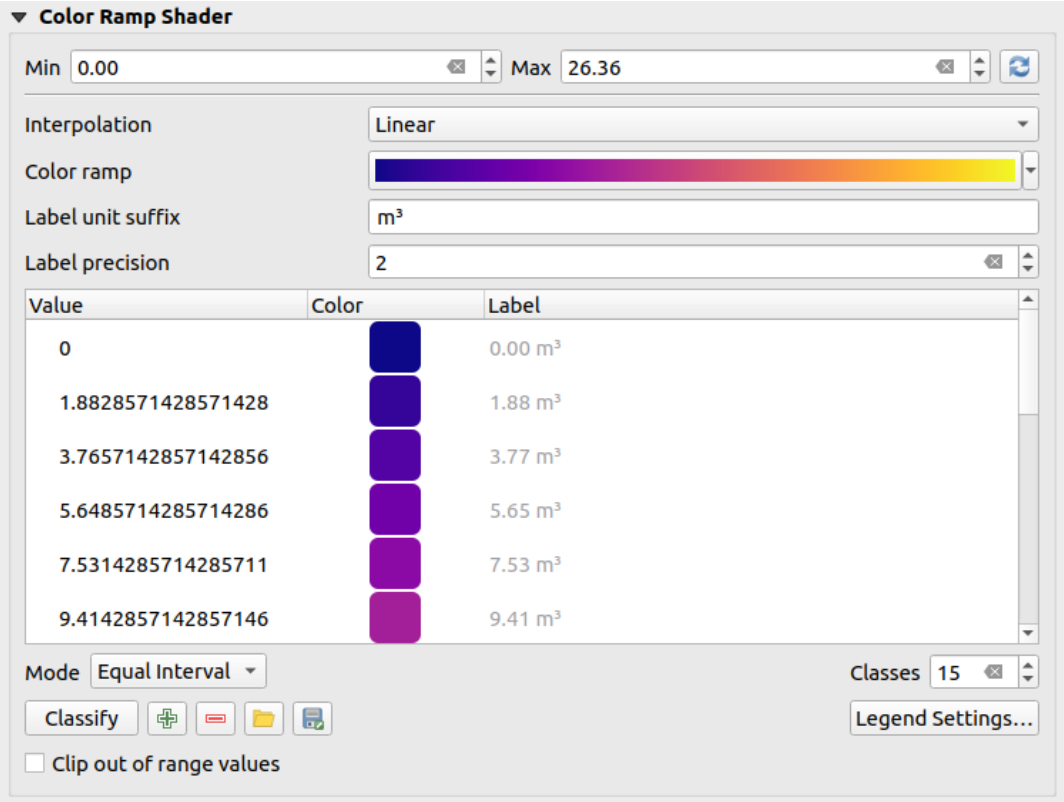


Abb. 13.10: Klassifizierung eines Datensatzes mit einem Farbverlauf-Shader

1. Die Werte *Min* und *Max* müssen definiert und zur Interpolation der Klassengrenzen verwendet werden. Standardmäßig ermittelt QGIS diese Werte aus dem Datensatz, sie können aber geändert werden.
2. Der Eintrag *Interpolation* definiert, wie skalare Elemente ihre Farbe zugewiesen bekommen:
   - *Diskret* (ein <= Symbol erscheint im Kopf der *Wert* Spalte): Die Farbe wird vom nächstgelegenen Farbkarteneintrag mit gleichem oder höherem Wert übernommen
   - *Linear*: Die Farbe wird linear aus den Farbkarteneinträgen über und unter dem Pixelwert interpoliert, d. h. jedem Datensatzwert entspricht eine eindeutige Farbe
   - *Genau* (ein = Sybol erscheint im Kopf über der *Wert* Spalte): Nur Pixel mit einem Wert, der einem Eintrag in der Farbkarte entspricht, werden mit einer Farbe versehen; andere werden nicht dargestellt.
3. Das *Farbverlauf*-Werkzeug hilft Ihnen bei der Auswahl des Farbverlaufs, der dem Datensatz zugewiesen werden soll. Wie bei *diesem Werkzeug* üblich, können Sie einen neuen Farbverlauf erstellen und den aktuell ausgewählteb bearbeiten oder speichern. Der Name des Farbverlaufs wird in der Konfiguration gespeichert.
4. Mit *Beschriftungseinheitssuffix* wird eine Beschriftung hinter dem Wert in der Legende hinzugefügt, und mit *Beschriftungsgenauigkeit* wird die Anzahl der anzuzeigenden Dezimalstellen festgelegt.
5. Mit dem Klassifizierungs *Modus* können Sie festlegen, wie die Werte auf die Klassen verteilt werden:
   - *Gleiches Intervall*: Mit der Anzahl der *Klassen* werden Grenzwerte so definiert, dass die Klassen alle die gleiche Größe haben.

- *Fortlaufend*: Klassennummer und Farbe werden aus den Stopps der Farbrampe geholt; die Grenzwerte werden entsprechend der Verteilung der Stopps in der Farbrampe gesetzt.
- *Quantile*: Mit der Anzahl der *Klassen* werden Grenzwerte definiert, sodass die Klassen die gleiche Anzahl von Elementen haben. Nicht verfügbar für *Netz-Layer*.

6. Sie können dann mit *Klassifizieren* die Klassen automatisch erstellen oder die Klassen optimieren:

   - Die Schaltfläche Werte manuell hinzufügen fügt der Tabelle einen Wert hinzu.
   - Die Schaltfläche Gewählte Zeile(n) entfernen löscht ausgewählte Werte aus der Tabelle.
   - Mit einem Doppelklick in die Spalte *Wert* können Sie den Klassenwert ändern.
   - Ein Doppelklick in die Spalte *Farbe* öffnet den Dialog *Farbe ändern*, in dem Sie eine Farbe auswählen können, die für diesen Wert gelten soll.
   - Ein Doppelklick in die Spalte *Beschriftung* ändert die Beschriftung der Klasse, aber dieser Wert wird nicht angezeigt, wenn Sie das Werkzeug zum Identifizieren von Objekten verwenden.
   - Wenn Sie mit der rechten Maustaste auf ausgewählte Zeilen in der Farbtabelle klicken, wird ein Kontextmenü angezeigt, in dem Sie die Optionen *Farbe ändern…* und *Deckkraft ändern…* für die Auswahl auswählen können.

   Sie können die Schaltflächen Farbabbildung aus Datei laden oder Farbabbildung in Datei speichern verwenden, um eine vorhandene Farbtabelle zu laden oder die Farbtabelle zur späteren Verwendung zu speichern.

7. Mit linear *Interpolation* können Sie auch konfigurieren:

   - *Werte auf Bereich beschränken*: Standardmäßig weist die lineare Methode den Werten im Datensatz, die kleiner als der eingestellte *Min* (bzw. größer als der eingestellte *Max*) Wert sind, die Farbe der ersten Klasse (bzw. der letzten Klasse) zu. Aktivieren Sie diese Einstellung, wenn Sie diese Werte nicht darstellen wollen.
   - *Legendeneinstellungen* zur Anzeige im Bedienfeld *Layer* und in der *Layout-Legende*. Mehr Details unter *Anpassen der Rasterlegende*.

Anpassen der Rasterlegende

Wenn Sie einen Farbverlauf auf einen Raster- oder Netzlayer anwenden, möchten Sie vielleicht eine Legende anzeigen, die die Klassifizierung zeigt. Standardmäßig zeigt QGIS eine kontinuierliche Farbrampe mit Minimal- und Maximalwerten im Bedienfeld *Layer* und in der *Layout-Legende* an. Dies kann über die Schaltfläche *Legendeneinstellungen…* im Klassifizierungs-Fenster angepasst werden.

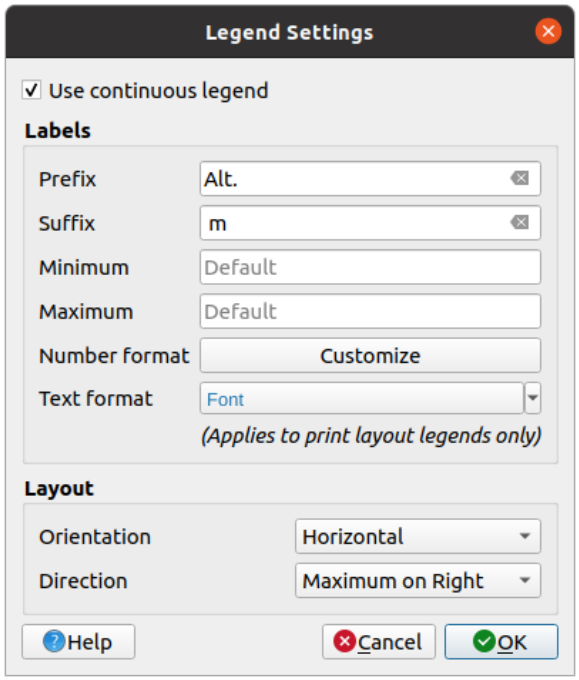


Abb. 13.11: Anpassen einer Rasterlegende

In diesem Dialogfeld können Sie festlegen, ob Sie die Option *Fortlaufende Legende verwenden* möchten: Wenn diese Option nicht aktiviert ist, werden in der Legende getrennte Farben angezeigt, die den verschiedenen angewandten Klassen entsprechen. Diese Option ist nicht verfügbar für die Darstellungsart Einkanalgraustufen.

Wenn Sie das Kontrollkästchen *Fortlaufende Legende verwenden* aktivieren, können Sie sowohl die Beschriftungen als auch die Layout-Eigenschaften der Legende konfigurieren.

Beschriftungen

- Hinzufügen eines *Präfix* und eines *Suffix* zu den Beschriftungen
- Ändern Sie die Werte *Minimum* und *Maximum* für die Anzeige in der Legende
- *Anpassen* des *Zahlenformats*
- *Anpassen* des *Textformats* zur Verwendung in der Layout-Legende.

Layout

- Steuert die *Orientierung* des Legenden-Farbverlaufs; sie kann **Vertikal** oder **Horizontal** sein
- Steuerung der *Richtung* der Werte in Abhängigkeit von der Ausrichtung:
  - Wenn Sie vertikal arbeiten, können Sie das **Maximum oben** oder das **Minimum oben** anzeigen.
  - Bei horizontaler Anordnung können Sie das **Maximum rechts** oder das **Minimum rechts** anzeigen.

Layerdarstellung

Unabhängig von der Darstellungsart, die Sie für Ihren Layer ausgewählt haben, können Sie spezielle Darstellungs-Effekte für den gesamten Rasterlayer einstellen:

- Verwenden Sie einen der Mischmodi (siehe *Mischmodi*)
- Setzen Sie benutzerdefinierte Werte für *Helligkeit*, *Sättigung*, *Gamma* und *Kontrast*.
- Mit dem *Farben umkehren* wird der Layer mit entgegengesetzten Farben dargestellt. Praktisch, um zum Beispiel OpenStreetMap-Kacheln in den dunklen Modus zu schalten.

- Schalten Sie den Layer auf die Option *Graustufen* entweder ‚Nach Helligkeit', ‚Nach Leuchtkraft' oder ‚Nach Durchschnitt'.
- *Einfärben* und Einstellen der *Stärke* von *Farbton* in der Farbtabelle

Drücken Sie *Zurücksetzen*, um alle benutzerdefinierten Änderungen an der Darstellung des Layers zu entfernen.

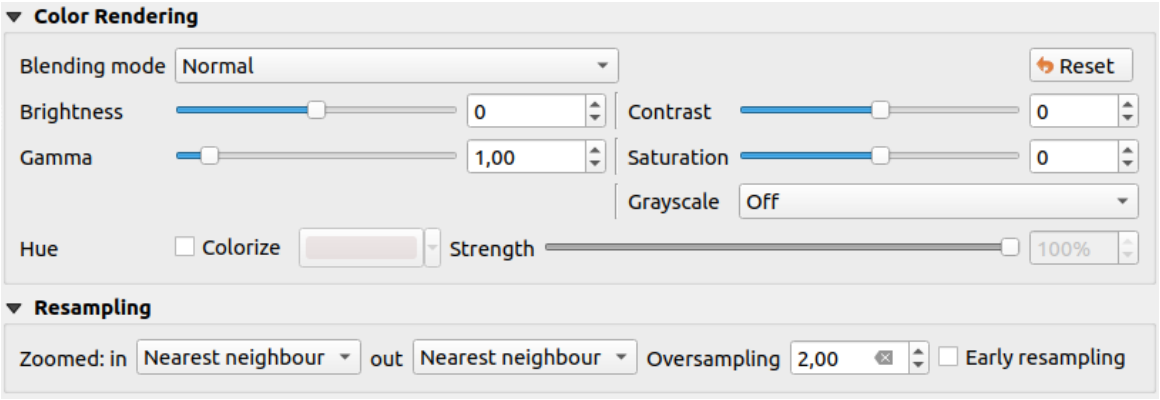


Abb. 13.12: Raster Symbolisierung - Einstellungen für Layerdarstellung und Abtastung

Abtastung

Die Option *Abtastung* hat Auswirkungen, wenn Sie ein Bild vergrößern oder verkleinern. Abtastung-Modi können das Aussehen der Karte optimieren. Sie berechnen eine neue Grauwertmatrix durch eine geometrische Transformation.

Bei Anwendung der Methode „Nächster Nachbar" kann die Karte beim Heranzoomen eine pixelige Struktur erhalten. Dieses Erscheinungsbild kann durch die Methode „Bilinear (2x2-Kernel)" oder „Cubic (4x4-Kernel)" verbessert werden, bei der scharfe Kanten verwischt werden. Der Effekt ist ein glatteres Bild. Diese Methode kann z. B. auf digitale topografische Rasterkarten angewendet werden.

Frühe Abtastung: erlaubt es, das Raster-Rendering auf der Provider-Ebene zu berechnen, wo die Auflösung der Quelle bekannt ist, und gewährleistet einen besseren Zoom beim Rendering mit QGIS Custom Styling. Sehr praktisch für Kachelraster, die mit einer *Interpretationsmethode* geladen wurden.

13.1.4 Transparenz

QGIS bietet die Möglichkeit, die *Transparenz* eines Layers einzustellen.

Verwenden Sie den Schieberegler *Globale Deckkraft*, um festzulegen, inwieweit die darunter liegenden Layer (falls vorhanden) durch den aktuellen Raster-Layer sichtbar sein sollen. Dies ist sehr nützlich, wenn Sie Raster Layer überlagern (z.B. eine schattierte Reliefkarte, die von einer klassifizierten Rasterkarte überlagert wird). Das Aussehen der Karte wird dadurch dreidimensionaler. Die Deckkraft des Rasters kann durch Daten definiert werden und z. B. in Abhängigkeit von der Sichtbarkeit eines anderen Layers, von zeitlichen Variablen, auf verschiedenen Seiten eines Atlasses usw. variieren.

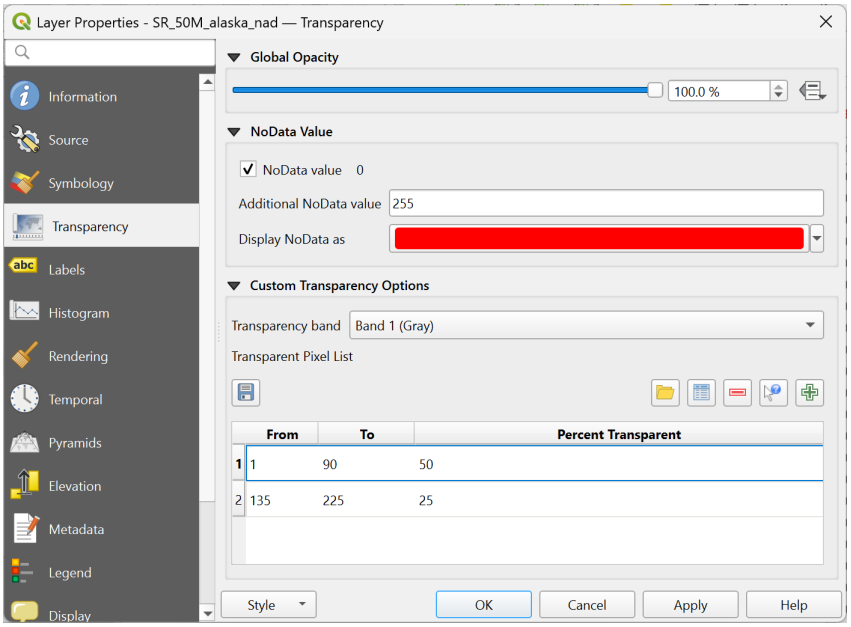


Abb. 13.13: Rasterlayereigenschaften - Transparenz-Reiter

Mit *Leerwert* meldet QGIS den ursprünglichen Quellwert ohne Daten (falls definiert), den Sie beim Rendering berücksichtigen können. Zusätzlich können Sie einen Rasterwert eingeben, der als *Zusätzlicher Leerwert* behandelt werden soll. Mit dem Farbselektor *Leerwert anzeigen als* können Sie den Pixeln, die keine Daten enthalten, eine benutzerdefinierte Farbe zuweisen, anstelle der standardmäßigen transparenten Darstellung.

Eine noch flexiblere Möglichkeit, die Transparenz anzupassen, bietet der Abschnitt *Benutzerdefinierte Transparenzeinstellungen*:

- Verwenden Sie *Transparenzkanal*, um Transparenz für ein ganzes Band anzuwenden.
- Geben Sie eine Liste von Pixeln an, die transparent gemacht werden sollen, mit den entsprechenden Transparenzstufen:

  1. Klicken Sie den Werte manuell hinzufügen Knopf. Eine neue Zeile erscheint in der Pixelliste.
  2. Bei einbandbasierten Symbologien (z. B. DEMs) geben Sie die Werte **Von** und **Bis** ein und passen Sie den Wert **Prozentuale Transparenz** entsprechend an.
  3. Für multibandbasierte Symbologie (z. B. RGB-Bilder) geben Sie die Werte für **Rot**, **Grün** und **Blau** der Pixel ein und passen Sie den Wert für **Prozentuale Transparenz** an. QGIS unterstützt bei der Definition der Transparenz eine **Toleranz** für Pixelwerte. Das bedeutet, dass auch Pixel mit Farben, die den angegebenen RGB-Werten nahekommen, transparent gemacht werden können. Beachten Sie, dass diese Funktion nur für Multiband-Raster gilt.
  4. Alternativ können Sie die Pixelwerte auch direkt aus dem Raster holen, indem Sie die Schaltfläche Werte der Anzeige hinzufügen verwenden. Geben Sie dann den Transparenzwert ein.
  5. Wiederholen Sie die Schritte, um weitere Werte mit benutzerdefinierter Transparenz anzupassen.
  6. Drücken Sie die Schaltfläche *Anwenden* um das Ergebnis auf die Karte anzeigen zu lassen.

  Wie Sie sehen können, ist es recht einfach, die benutzerdefinierte Transparenz einzustellen, aber es kann ziemlich viel Arbeit sein. Deshalb können Sie die Schaltfläche In Datei exportieren verwenden, um Ihre Transparenzliste in einer Datei zu speichern. Die Schaltfläche Aus Datei importieren lädt Ihre Transparenzeinstellungen und wendet sie auf den aktuellen Raster Layer an.

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 *Projektbeschriftungsregeln einstellen*.
- the Automated placement settings (applies to all layers) button: configure general properties on label placement and conflicts resolution. More details at *Einstellung des automatischen Platzierungssystems*.

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.

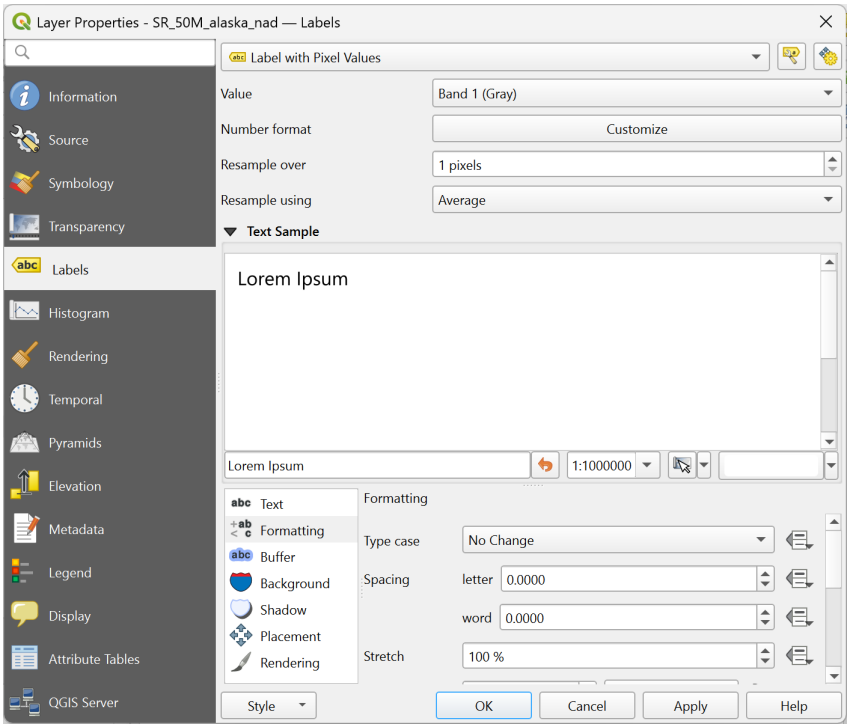


Abb. 13.14: Raster layer labeling settings

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.

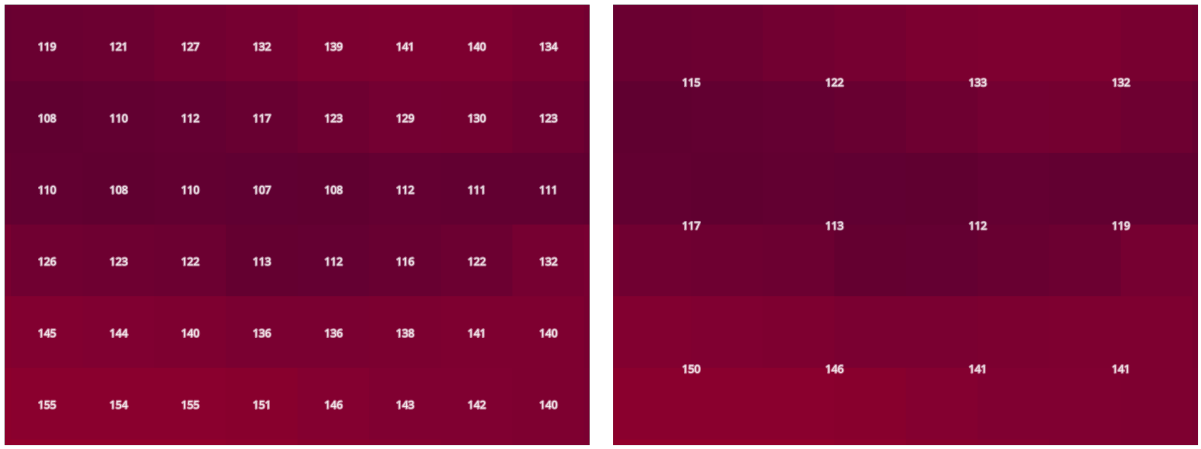


Abb. 13.15: Pixels labeled using various resampling options

Below are displayed options to customize the labels, under various tabs:

- abc *Text*
- *Formatting*
- abc *Buffer*
- *Background*
- *Shadow*
- *Placement*
- *Rendering*

Description of how to set each property is exposed at *Einrichten einer Beschriftung*.

13.1.6 Histogramm

Auf dem Reiter **|RasterHistogramm|** *Histogramm* können Sie sich die Verteilung der Werte in Ihrem Raster ansehen. Das Histogramm wird erzeugt, wenn Sie die Schaltfläche *Histogramm berechnen* drücken. Alle vorhandenen Bänder werden zusammen angezeigt. Sie können das Histogramm mit der Schaltfläche als Bild speichern.

Am unteren Rand des Histogramms können Sie im Dropdown-Menü unter *Extrem-Stil setzen für* ein Rasterband auswählen für welches sie dann den Min- und Max-Wert festlegen können. Das Dropdown-Menü *Präfs/Aktionen* bietet Ihnen erweiterte Optionen zur Anpassung des Histogramms:

- Mit der Option *Sichtbarkeit* können Sie Histogramme für einzelne Bänder anzeigen. Dazu müssen Sie die Option *Gewählten Kanal anzeigen* wählen.
- Die *Extrema-Optionen* ermöglichen es Ihnen, ‚Extrema-Markierungen immer anzeigen', ‚Zu Extrema zoomen' und ‚Stile auf Extrema setzen' zu aktivieren.
- Mit der Option *Aktionen* können Sie das Histogramm „Zurücksetzen" oder „neu berechnen", nachdem Sie die Minimal- oder Maximalwerte geändert haben.

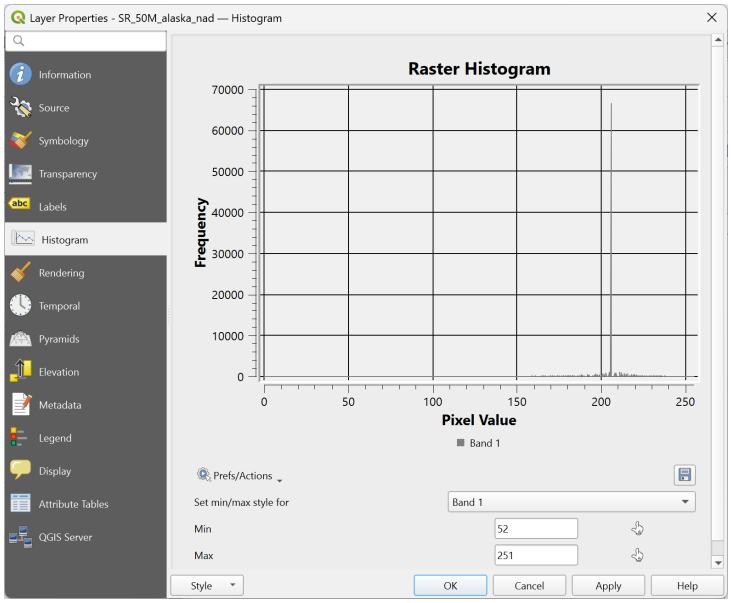


Abb. 13.16: Histogramm-Reiter der Rasterlayer-Eigenschaften

13.1.7 Darstellung

In dem Reiter *Darstellung*, ist es möglich:

- Setzen Sie *Maßstabsabhängige Sichtbarkeit* für den Layer: Sie können den Maßstab für *Maximum (inklusive)* und *Minimum (exklusive)* festlegen und damit einen Bereich definieren, in dem der Layer sichtbar ist. Die Schaltfläche Auf aktuelle Kartenmaßstab setzen hilft Ihnen, den aktuellen Maßstab der Kartenansicht als Grenze zu verwenden. Siehe *Einstellen des Sichtbarkeitsmaßstabs* für weitere Informationen.

  Bemerkung

  Sie können die skalierungsabhängige Sichtbarkeit eines Layers auch über das Bedienfeld *Layer* aktivieren: Klicken Sie mit der rechten Maustaste auf den Layer und wählen Sie im Kontextmenü *Maßstabsabhängige Sichtbarkeit des Layers setzen*.

- *Layeraktualisierungsintervall*: steuert, ob und wie regelmäßig ein Layer aufgefrischt werden kann. Verfügbare *Konfiguration*-Optionen sind:
  - *Daten neu laden*: Der Layer wird vollständig aktualisiert. Alle zwischengespeicherten Daten werden verworfen und vom Anbieter neu geholt. Dieser Modus kann dazu führen, dass die Karte langsamer aktualisiert wird.
  - *Nur Layer neu zeichnen*: Dieser Modus ist nützlich für Animationen oder wenn der Stil des Layers in regelmäßigen Abständen aktualisiert werden soll. Die Aktualisierung der Kartenansicht wird aufgeschoben, um ein mehrfaches Aktualisieren zu vermeiden, wenn für mehr als einen Layer ein Intervall für die automatische Aktualisierung festgelegt wurde.

  Es ist auch möglich, das *Intervall (Sekunden)* zwischen aufeinanderfolgenden Auffrischungen einzustellen.

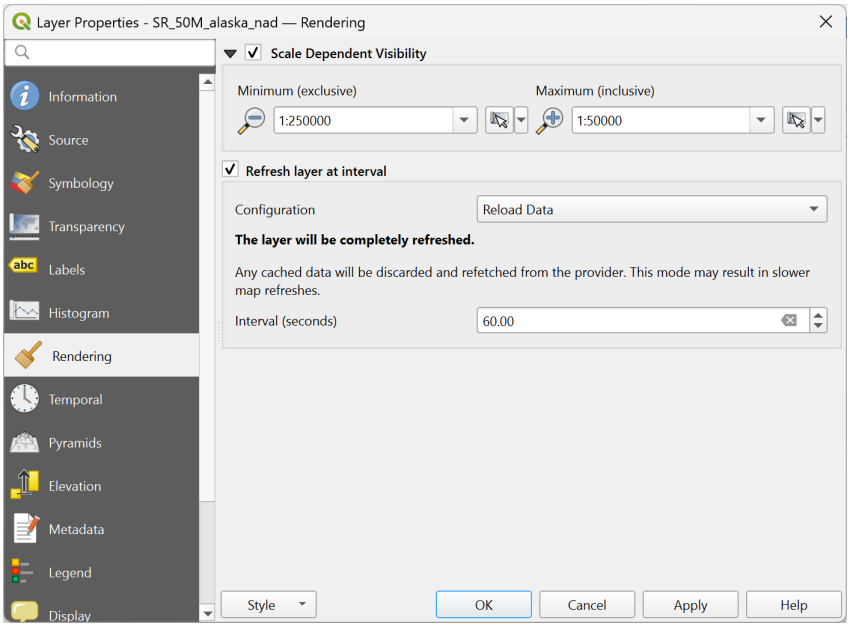


Abb. 13.17: Darstellung-Reiter der Rasterlayer-Eigenschaften

13.1.8 Eigenschaften Zeitlich

Die Registerkarte *Zeitlich* bietet Optionen zur Steuerung der Darstellung des Layers im Zeitverlauf. Für eine solche dynamische Darstellung muss die *zeitabhängige Darstellung* in der Kartenansicht aktiviert sein.

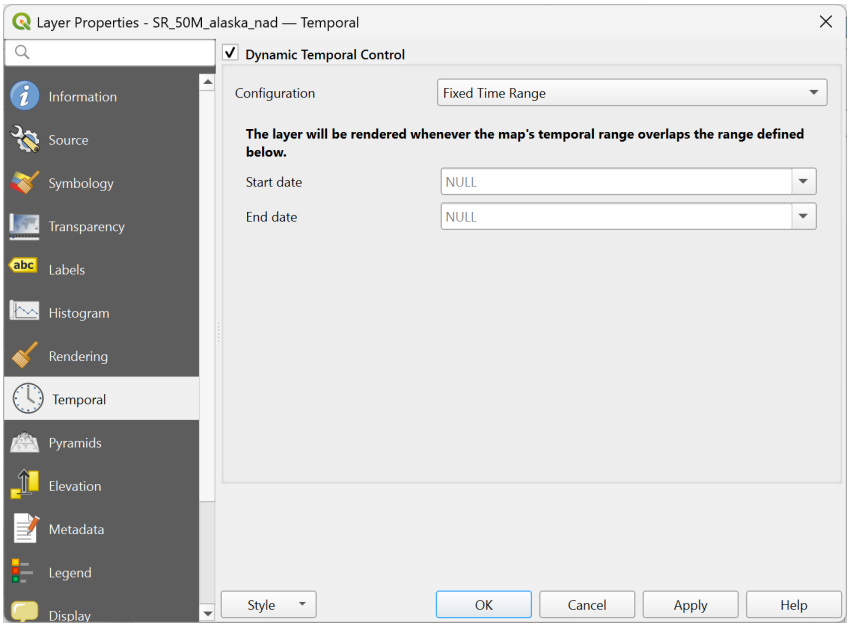


Abb. 13.18: Zeitlich-Reiter der Rasterlayer-Eigenschaften

Aktivieren Sie die Option *Dynamische Zeitsteuerung* und legen Sie fest, ob der Layer neu gezeichnet werden soll:

- *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.
- *Fester Zeitraum je Kanal*: Zeigt ein Band nur an, wenn die aktuelle Animationszeit zwischen dem Datumsbereich *Anfang* und *Ende* liegt. Mit dieser Option können Sie diese Zeitspannen für jedes Band entweder manuell festlegen oder mit der Schaltfläche automatisch Datums- und Zeitwerte generieren, was eine detaillierte zeitliche Analyse und Visualisierung ermöglicht. Dieser Modus ist besonders nützlich für die Arbeit mit Raster-Layer, bei denen jedes Band einem bestimmten Zeitraum entspricht, wie z. B. bei NetCDF-Dateien.
- *Stellt zeitliche Werte*: Interpretiert jedes Pixel in der Raster-Layer als Datums-/Zeitwert. Wenn dieser zeitliche Modus aktiv ist, werden Pixel, die nicht in den im Renderkontext angegebenen zeitlichen Bereich fallen, ausgeblendet, sodass nur zeitlich relevante Daten angezeigt werden. Dieser Modus ist wirksam für:
  - Analyse von Landnutzungsänderungen, wie beispielsweise die Beobachtung von Entwaldungsmustern.
  - Untersuchung von Überschwemmungen durch Vergleich der Wasserbedeckung zu verschiedenen Zeitpunkten.
  - Bewertung der Bewegungskosten in der Geländanalyse, beispielsweise mithilfe des Tools „r.walk“ von GRASS GIS zur Berechnung der Reisekosten in einer Landschaft.
- *Nur Layer neu zeichnen*: der Layer wird bei jedem neuen Animationsframe neu gezeichnet. Dies ist nützlich, wenn der Layer zeitbasierte Expressionswerte für Renderer-Einstellungen verwendet (z. B. datendefinierte Renderer-Opazität, um eine Rasterebene ein- und auszublenden).

13.1.9 Pyramiden

Hochauflösende Layer können die Navigation in QGIS verlangsamen. Durch die Erstellung von Kopien der Daten mit geringerer Auflösung (Pyramiden) kann die Leistung erheblich verbessert werden, da QGIS je nach Zoomstufe die am besten geeignete Auflösung auswählt.

Um Pyramiden zu erstellen, benötigen sie Schreibrecht in dem Ordner, in dem sich die Originaldaten befinden.

Wählen Sie aus der Liste *Auflösungen* die Auflösungen aus, für die Sie Pyramiden erstellen möchten, indem Sie diese anklicken.

Wenn Sie **Intern (wenn möglich)** aus dem Dropdown-Menü *Übersichtsformat* wählen, versucht QGIS, Pyramiden intern zu erstellen.

> **Bemerkung**
>
> Bitte beachten Sie, dass die Erstellung von Pyramiden die ursprüngliche Datei verändern kann und dass das Erstellen von Pyramiden nicht mehr rückgängig gemacht werden kann. Wenn Sie eine „nicht-pyramidierte“ Version Ihres Rasters aufbewahren möchten, erstellen Sie vor dem Pyramidenbau eine Sicherungskopie.

Wenn Sie **Extern** und **Extern (Erdas Imagine)** wählen, werden die Pyramiden in demselben Ordner, in dem sich das Originalraster befindet erstellt, mit demselben Namen und der Erweiterung `.ovr`.

Für die Pyramidenberechnung können mehrere *Abtastmethoden* verwendet werden:

- Nächster Nachbar
- Mittel
- Gauss
- Kubisch (4x4 Kern)
- Kubische B-Spline (4x4 Kern)
- Laczos
- Modus

- Keine

Klicken Sie schließlich auf *Pyramiden erzeugen*, um den Vorgang zu starten.

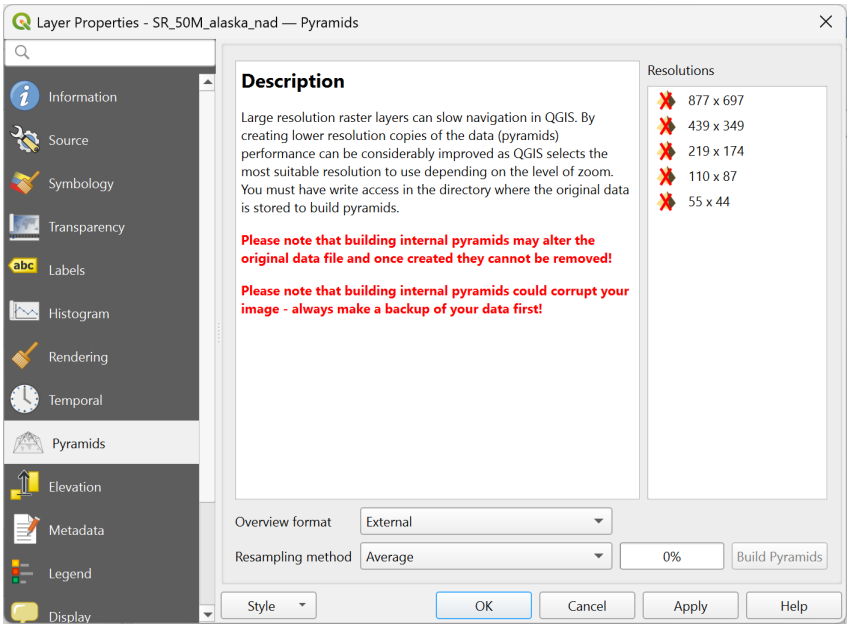


Abb. 13.19: Pyramiden-Reiter der Rasterlayer-Eigenschaften

13.1.10 Höhe

The *Elevation* tab provides options to control the layer elevation properties within a *3D map view* and its appearance in the *profile tool charts*. Specifically, you can choose to *Disable* this configuration if the layer does not contain elevation data or you can set:

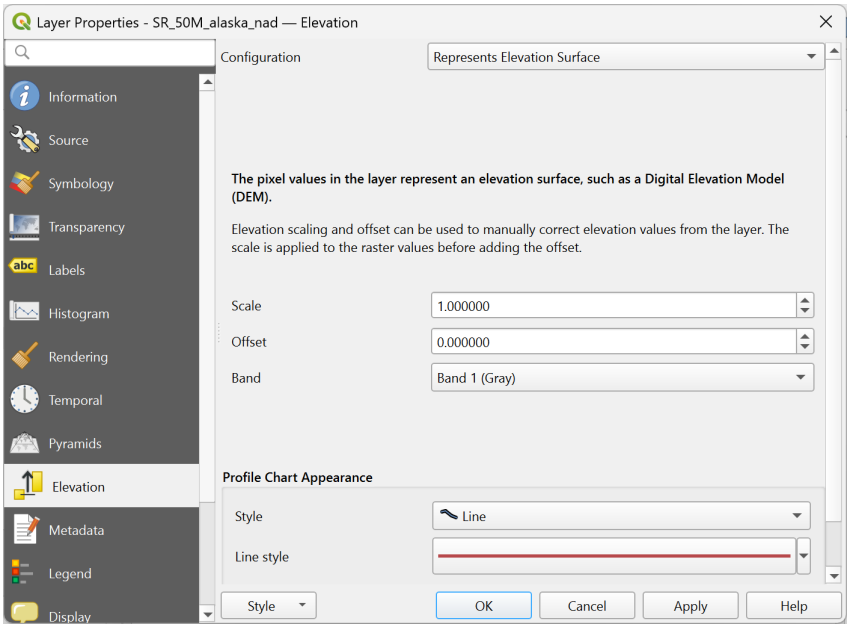


Abb. 13.20: Höhe-Reiter der Rasterlayer-Eigenschaften

- *Represents Elevation Surface*: whether the raster layer represents a height surface (e.g DEM) and the pixel values

should be interpreted as elevations. Choose this option if you want to display a raster in an *elevation profile view*. You will also need to fill in the *Band* to pick values from and can apply a *Scale* factor and an *Offset*.

- *Fixed Elevation Range*: The raster layer (or selected raster band) is associated with a fixed elevation range. This mode can be used when a layer has a single fixed elevation or a range (slice) of elevation values. If a range is specified, pixels will be extruded over this range. You can set the *Lower* and *Upper* elevation range values for the layer, and specify whether the lower or upper *Limits* are inclusive or exclusive.
- *Fixed Elevation Range Per Band*: Each band in the raster can have a fixed elevation range associated with it. This is designed for data sources that expose elevation-related data in bands, such as NetCDF files. For example, a raster with temperature data at different ocean depths. When rendering, the uppermost matching band will be selected and used for the layer's data. This feature is exposed as a user-editable table for raster bands with lower and upper values. Users can either populate the lower and upper values manually or use an *Expression* to auto-fill all band values based on expression. The expression-based fill allows you to design expressions that extract useful information from band names. For example, extracting the depth value from a band name like „Band 001: depth=-5500 (meters)".
- *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.
- *Profildiagrammerscheinungsbild*: steuert die Darstellung der Rasterhöhendaten im Profildiagramm. Der *Stil* des Profils kann eingestellt werden als:
  - eine *Linie* mit einem bestimmten *Linienstil*
  - eine Höhenfläche, die mit einem Füllsymbol entweder über (*Oberhalb füllen*) oder unter (*Unten füllen*) der Höhenkurvenlinie dargestellt wird. Die Oberflächensymbolik wird mit dargestellt:
    - einem *Füllstil*
    - und einer *Begrenzung*: die maximale (bzw. minimale) Höhe, die bestimmt, wie hoch die Füllfläche sein wird

13.1.11 Metadaten

Der Tab *Metadaten* ermöglicht es Ihnen, Metadaten für Ihren Layer zu erstellen oder zu editieren. Siehe *Metadaten* für weitere Informationen.

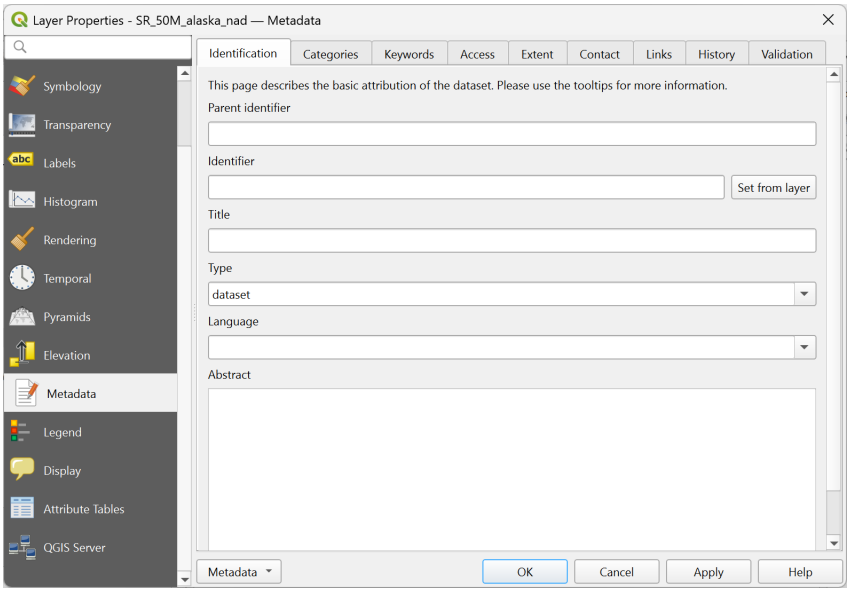


Abb. 13.21: Metadaten-Reiter der Rasterlayer-Eigenschaften

13.1.12 Eigenschaften Legende

Die Registerkarte *Legende* bietet Ihnen erweiterte Einstellungen für das Bedienfeld *Layer* und/oder die *Drucklayout Legende*. Diese Optionen umfassen:

- Abhängig von der auf den Layer angewendeten Symbologie können Sie mehrere Einträge in der Legende erhalten, die nicht unbedingt lesbar oder nützlich für die Anzeige sind. Das *Legendenplatzhalterbild* hilft Ihnen bei der *Auswahl eines Bildes* zum Ersetzen, das sowohl im Bedienfeld *Layer* als auch in der Drucklayoutlegende angezeigt wird.
- Die Funktion *In Legende eingebettete Bedienelemente* bietet Ihnen eine Liste von Elementen, die Sie in den Layer-Baum im Layer-Bedienfeld einbetten können. Die Idee dahinter ist, einen schnellen Zugriff auf einige Aktionen zu haben, die oft mit dem Layer verwendet werden (Transparenz, Filterung, Auswahl, Stil oder andere Dinge…).

  Standardmäßig stellt QGIS ein Transparenz-Widget zur Verfügung, das jedoch durch Plugins erweitert werden kann, die ihre eigenen Widgets registrieren und den von ihnen verwalteten Layern eigene Aktionen zuweisen.

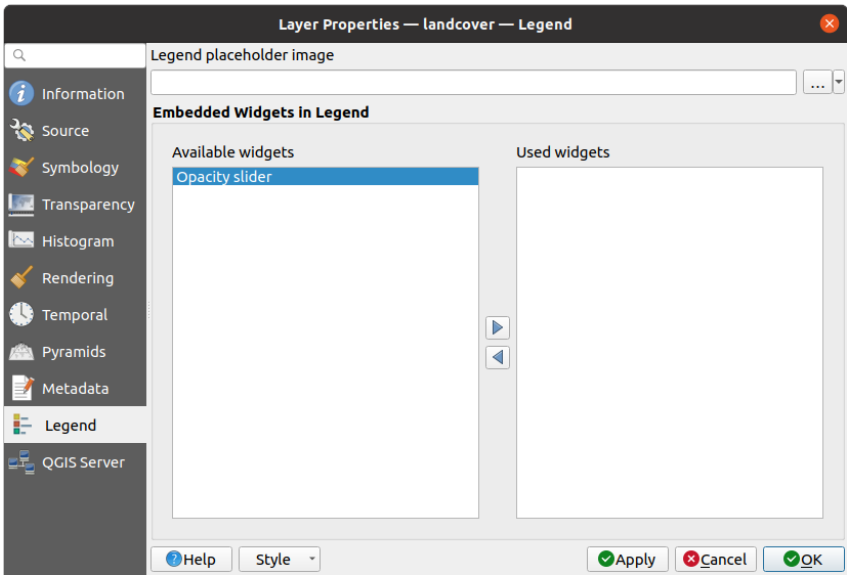


Abb. 13.22: Legende-Reiter der Rasterlayer-Eigenschaften

13.1.13 Anzeige

Der Reiter *Anzeige* hilft Ihnen bei der Konfiguration von HTML-Kartentipps, die zur Identifizierung von Pixeln verwendet werden können:

- *Kartentipps einschalten* steuert, ob Kartentipps für den Layer angezeigt werden sollen
- The *HTML Map Tip* provides a complex and full HTML text editor for map tips, mixing QGIS expressions and html styles and tags (multiline, fonts, images, hyperlink, tables, …). You can check the result of your code sample in the *Preview* frame. You can also select and edit existing expressions using the *Insert/Edit Expression* button.

  You might look for expressions located in *Rasters* group or the `@layer_cursor_point` variable in the *Expressions* dialog.

  Bemerkung

  Understanding the *Insert/Edit Expression* button behavior

  Wenn Sie Text innerhalb eines Ausdrucks (zwischen „[%“ und „%]“) auswählen oder wenn kein Text ausgewählt ist, sich der Cursor jedoch innerhalb eines Ausdrucks befindet, wird automatisch der gesamte Ausdruck zur Bearbeitung ausgewählt. Befindet sich der Cursor oder ausgewählter Text außerhalb eines Ausdrucks, wird das Dialogfeld mit der Auswahl geöffnet.

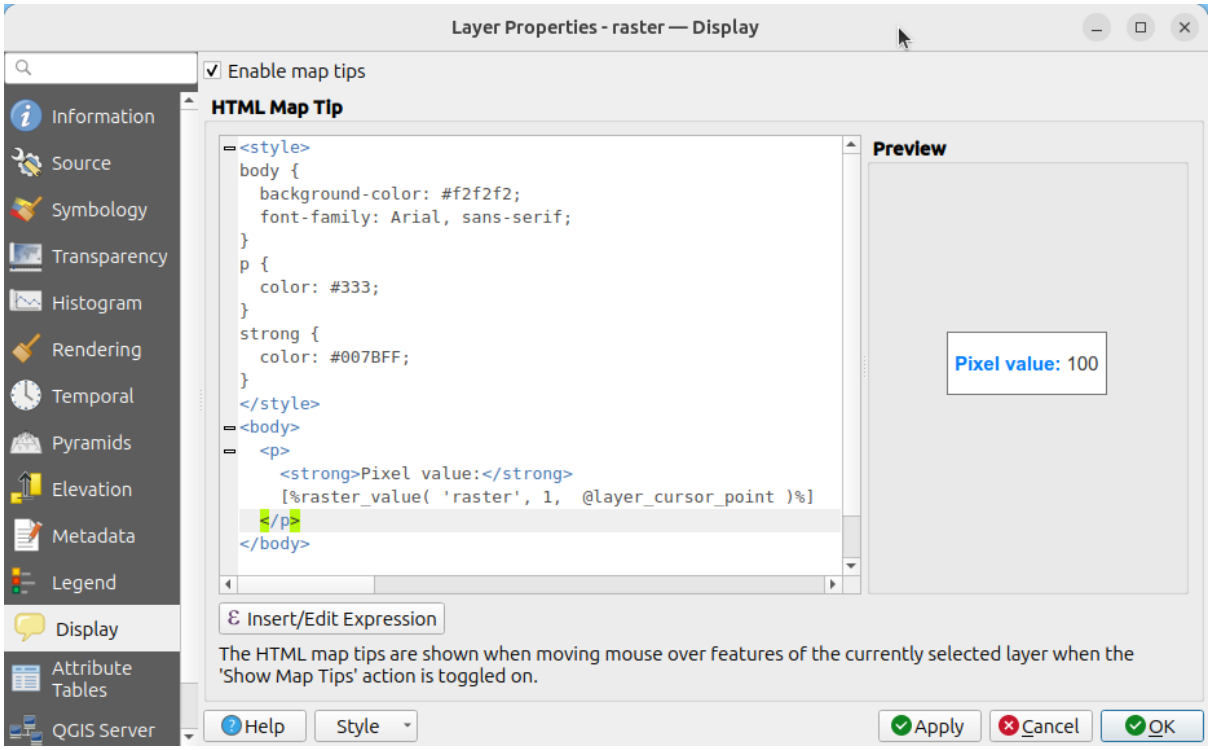


Abb. 13.23: Anzeige-Reiter der Rasterlayer-Eigenschaften

So zeigen Sie Kartentipps an:

1. Wählen Sie den Menüpunkt *Ansicht* ► *Kartenhinweise anzeigen* oder klicken Sie auf das Symbol

in der *Attributwerkzeugleiste*.

2. Vergewissern Sie sich, dass der Layer, auf den Sie abzielen, aktiv ist und dass die Eigenschaft *Kartentipps anzeigen* aktiviert ist.

3. Bewegen Sie den Mauszeiger über ein Pixel, und die entsprechenden Informationen werden eingeblendet.

Der Kartentipp ist ein ebenenübergreifendes Objekt, d. h., wenn er einmal aktiviert ist, bleibt er eingeschaltet und gilt für alle mit dem Kartentipp versehenen Layer im Projekt, bis er wieder ausgeschaltet wird.

13.1.14 QGIS Server

Die Registerkarte *QGIS Server* hilft Ihnen, Einstellungen der Daten zu konfigurieren, wenn diese mit QGIS Server veröffentlicht werden. Die Konfiguration betrifft:

- *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`.
- *Beschreibung*: ein *Titel* und eine *URL* zur Identifizierung der Datenquelle.
- *Metadaten-URL*: eine Liste von *URL* für die Metadaten, die vom *Type* `FGDC` oder `TC211` sein können und in dem *Format* `text/plain` oder `text/xml`
- *Legenden-URL*: eine *URL* für die Legende, entweder als `image/png` oder `image/jpeg` *Format*`.

Bemerkung

Wenn der zu veröffentlichende Raster Layer bereits von einem Webdienst bereitgestellt wird, können weitere *Eigenschaften* eingestellt werden.

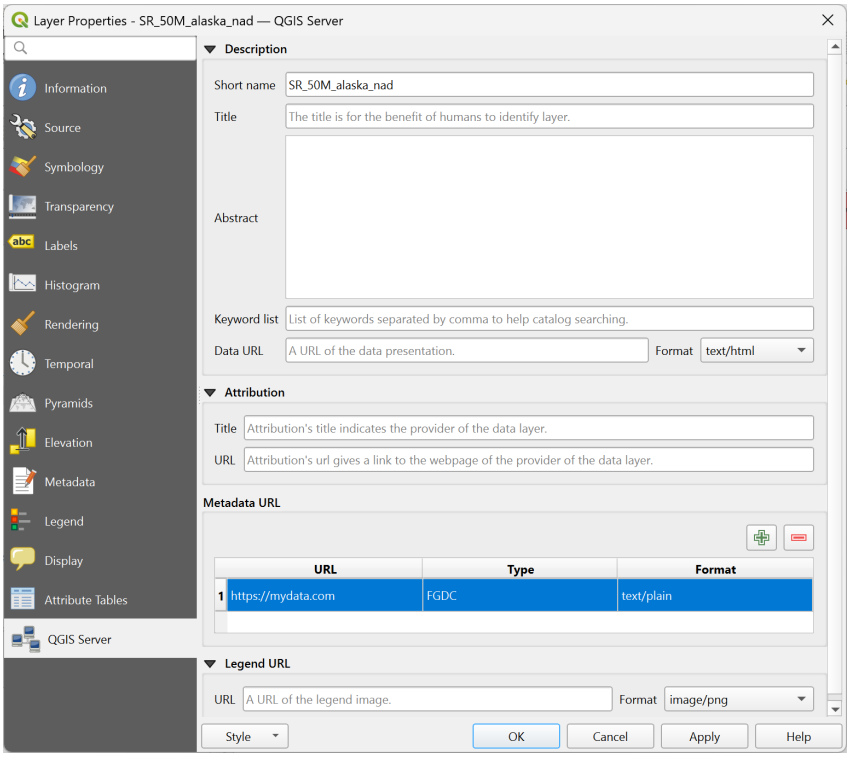


Abb. 13.24: QGIS-Server-Reiter der Rasterlayer-Eigenschaften

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*.

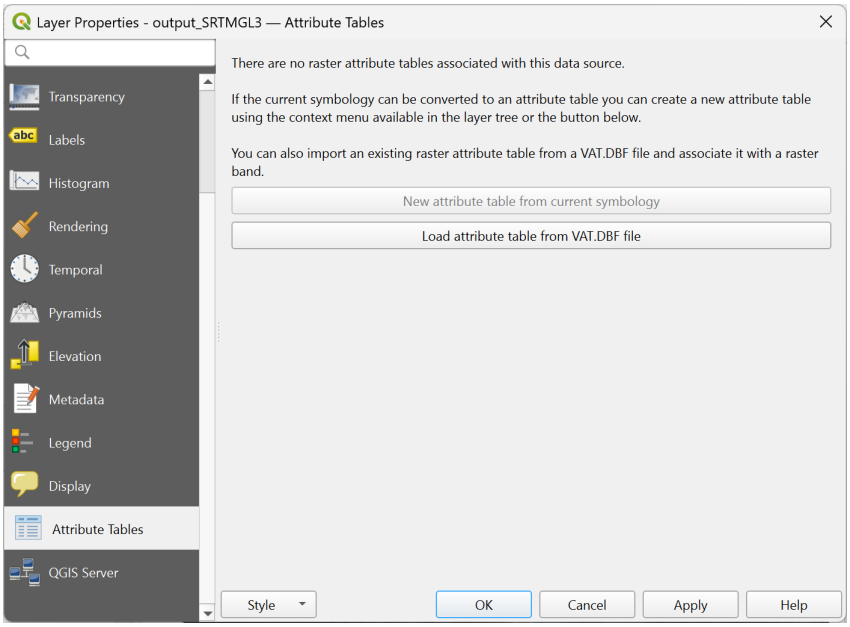


Abb. 13.25: Accessing the raster attribute table

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.

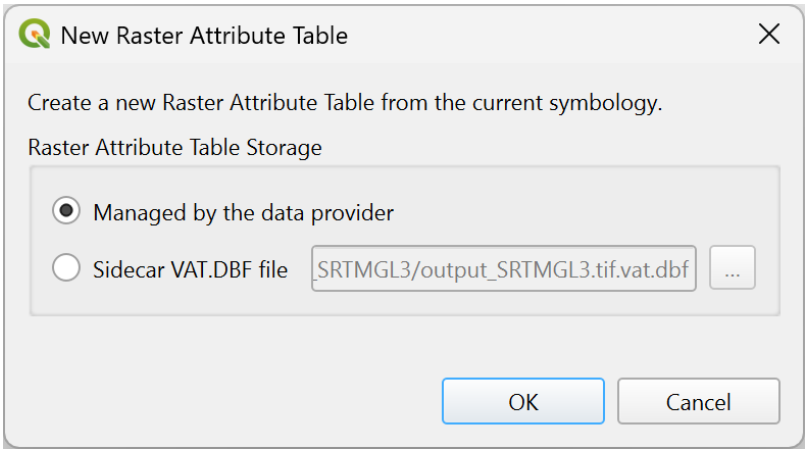


Abb. 13.26: Create new attribute table from current symbology

- *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.

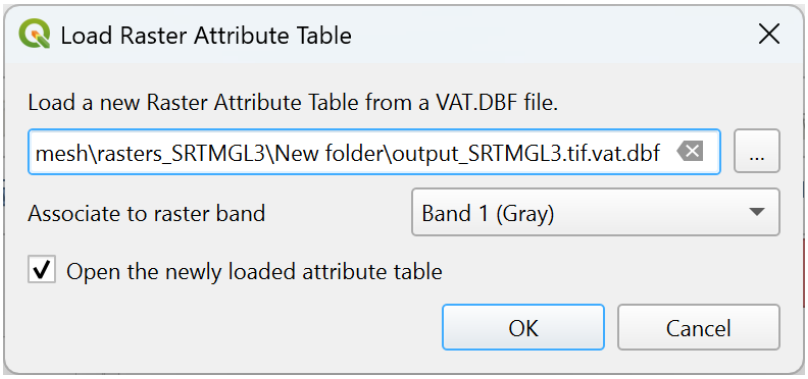


Abb. 13.27: Load attribute table from VAT.DBF file

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.

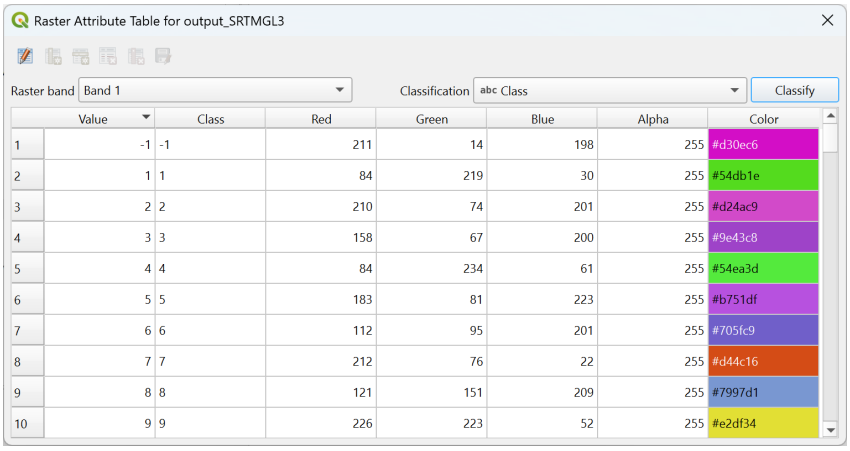


Abb. 13.28: Raster Attribute Table

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.

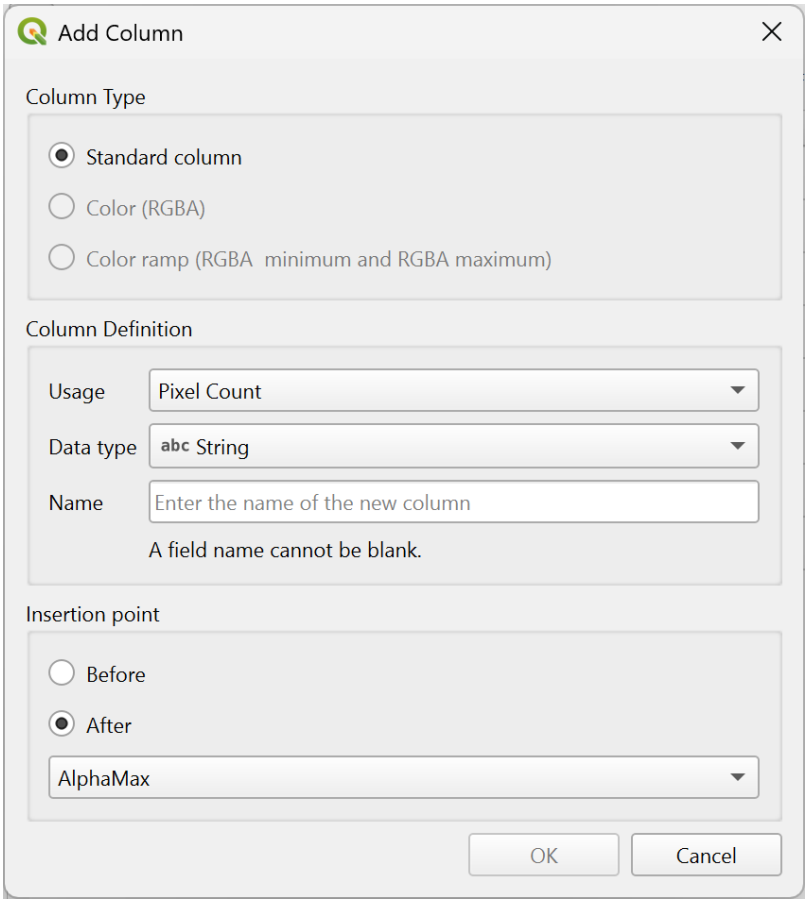


Abb. 13.29: Add Column dialog

- *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 Identifizieren von Rasterzellen

Mit dem Werkzeug *Objekte abfragen* können Sie Informationen über bestimmte Punkte in einem Raster Layer erhalten.

So verwenden Sie das Werkzeug *Objekte abfragen*:

1. Wählen Sie den Rasterlayer im Layer-Bedienfeld aus.
2. Klicken Sie auf das Werkzeug *Objekte abfragen* in der Werkzeugleiste oder drücken Sie `Strg+Umschalt+I`.
3. Click on the point in the raster layer that you want to identify. The pixel will get highlighted.

The Identify Results panel will open in its default `Tree` view and display information about the clicked point. Formatting of the results vary depending on the provider of the layer. For example:

- For a local raster layer: below the name of the layer, you have on the left the band(s) of the clicked pixel, and on the right their respective value.
- For a remote layer such as WMS, a *Format* menu allows you to select whether the information should be displayed as *HTML*, *Feature* or *Text*.

These values can also be rendered (from the *View* menu located at the bottom of the panel) in:

- eine `Tabelle` - zeigt die Informationen über die identifizierten Objekte und ihre Werte in einer Tabelle.
- eine `Graph`-Ansicht - zeigt die Informationen über die identifizierten Objekte und ihre Werte in einem Graphen.

In der Baum-Ansicht finden Sie unter den Pixelattributen Information, die *abgeleitet* sind, wie zum Beispiel:

- `X-Koordinate` und `Y-Koordinate` des angeklickten Punktes
- Column and row of the point clicked (pixel) when compatible

13.2 Rasteranalyse

13.2.1 Rasterrechner

Die Funktion *Rasterrechner* im Menü *Raster* ermöglicht es, Berechnungen auf der Basis von vorhandenen Rasterpixelwerten durchzuführen (siehe Abb. 13.30). Die Ergebnisse werden in einen neuen Layer in einem GDAL-unterstützten Format geschrieben.

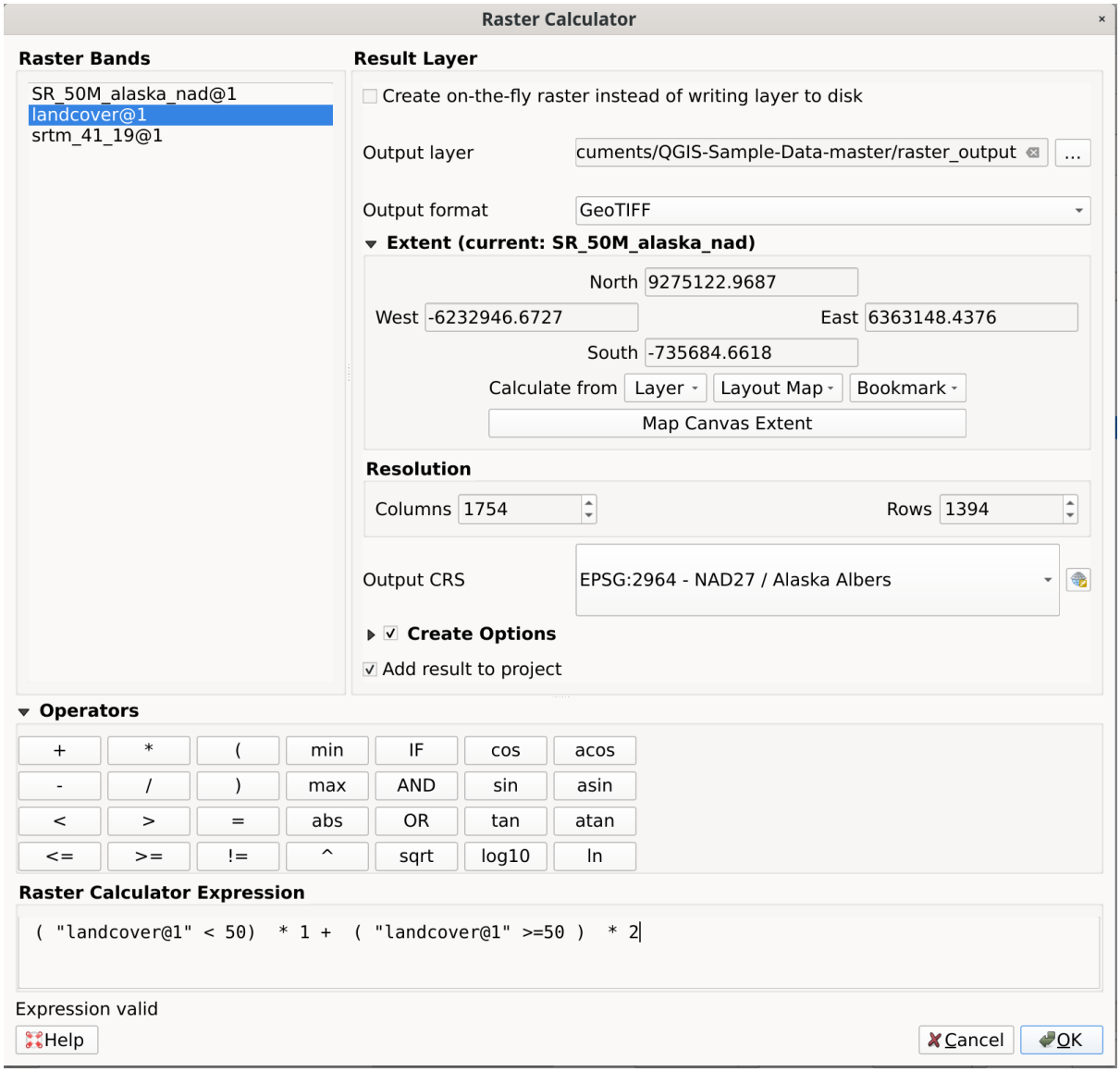


Abb. 13.30: Rasterrechner

Die Liste *Rasterkanäle* enthält alle geladenen Layer, die verwendet werden können. Um ein Raster zum Feld für den Rasterberechnungsausdruck hinzuzufügen, doppelklicken Sie auf seinen Namen in der Liste Felder. Sie können dann die Operatoren verwenden, um Berechnungsausdrücke zu konstruieren, oder Sie können sie einfach in das Feld eingeben.

Im Abschnitt *Ergebnislayer* müssen Sie einen Ausgabelayer definieren. Sie können:

- *Temporärraster erstellen, statt auf der Festplatte zu speichern*:
  - Ist diese Option nicht aktiviert, wird die Ausgabe als neue einfache Datei auf der Festplatte gespeichert. Ein *Ausgabe-Layer* mit Angabe des Pfads und ein *Ausgabeformat* sind erforderlich.
  - Wenn dieses Kontrollkästchen aktiviert ist, wird ein virtueller Layer erstellt, d. h. ein Rasterlayer, der durch seine URI definiert ist und dessen Pixel on-the-fly berechnet werden. Es handelt sich dabei nicht um eine neue Datei auf der Festplatte; der virtuelle Layer ist immer noch mit den in der Berechnung verwendeten Rastern verbunden, was bedeutet, dass ein Löschen oder Verschieben dieser Raster den Layer zerstören würde. Ein *Layername* kann angegeben werden, ansonsten wird der Berechnungsausdruck als solcher verwendet. Wird der virtuelle Layer aus dem Projekt entfernt, so wird er gelöscht, er kann aber auch über das Kontextmenü des Layers *Export* ► *Speichern als…* in einer Datei gespeichert werden.

- Define the *Spatial extent* of the calculation using the *extent selector* widget.
- Legen Sie die *Auflösung* des Layers anhand der Spalten- und Zeilennummer fest. Wenn der Input Layer eine andere Auflösung hat, werden die Werte mit dem Nearest Neighbor Algorithmus neu abgetastet.
- 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*.
- Mit dem Kontrollkästchen *Ergebnis zum Projekt hinzufügen* wird der Ergebnis-Layer automatisch zum Legendenbereich hinzugefügt und kann visualisiert werden. Bei virtuellen Rastern standardmäßig aktiviert.

Der Abschnitt *Operatoren* enthält alle verfügbaren Operatoren. Um einen Operator zum Ausdrucksfeld des Rasterrechners hinzuzufügen, klicken Sie auf die entsprechende Schaltfläche. Mathematische Berechnungen (`+`, `-`, `*`, …) und trigonometrische Funktionen (`sin`, `cos`, `tan`, …) sind verfügbar. Bedingte Ausdrücke (`=`, `!=`, `<`, `>=`, …) geben entweder 0 für falsch oder 1 für wahr zurück und können daher mit anderen Operatoren und Funktionen verwendet werden.

Siehe auch

Algorithmen *Raster calculator* und *Raster calculator (virtual)*

Rasterrechnerausdruck

Der Dialog

Der Dialog *Rasterrechnerausdruck* bietet die Möglichkeit, Ausdrücke für Pixelberechnungen zwischen einer Reihe von Raster-Layern zu schreiben.

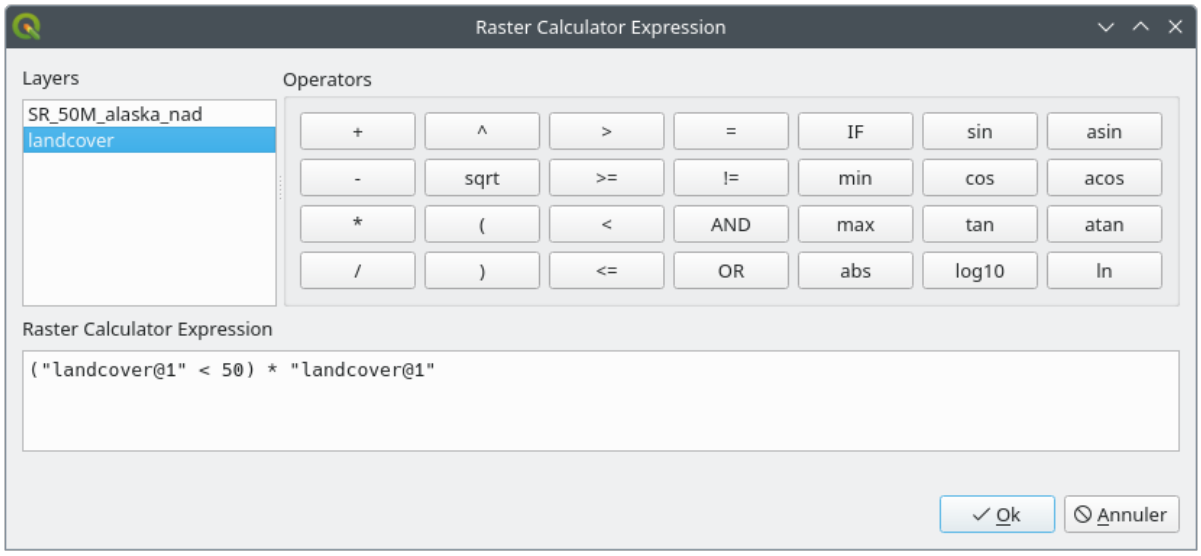


Abb. 13.31: Rasterrechnerausdruck

- *Layer*: Zeigt die Liste aller in der Legende geladenen Raster Layer an. Diese können verwendet werden, um das Ausdrucksfeld zu füllen (Doppelklick zum Hinzufügen). Rasterlayer werden durch ihren Namen und die Nummer des Bandes bezeichnet: `layer_name@band_number`. Zum Beispiel wird das erste Band eines Layers mit dem Namen `DEM` als `DEM@1` bezeichnet.
- *Operatoren*: enthält eine Reihe von Berechnungsoperatoren für die Pixelmanipulation:
  - Arithmetisch: `+`, `-`, `*`, `` `sqrt``, `abs`, `ln`, …

- Trigonometrisch: *sin*, `cos`, `tan`, …
- Vergleich: =, `!=`, <, >=, …
- Logisch: `IF`, `AND`, `OR`, `(`, `)`
- Statistisch: `min`, `max`

Klicken Sie auf die entsprechende Schaltfläche, um dem Ausdrucksfeld des Rasterrechners einen Operator hinzuzufügen.

- *Rasterrechnerausdruck* ist der Bereich, in dem der Ausdruck zusammengesetzt ist

Beispiele

Höhenwerte von Meter zu Fuß konvertieren

Für das Erstellen eines Rasterlayers in Fuß aus einem Rasterlayer in Metern müssen Sie den Konvertierungsfaktor von Metern zu Fuß benutzen: 3.28. Der Ausdruck lautet:

```
"elevation@1" * 3.28
```

Eine Maske verwenden

Wenn Sie Teile des Rasterlayers ausmaskieren wollen, weil Sie nur an Höhenwerten über 0 Metern interessiert sind, können Sie den folgenden Ausdruck zum Erstellen einer Maske und zum gleichzeitigen Anwenden auf den Rasterlayer verwenden.

```
("elevation@1" >= 0) * "elevation@1"
```

Mit anderen Worten, für jede Zelle größer oder gleich 0 wird der bedingte Ausdruck mit 1 ausgewertet, was den ursprünglichen Wert durch Multiplikation mit 1 beibehält, ansonsten wird der bedingte Ausdruck zu 0 ausgewertet, was den Rasterwert auf 0 setzt und die Maske im laufenden Betrieb erzeugt.

Ein Raster klassifizieren

Wenn Sie ein Raster klassifizieren wollen, sagen wir beispielsweise in zwei Höhenklassen, können Sie die folgenden Ausdrücke verwenden um ein Raster mit zwei Werten 1 und 2 in einem Schritt erstellen.

```
("elevation@1" < 50) * 1 + ("elevation@1" >= 50) * 2
```

In anderen Worten stelle den Wert für jede Zelle mit einem Wert kleiner 50 auf 1 ein. Für jede Zelle mit einem Wert größer als oder gleich 50 stelle den Wert 2 ein.

Oder Sie können den Operator `IF` verwenden.

```
if ( elevation@1 < 50 , 1 , 2 )
```

KAPITEL 14

Working with Mesh Data

14.1 What's a mesh?

Ein Netz-Layer ist ein unstrukturiertes Gitter, das in der Regel zeitliche und andere Komponenten enthält. Die räumliche Komponente umfasst eine Sammlung von Knotenpunkten, Kanten und/oder Flächen im 2D- oder 3D-Raum:

- **Eckpunkte –** XY(Z)-Punkte (im Koordinatenreferenzsystem der Ebene)
- **Kanten –** verbinden Paare von Eckpunkten
- **Flächen –** Eine Fläche ist eine Reihe von Kanten, die eine geschlossene Form bilden – in der Regel ein Dreieck oder ein Viereck (Quad), seltener Polygone mit mehr Eckpunkten.

Relying on the above, mesh layers can thus have different types of structure:

- 1D Meshes: consist of vertices and edges. An edge connects two vertices and can have assigned data (scalars or vectors) on it. The 1D mesh network can be for example used for modelling of an urban drainage system.
- 2D meshes: consist of faces with triangles, regular or unstructured quads.
- 3D layered meshes: consist of multiple stacked 2D unstructured meshes each extruded in the vertical direction (levels) by means of a vertical coordinate. The vertices and faces have the same topology in each vertical level. The mesh definition (vertical level extrusion) could in general change in time. The data is usually defined in volume centres or by some parametric function.

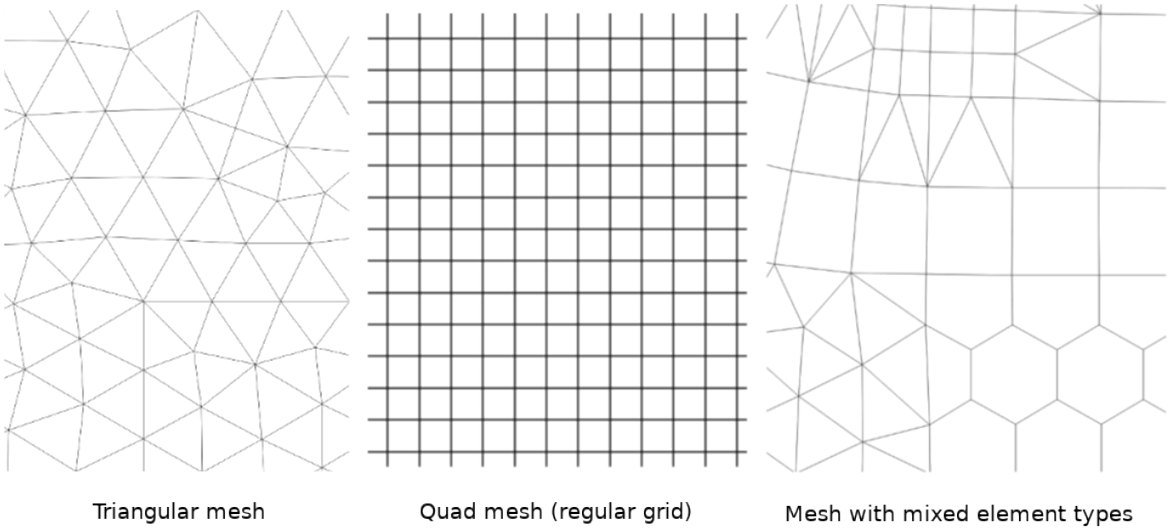


Abb. 14.1: Different mesh types

Mesh provides information about the spatial structure. In addition, the mesh can have datasets (groups) that assign a value to every vertex. For example, having a triangular mesh with numbered vertices as shown in the image below:

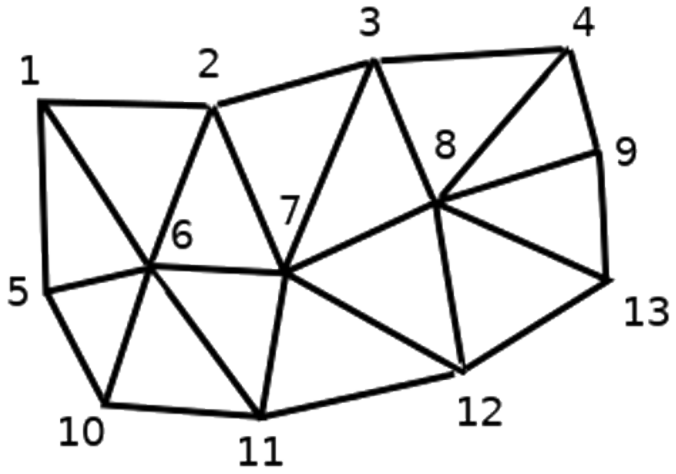


Abb. 14.2: Triangular grid with numbered vertices

Each vertex can store different datasets (typically multiple quantities), and those datasets can also have a temporal dimension. Thus, a single file may contain multiple datasets.

The following table gives an idea about the information that can be stored in mesh datasets. Table columns represent indices of mesh vertices, each row represents one dataset. Datasets can have different datatypes. In this case, it stores wind velocity at 10m at a particular moments in time (t1, t2, t3).

In a similar way, the mesh dataset can also store vector values for each vertex. For example, wind direction vector at the given time stamps:

Tab. 14.1: Example of mesh dataset

10 metre wind	1	2	3	...
10 metre speed at time=t1	17251	24918	32858	…
10 metre speed at time=t2	19168	23001	36418	…
10 metre speed at time=t3	21085	30668	17251	…
…	…	…	…	…
10m wind direction time=t1	[20,2]	[20,3]	[20,4.5]	…
10m wind direction time=t2	[21,3]	[21,4]	[21,5.5]	…
10m wind direction time=t3	[22,4]	[22,5]	[22,6.5]	…
…	…	…	…	…

We can visualize the data by assigning colors to values (similarly to how it is done with *Singleband pseudocolor* raster rendering) and interpolating data between vertices according to the mesh topology. It is common that some quantities are 2D vectors rather than being simple scalar values (e.g. wind direction). For such quantities it is desirable to display arrows indicating the directions.

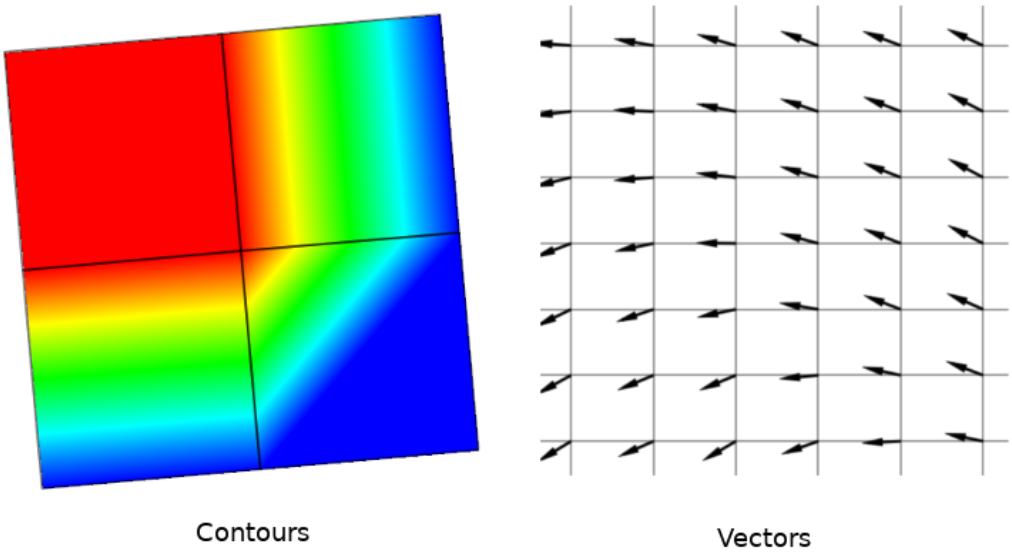


Abb. 14.3: Possible visualisation of mesh data

14.2 Supported formats

QGIS accesses mesh data using the MDAL drivers, and natively supports a variety of formats. Whether QGIS can edit a mesh layer depends on the format and the mesh structure type.

To load a mesh dataset into QGIS, use the *Mesh* tab in the *Data Source Manager* dialog. Read *Einen Netz-Layer laden* for more details.

14.3 Mesh Dataset Properties

The *Layer Properties* dialog for a mesh layer provides general settings to manage dataset groups of the layer and their rendering (active dataset groups, symbology, 2D and 3D rendering). It also provides information about the layer.

Sie können den *Layereigenschaften* Dialog öffnen, indem Sie:

- im *Layer `Bedienfeld einen Doppelklick auf den Layer machen oder einen Rechtsklick und im Kontextmenü :guilabel:`Eigenschaften …* auswählen
- in der Menüleiste *Layer* ► *Layereigenschaften …* auswählen wenn der Layer aktiv ist

The mesh *Layer Properties* dialog provides the following sections:

Tab. 14.2: Tabs of the Mesh Layer Properties

Information	*Source*	*Symbology* [1]
Labels [1]	*3D View* [1]	*Rendering*
Temporal	*Elevation*	*Metadata*

Bemerkung

Most of the properties of a mesh layer can be saved to or loaded from a `.qml` using the *Style* menu at the bottom of the dialog. More details at *Verwalten von benutzerdefinierten Stilen*.

[1] Also available in the *Layer styling panel*

14.3.1 Information

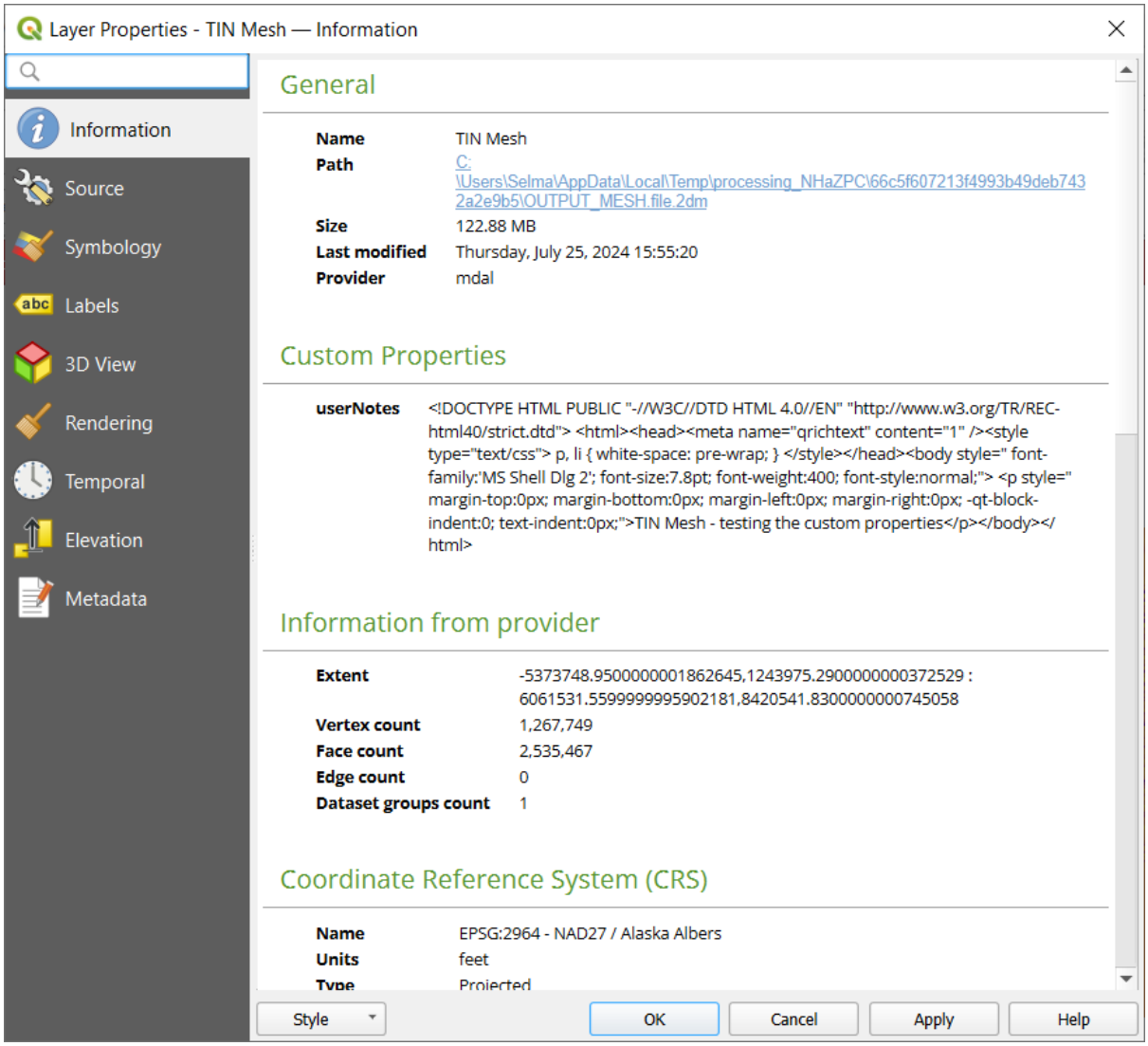


Abb. 14.4: Mesh Layer Information Properties

Die Registerkarte |Metadaten| *Information* ist nur zum Lesen vorgesehen und stellt einen interessanten Ort dar, um schnell zusammengefasste Informationen und Metadaten über den aktuellen Layer zu erhalten. Die bereitgestellten Informationen sind:

- general such as name in the project, source path, list of auxiliary files, last save time and size, the used provider
- Benutzerdefinierte Eigenschaften, die verwendet werden, um zusätzliche Informationen über den Layer im aktiven Projekt zu speichern. Zu den standardmäßigen benutzerdefinierten Eigenschaften können *Layer-Notizen* gehören. Weitere Eigenschaften können mit PyQGIS erstellt und verwaltet werden, insbesondere mit der Methode setCustomProperty().
- based on the provider of the layer: extent, vertex, face, edges and/or dataset groups count
- das Koordinatenbezugssystem: Name, Einheiten, Methode, Genauigkeit, Referenz (d.h., ob es statisch oder dynamisch ist)
- extracted from filled *metadata*: access, extents, links, contacts, history…

14.3.2 Quelle

The *Source* tab displays basic information about the selected mesh, including:

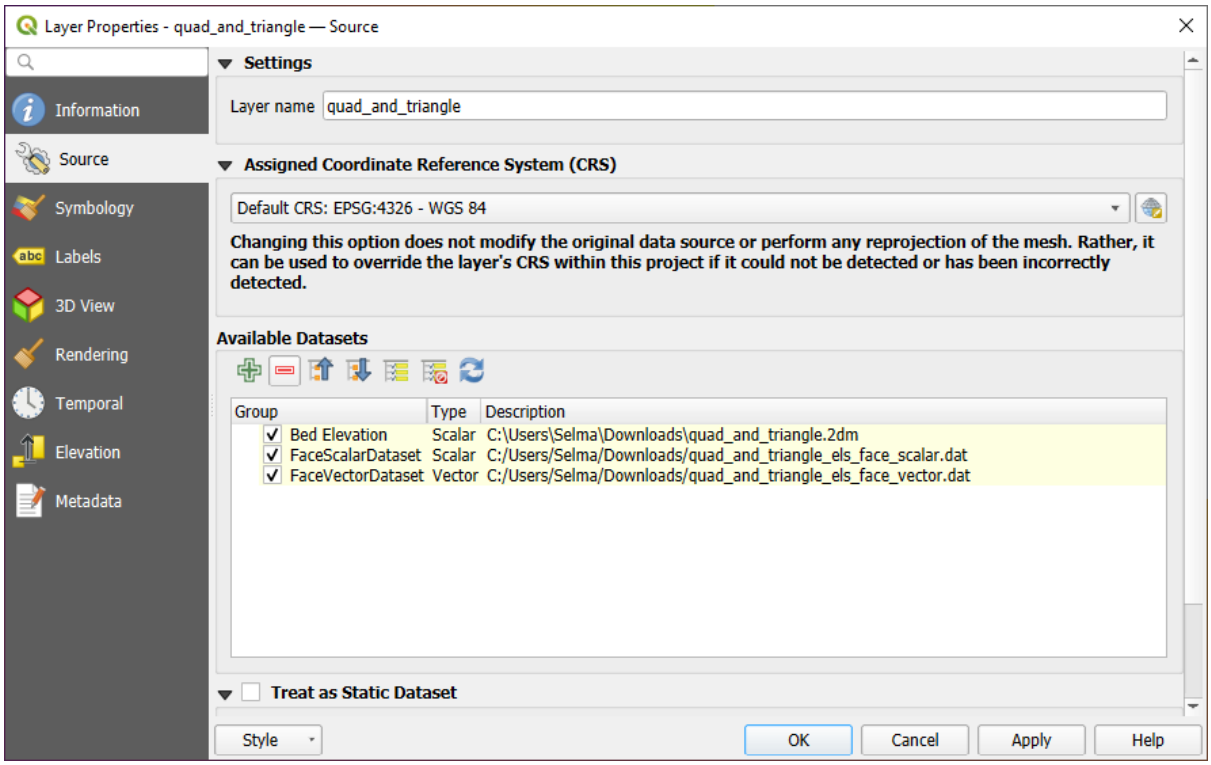


Abb. 14.5: Mesh Layer Source Properties

- the layer name to display in the *Layers* panel
- setting the Coordinate Reference System: Displays the layer's *Assigned Coordinate Reference System (CRS)*.

  You can change the layer's CRS by selecting a recently used one in the drop-down list or clicking on *Select CRS* button (see *Auswahl des Koordinatenbezugssystems*). Use this process only if the CRS applied to the layer is wrong or if none was applied.
- The *Available datasets* frame lists all the dataset groups (and subgroups) in the mesh layer, with their type and description in a tree view. Both regular datasets (i.e. their data is stored in the file) and virtual datasets (which are *calculated on the fly*) are listed.
  - 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
  - If you are interested in few datasets, you can uncheck the others and make them unavailable in the project
  - Double-click over a name and you can rename the dataset.
  - Reset to defaults: checks all the groups and renames them back to their original name in the provider.
  - Right-click over a virtual dataset group and you can:
    - *Remove dataset group* from the project

    * *Save dataset group as…* a file on disk, to any supported format. The new file is kept assigned to the current mesh layer in the project.

- Checking the *Treat as static dataset* group allows to ignore the *map temporal navigation* properties while rendering the mesh layer. For each active dataset group (as selected in *Symbology* ► *Datasets* tab), you can:
  - set to *None*: the dataset group is not displayed at all
  - *Display dataset*: e.g., for the „bed elevation“ dataset which is not time aware
  - extract a particular date time: the dataset matching the provided time is rendered and stays fixed during map navigation.

14.3.3 Symbolisierung

Click the *Symbology* button to activate the dialog. Symbology properties are divided into several tabs:

- *Datasets*
- *Contours*
- *Vectors*
- *Rendering*
- *Stacked mesh averaging method*

Datasets

The tab Datasets is the main place to control and set which datasets will be used for the layer. It presents the following items:

- *Groups* available in the mesh dataset, with whether they provide:
  - scalar dataset
  - or vector dataset: by default, each vector dataset has a scalar dataset representing its magnitude automatically generated.

  Click on the icon next to the dataset name to select the group and type of data to represent.
- *Selected dataset group(s) metadata*, with details on:
  - the mesh type: edges or faces
  - the data type: vertices, edges, faces or volume
  - whether it's of vector type or not
  - the original name in the mesh layer
  - the unit, if applicable
- *blending mode* available for the selected datasets.

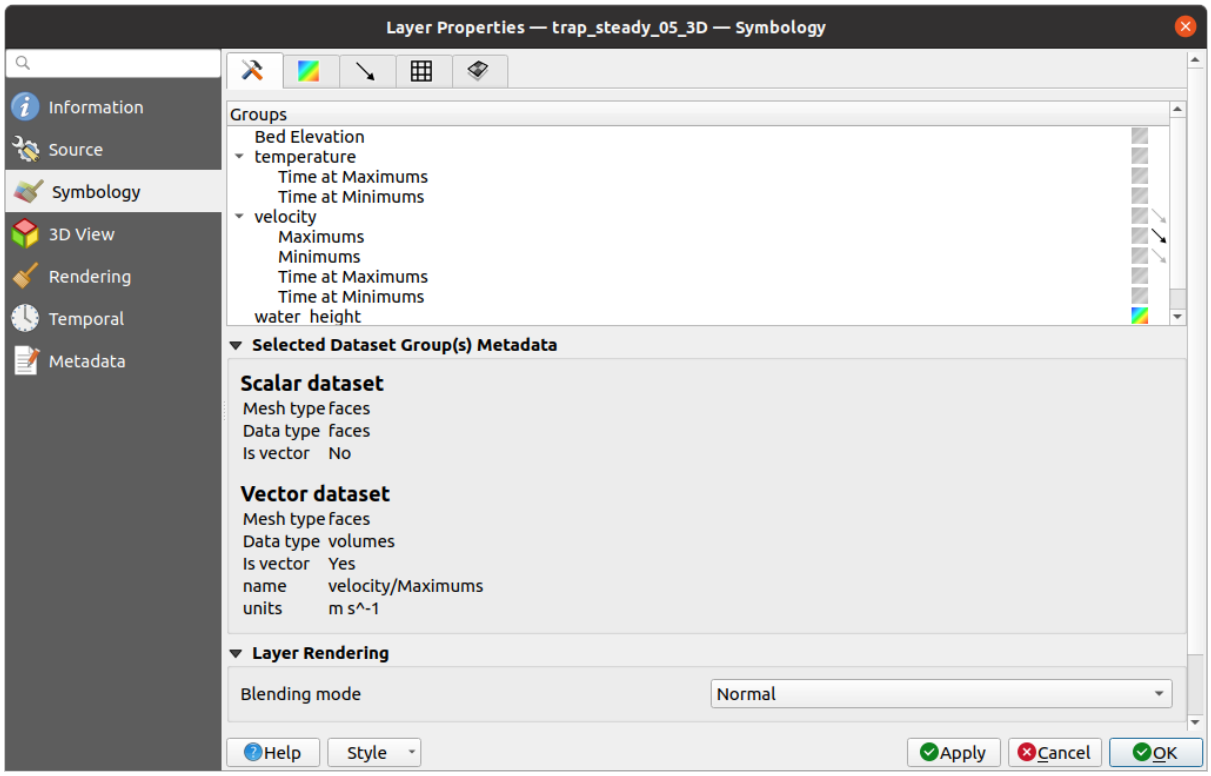


Abb. 14.6: Mesh Layer Datasets

You can apply symbology to the selected vector and/or scalar group using the next tabs.

Contours Symbology

Bemerkung

The Contours tab can be activated only if a scalar dataset has been selected in the *Datasets* tab.

In the Contours tab you can see and change the current visualization options of contours for the selected group, as shown in Abb. 14.7 below:

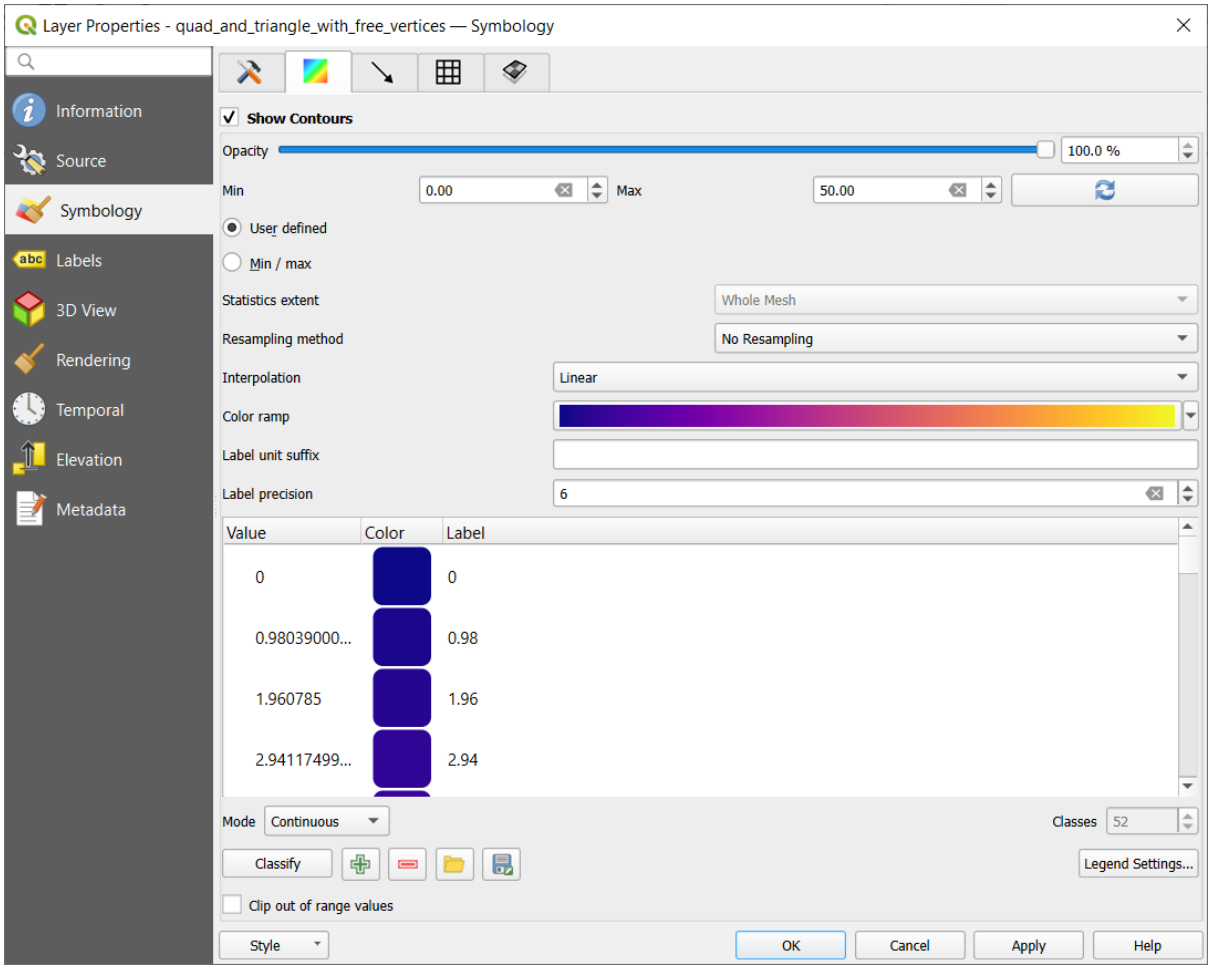


Abb. 14.7: Styling Contours in a Mesh Layer

- For 1D mesh, set the *Stroke width* of the edges. This can be a fixed size for the whole dataset, or vary along the geometry (more details with the *interpolated line renderer*)
- Use the slider or the spinbox to set the *Opacity* of the current group, if of a 2D mesh type.
- *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).
- For 2D/3D meshes, select the *Resampling method* to interpolate the values on the surrounding vertices to the faces (or from the surrounding faces to the vertices) using the *Neighbour average* method. Depending on whether the dataset is defined on the vertices (respectively on the faces), QGIS defaults this setting to *None* (respectively *Neighbour average*) method in order to use values on vertices and keep the default rendering smooth.
- Classify the dataset using the *color ramp shader* classification.

Vectors Symbology

Bemerkung

The Vectors tab can be activated only if a vector dataset has been selected in the *Datasets* tab. Click on the Vectors icon on the right side of the *Datasets* tab.

In the Vectors tab you can see and change the current visualization options of vectors for the selected group, as shown in Abb. 14.8:

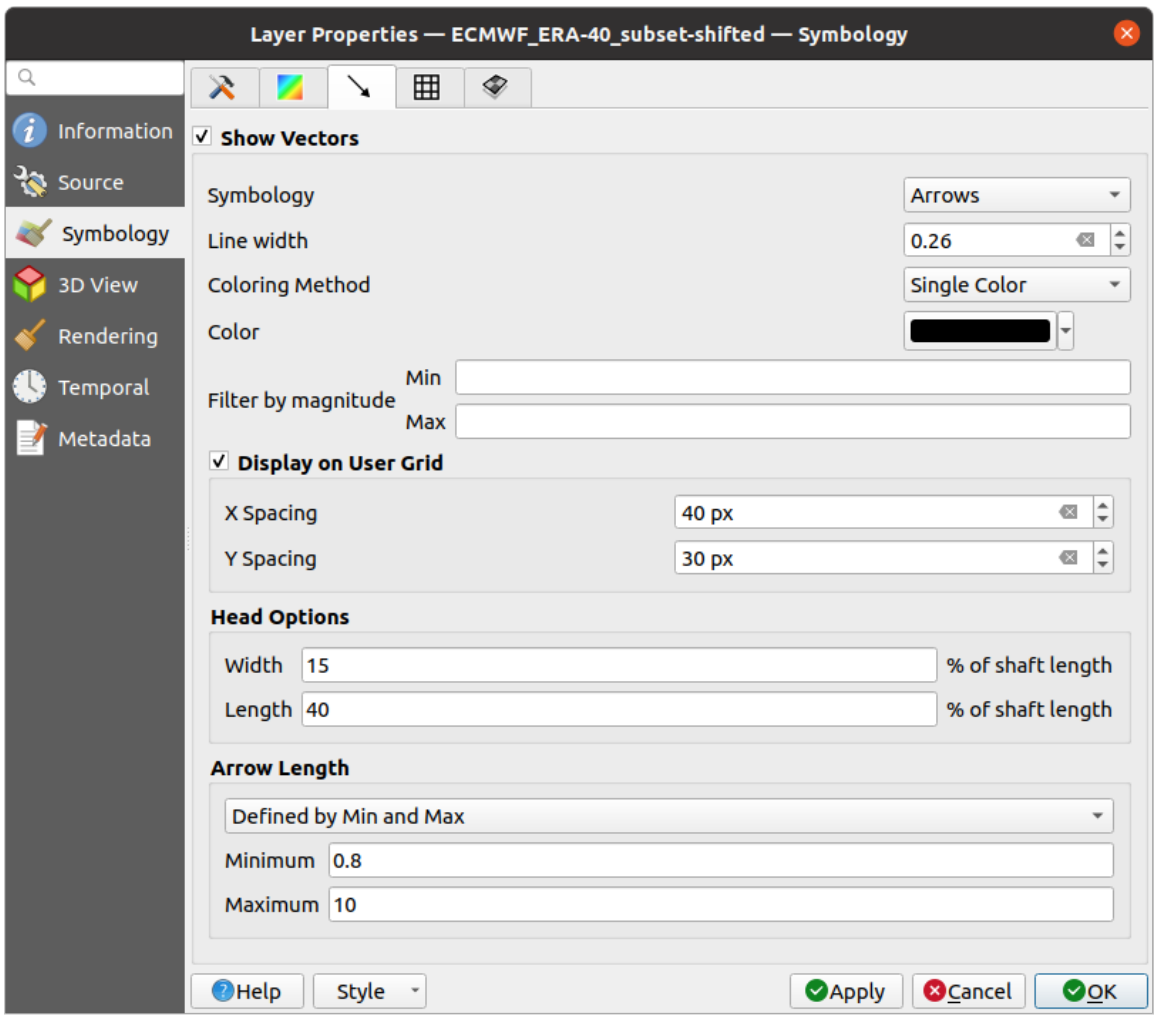


Abb. 14.8: Styling Vectors in a Mesh Layer with arrows

Mesh vector dataset can be styled using various types of *Symbology*:

- **Arrows**: vectors are represented with arrows at the same place as they are defined in the raw dataset (i.e. on the nodes or center of elements) or on a user-defined grid (hence, they are evenly distributed). The arrow length is proportional to the magnitude of the arrow as defined in the raw data but can be scaled by various methods.
- **Streamlines**: vectors are represented with streamlines seeded from start points. The seeding points can start from the vertices of the mesh, from a user grid or randomly.
- **Traces**: a nicer animation of the streamlines, the kind of effect you get when you randomly throws sand in the water and see where the sand items flows.

- **Wind barbs**: vectors are represented with wind barbs, a common way to represent wind speed and direction.

Available properties depend on the selected symbology as shown in the following table.

Tab. 14.3: Availability and meaning of the vectors symbology properties

Label, Description and Properties	Arrow	Streamlines	Traces	Wind Barbs
Line width: Width of the vector representation	✓	✓	✓	✓
Coloring method: • a *Single color* assigned to all vectors • or a variable color based on vectors magnitude, using a *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: • **Defined by Min and Max**: You specify the minimum and maximum length for the arrows, QGIS will interpolate their size based on the underlying vector's magnitude • **Scale to magnitude**: arrow length is proportional to their vector's magnitude • **Fixed**: all the vectors are shown with the same length	✓			
Streamlines seeding method: • **On mesh/grid**: relies on the user grid to display the vectors • **Randomly**: vector placement is randomly done with respect to a certain density		✓		
Particles count: The amount of „sand" you want to throw into visualisation			✓	
Max tail length: The time until the particle fades out			✓	
Wind Barbs: • **Length**: Specifies the length of the wind barbs. Choose the units from a drop-down menu. • **Data Units**: Specifies the units of wind speed data. Wind barbs encode the wind in knots (nautical miles per hour). Therefore, you should select the units that your data are in, or use a custom *Multiplier* to convert your data units to knots. This ensures that the wind speed is accurately represented.				✓

Darstellung

In the tab Rendering tab, QGIS offers possibilities to display and customize the mesh structure. *Line width* and *Line color* can be set to represent:

- the edges for 1D meshes
- For 2D meshes:
  - *Native mesh rendering*: shows original faces and edges from the layer
  - *Triangular mesh rendering*: adds more edges and displays the faces as triangles

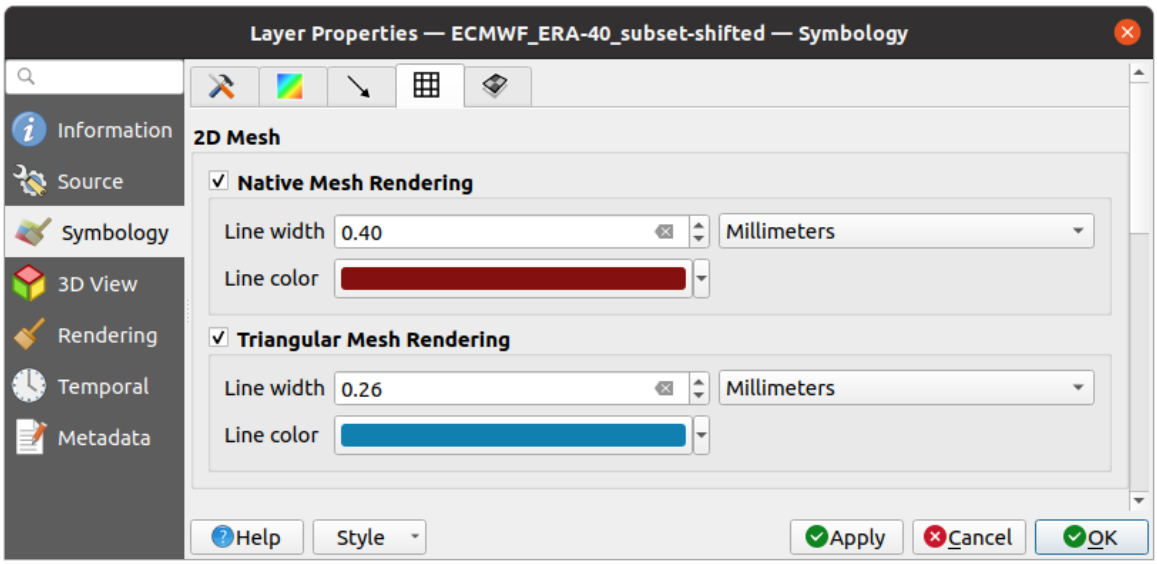


Abb. 14.9: 2D Mesh Rendering

Stacked mesh averaging method

3D layered meshes consist of multiple stacked 2D unstructured meshes each extruded in the vertical direction (`levels`) by means of a vertical coordinate. The vertices and faces have the same topology in each vertical level. Values are usually stored on the volumes that are regularly stacked over base 2d mesh. In order to visualise them on 2D canvas, you need to convert values on volumes (3d) to values on faces (2d) that can be shown in mesh layer. The Stacked mesh averaging method provides different averaging/interpolation methods to handle this.

You can select the method to derive the 2D datasets and corresponding parameters (level index, depth or height values). For each method, an example of application is shown in the dialog but you can read more on the methods at https://fvwiki.tuflow.com/Depth_Averaging_Results.

14.3.4 Beschriftungen Eigenschaften

The *Labels* tab offers you dynamic labeling of mesh vertices and native mesh faces based on geometric properties and custom expressions. This dialog can also be accessed from the *Layer Styling* panel.

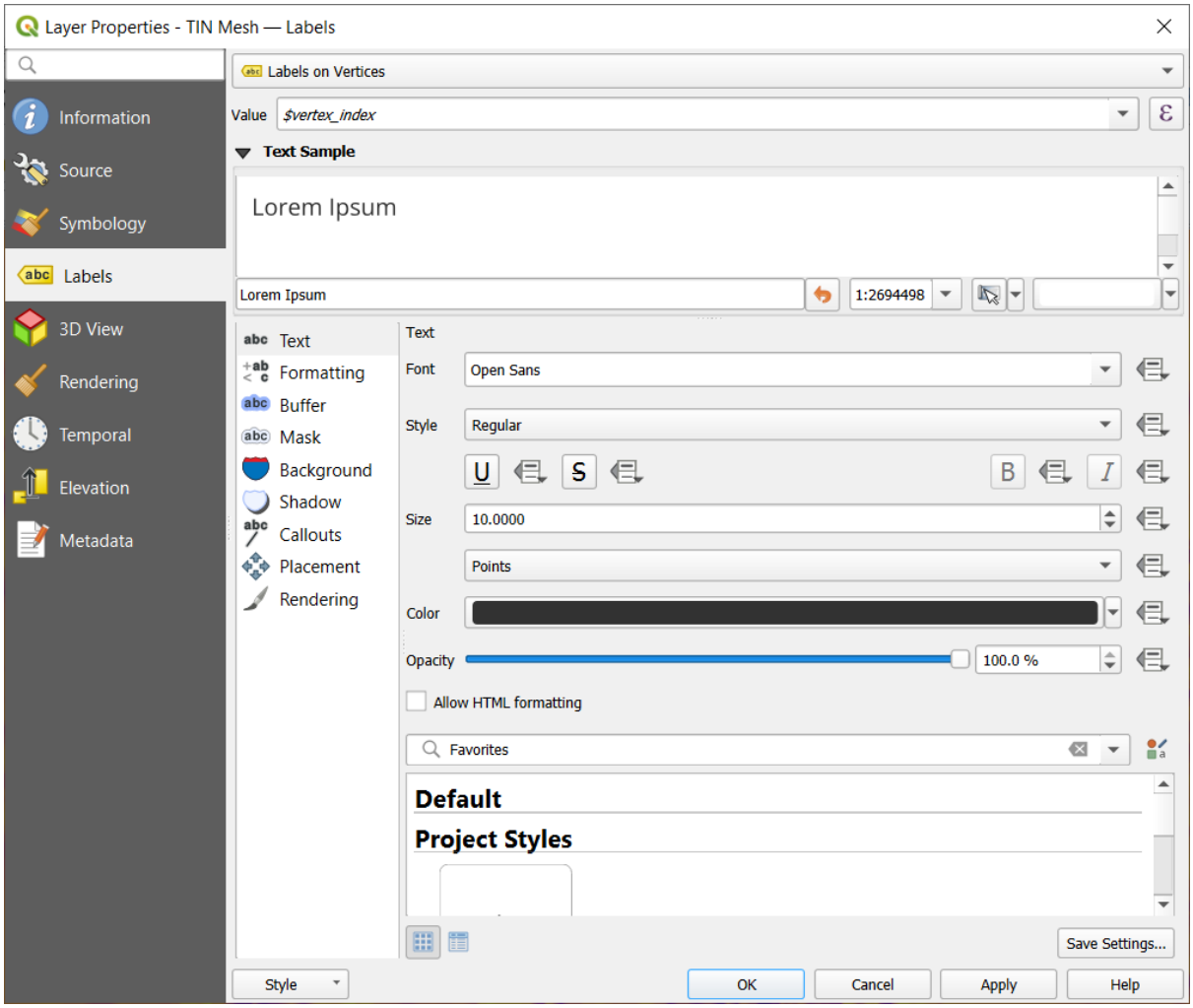


Abb. 14.10: Mesh Labels Properties

Choose the labeling method from the drop-down list. Available methods are:

- *No labels*: the default value, showing no labels from the layer.
- *Labels on vertices*: labels the mesh vertices as a point feature.
- *Labels on faces*: labels the mesh faces as a polygon feature.

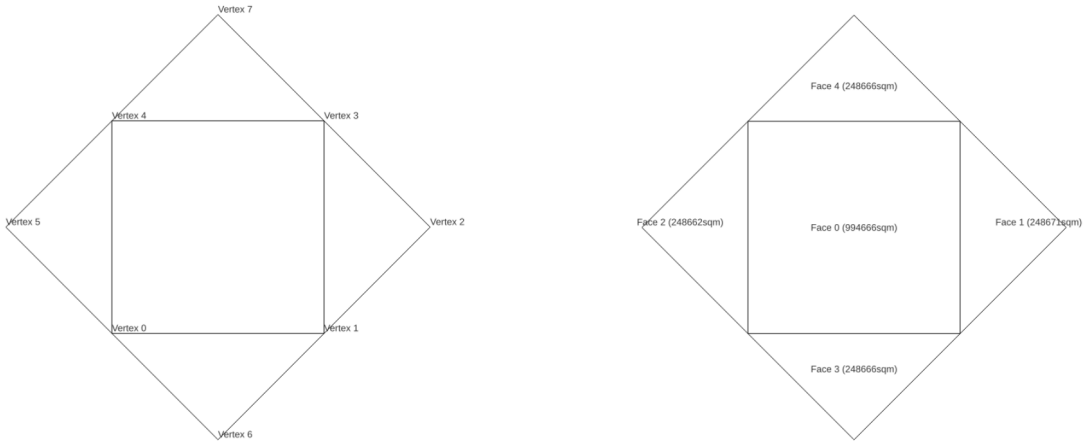


Abb. 14.11: Mesh Labeling methods

Once the labeling method has been selected, open Expression dialog and choose a *mesh function*. You can combine meshes functions with other functions from the *list* to modify the label.

Im Folgenden werden unter verschiedenen Registerkarten Optionen zur Anpassung der Etiketten angezeigt:

- *Text*
- *Formatierung*
- *Puffer*
- *Maske*
- *Hintergrund*
- *Schatten*
- *Führungslinien*
- *Platzierung*
- *Darstellung*

Eine Beschreibung, wie die einzelnen Eigenschaften einzustellen sind, findet sich unter *Einrichten einer Beschriftung*.

14.3.5 3D-Ansicht Eigenschaften

Mesh layers can be used as *terrain in a 3D map view* based on their vertices Z values. From the *3D View* properties tab, it's also possible to render the mesh layer's dataset in the same 3D view. Therefore, the vertical component of the vertices can be set equal to dataset values (for example, level of water surface) and the texture of the mesh can be set to render other dataset values with color shading (for example velocity).

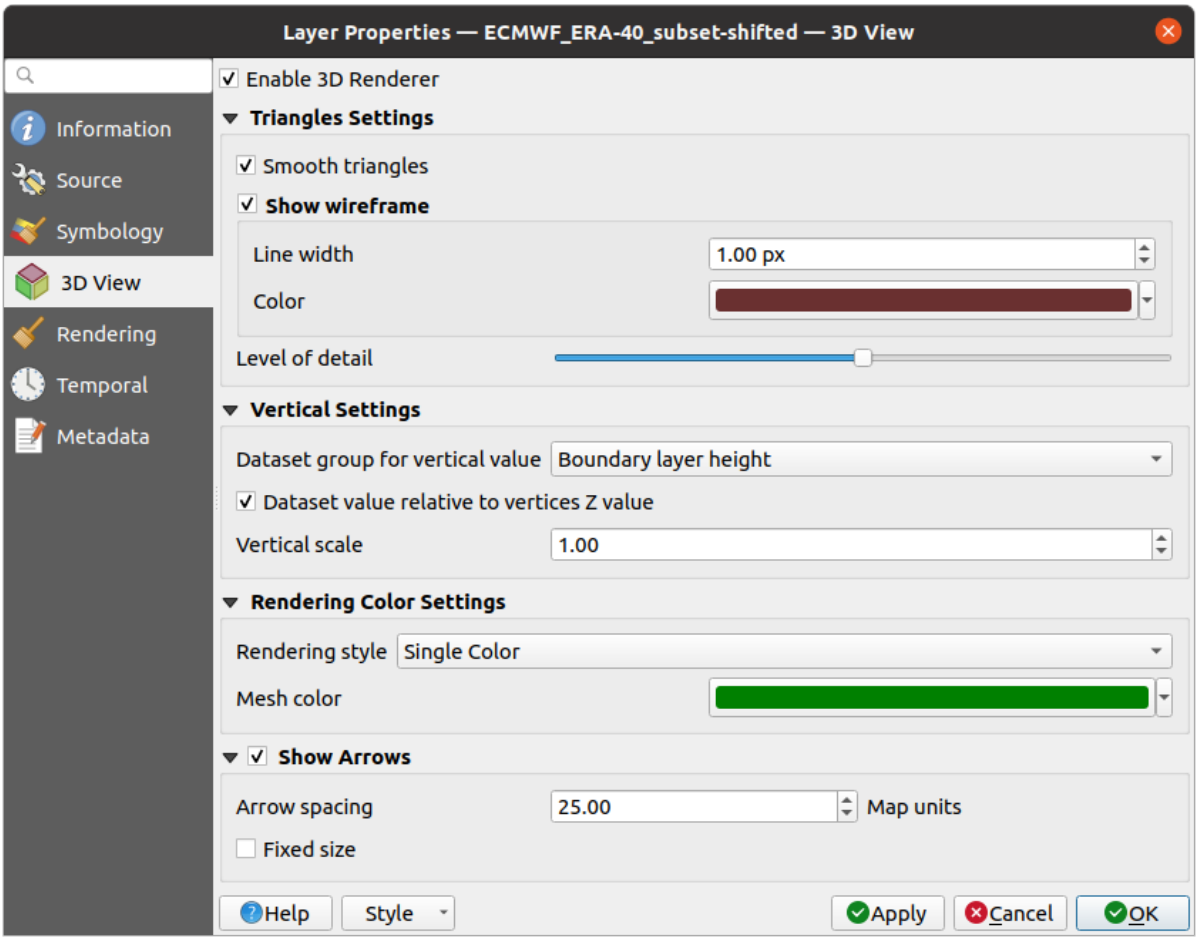


Abb. 14.12: Mesh dataset 3D properties

Check *Enable 3D Renderer* and you can edit following properties:

- Under *Triangle settings*
  - *Smooth triangles*: Angles between consecutive triangles are smoothed for a better 3D rendering
  - *Show wireframe* whose you can set the *Line width* and *Color*
  - *Level of detail*: Controls how *simplified* the mesh layer to render should be. On the far right, it is the base mesh, and the more you go left, the more the layer is simplified and is rendered with less details. This option is only available if the *Simplify mesh* option under the *Rendering* tab is activated.
- *Vertical settings* to control behavior of the vertical component of vertices of rendered triangles:
  - *Dataset group for vertical value*: the dataset group that will be used for the vertical component of the mesh
  - *Dataset value relative to vertices Z value*: whether to consider the dataset values as absolute Z coordinate or relative to the vertices native Z value
  - *Vertical scale*: the scale factor to apply to the dataset Z values
- *Rendering color settings* with a *Rendering style* that can be based on the color ramp shader set in *Contours Symbology* (*2D contour color ramp shader*) or as a *Single color* with an associated *Mesh color*
- *Show arrows*: displays arrows on mesh layer dataset 3D entity, based on the same vector dataset group used in the *vector 2D rendering*. They are displayed using the 2D color setting. It's also possible to define the *Arrow spacing* and, if it's of a *Fixed size* or scaled on magnitude. This spacing setting defines also the max size of arrows because arrows can't overlap.

14.3.6 Darstellung

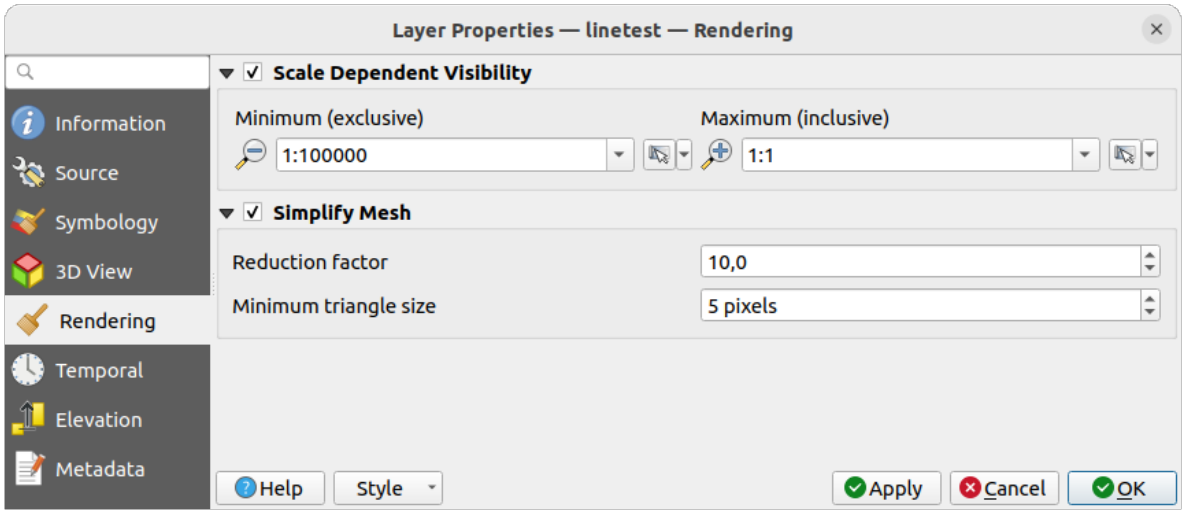


Abb. 14.13: Mesh rendering properties

Under the *Scale dependent visibility* group box, you can set the *Maximum (inclusive)* and *Minimum (exclusive)* scale, defining a range of scale in which mesh elements will be visible. Out of this range, they are hidden. The
Set to current canvas scale button helps you use the current map canvas scale as boundary of the range visibility. See *Einstellen des Sichtbarkeitsmaßstabs* for more information.

Bemerkung

Sie können die skalierungsabhängige Sichtbarkeit eines Layers auch über das Bedienfeld *Layer* aktivieren: Klicken Sie mit der rechten Maustaste auf den Layer und wählen Sie im Kontextmenü *Maßstabsabhängige Sichtbarkeit des Layers setzen*.

As mesh layers can have millions of faces, their rendering can sometimes be very slow, especially when all the faces are displayed in the view whereas they are too small to be viewed. To speed up the rendering, you can simplify the mesh layer, resulting in one or more meshes representing different *levels of detail* and select at which level of detail you would like QGIS to render the mesh layer. Note that the simplify mesh contains only triangular faces.

From the *Rendering* tab, check *Simplify mesh* and set:

- a *Reduction factor*: Controls generation of successive levels of simplified meshes. For example, if the base mesh has 5M faces, and the reduction factor is 10, the first simplified mesh will have approximately 500 000 faces, the second 50 000 faces, the third 5000,… If a higher reduction factor leads quickly to simpler meshes (i.e. with triangles of bigger size), it produces also fewer levels of detail.
- *Minimum triangle size*: the average size (in pixels) of the triangles that is permitted to display. If the average size of the mesh is lesser than this value, the rendering of a lower level of details mesh is triggered.

14.3.7 Eigenschaften Zeitlich

The *Temporal* tab provides options to control the rendering of the layer over time. It allows to dynamically display temporal values of the enabled dataset groups. Such a dynamic rendering requires the *temporal navigation* to be enabled over the map canvas.

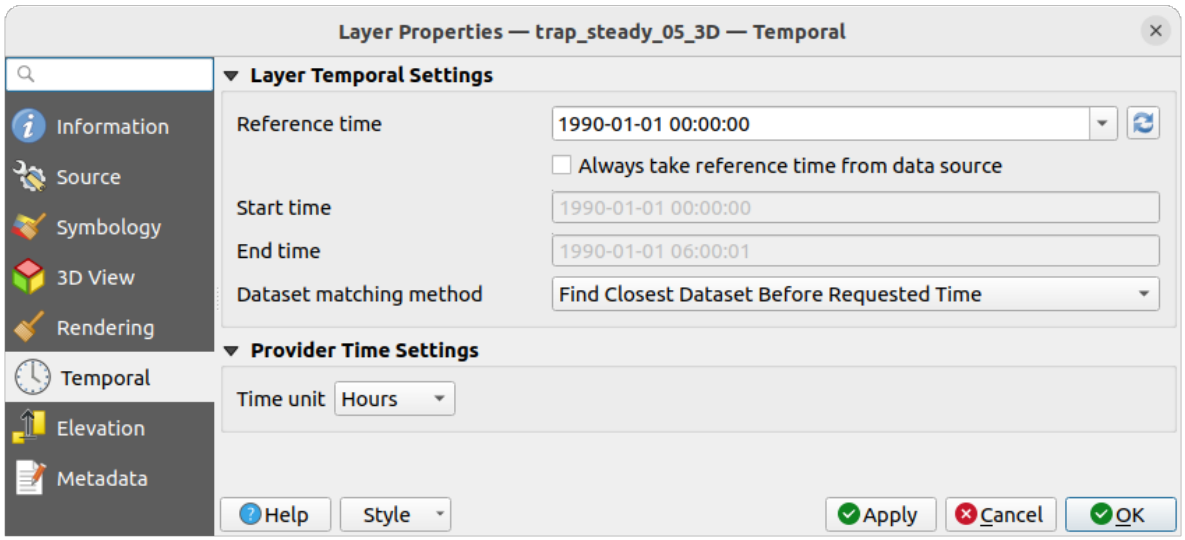


Abb. 14.14: Mesh Temporal properties

Layer temporal settings

- *Reference time* of the dataset group, as an absolute date time. By default, QGIS parses the source layer and returns the first valid reference time in the layer's dataset group. If unavailable, the value will be set by the project time range or fall back to the current date. The *Start time* and *End time* to consider are then calculated based on the internal timestamp step of the dataset.

  It is possible to set a custom *Reference time* (and then the time range), and revert the changes using the Reload from provider button. With *Always take reference time from data source* checked, you ensure that the time properties are updated from the file each time the layer is reloaded or the project reopened.

- *Dataset matching method*: determines the dataset to display at the given time. Options are *Find closest dataset before requested time* or *Find closest dataset from requested time (after or before)*.

Provider time settings

- *Time unit* extracted from the raw data, or user defined. This can be used to align the speed of the mesh layer with other layers in the project during map time navigation. Supported units are *Seconds*, *Minutes*, *Hours* and *Days*.

14.3.8 Höhe

The *Elevation* tab provides options to control the layer elevation properties within a *3D map view* and *2D map view* and its appearance in the *profile tool charts*. Specifically, you can configure how heights from your dataset are interpreted:

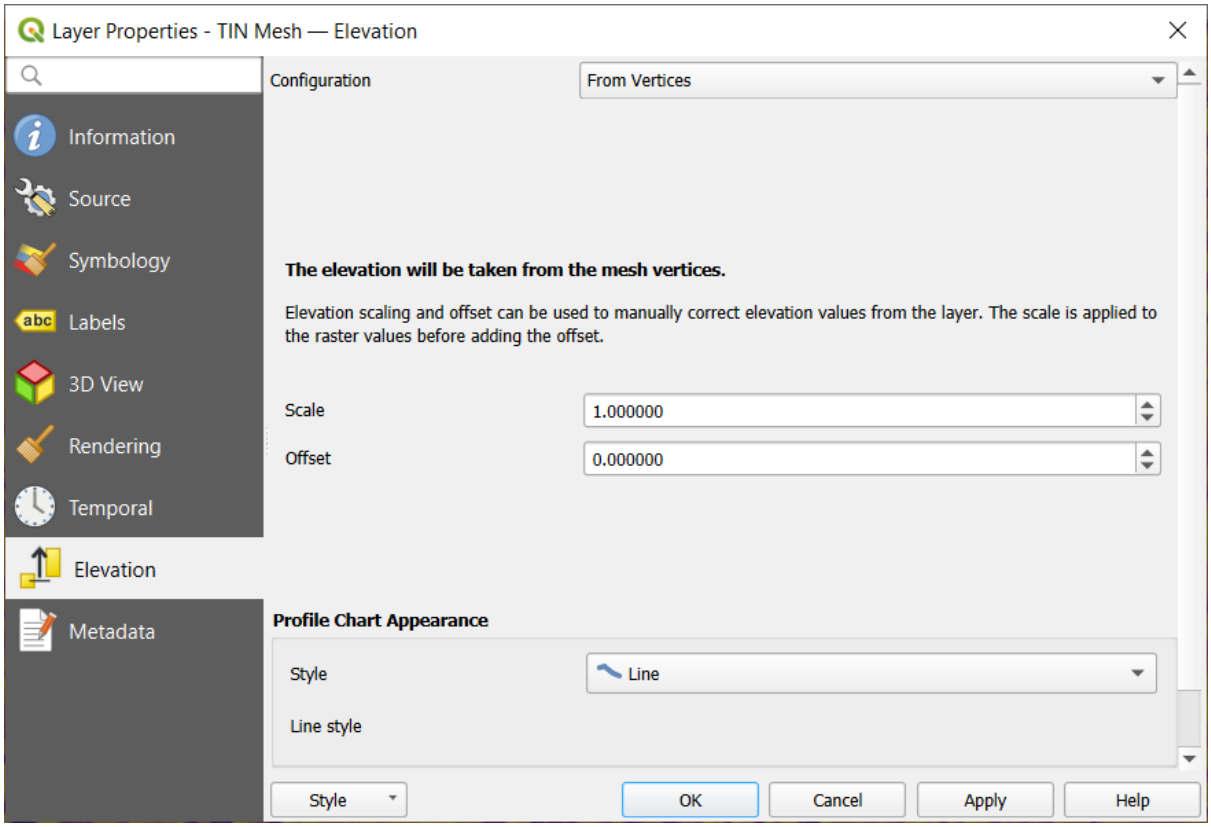


Abb. 14.15: Mesh Elevation properties

- *From vertices*: how the mesh layer vertices Z values should be interpreted as terrain elevation. You can apply a *Scale* factor and an *Offset*. This setting is available for *3D map view* and *profile tool charts*.
- *Fixed Elevation Range*: the mesh layer is linked to a fixed elevation range. This mode is applicable when a layer has either a single fixed elevation or a range (slice) of elevation values. If a range is specified, mesh values will be extruded over this range. You can set the *Lower* and *Upper* elevation range values for the layer, and specify whether the lower or upper *Limits* are inclusive or exclusive. When enabled, the layer will only be visible in *elevation filtered 2D maps* when the layer's range is included in the map's Z range.
- *Fixed Elevation Range Per Group*: each group in the mesh layer is associated with a fixed elevation range. This mode can be used when a layer has elevation data exposed through different dataset groups. This feature is exposed as a user-editable table for dataset groups with lower and upper values. You can either populate the lower and upper values manually or use an ε *Expression* to auto-fill all group values based on an expression. When enabled, the layer will be filtered in the *2D map view*, displaying only values within the map filter ranges. In *3D map view* or *profile tool charts*, the full extent of the layer will be displayed, ignoring the filtering.
- *Profile Chart Appearance*: controls the rendering of the mesh elements elevation in the profile chart. The profile *Style* can be set as:
  - eine *Linie* mit einem bestimmten *Linenstil*
  - eine Höhenfläche, die mit einem Füllsymbol entweder über (*Oberhalb füllen*) oder unter (*Unten füllen*) der Höhenkurvenlinie dargestellt wird. Die Oberflächensymbolik wird mit dargestellt:
    - einem *Füllstil*
    - und einer *Begrenzung*: die maximale (bzw. minimale) Höhe, die bestimmt, wie hoch die Füllfläche sein wird

14.3.9 Metadaten

Der Reiter *Metadata* bietet Ihnen Optionen zum Erstellen und Bearbeiten eines Metadatenberichts zu Ihrem Layer. Siehe *Metadaten* für weitere Informationen.

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).

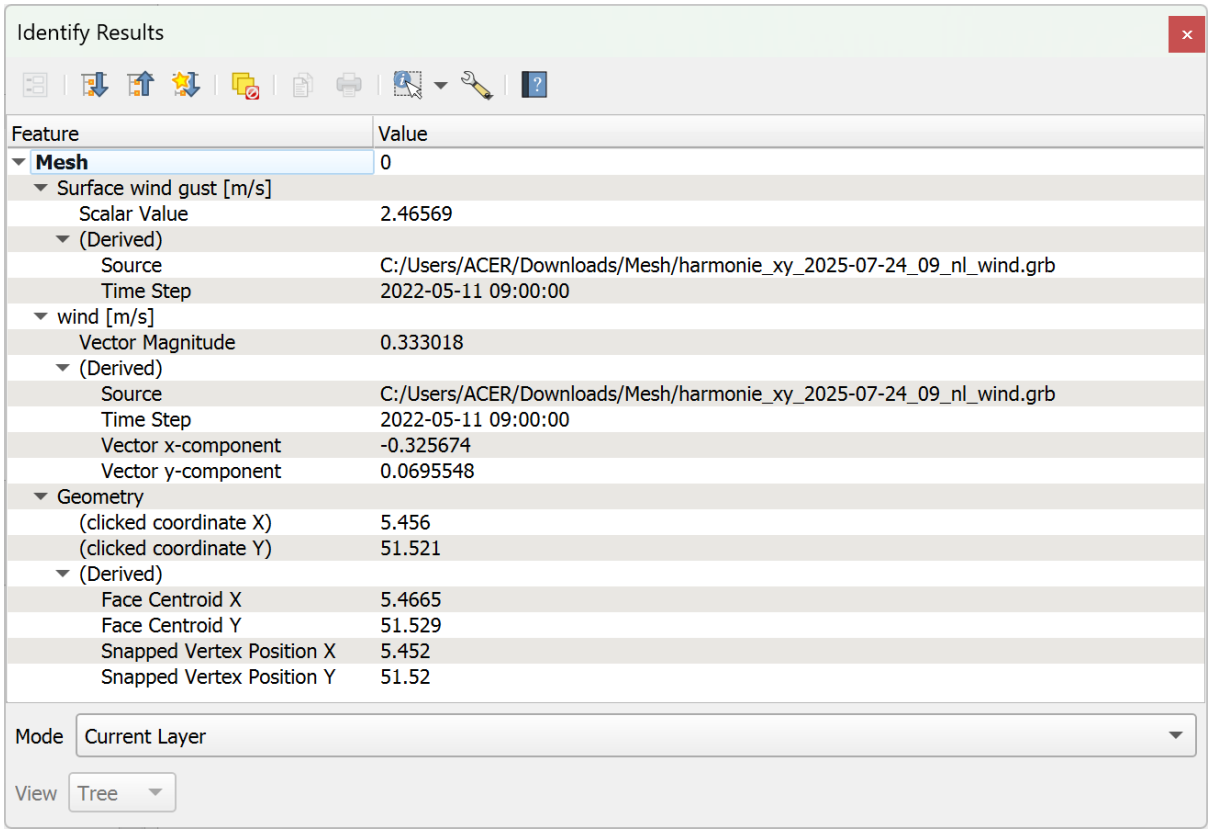


Abb. 14.16: Mesh Identify Results panel

14.5 Editing a mesh layer

QGIS allows to *create a mesh layer* from scratch or based on an existing one. You can create/modify the geometries of the new layer whom you can assign datasets afterwards. It's also possible to edit an existing mesh layer. Because the editing operation requires a frames-only layer, you will be asked to either remove any associated datasets first (make sure you have them available if they still are necessary) or create a copy (only geometries) of the layer.

Bemerkung

QGIS does not allow to digitize edges on mesh layers. Only vertices and faces are mesh elements that can be created. Also not all supported mesh formats can be edited in QGIS (see permissions).

14.5.1 Overview of the mesh digitizing tools

To interact with or edit a base mesh layer element, following tools are available.

Tab. 14.4: Tools for mesh digitizing

Label	Zweck	Location
Current Edits	Änderungen in allen ausgewählten Layern gleichzeitig speichern, rückgängig machen oder abbrechen	*Digitizing* toolbar
Toggle to Edit	Turn on/off the layer in edit mode	*Digitizing* toolbar
Save Edits	Save changes done to the layer	*Digitizing* toolbar
Undo	Undo the last change(s) - `Ctrl+Z`	*Digitizing* toolbar
Redo	Redo the last undone action(s) - `Ctrl+Shift+Z`	*Digitizing* toolbar
Enable Advanced Digitizing toc	Turn on/off the *Advanced Digitizing Panel*	*Advanced Digitizing* toolbar
Reindex Faces and Vertices	Recreate index and renumber the mesh elements for optimization	*Mesh* menu
Digitize Mesh Elements	Select/Create vertices and faces	*Mesh* *Digitizing* toolbar
Select Mesh Elements by Polyg	Select vertices and faces overlapped by a drawn polygon	*Mesh* *Digitizing* toolbar
Select Mesh Elements by Expre	Select vertices and faces using an expression	*Mesh* *Digitizing* toolbar
Transform Vertices Coordinate	Modify coordinates of a selection of vertices	*Mesh* *Digitizing* toolbar
Force by Selected Geometries	Split faces and constrain Z value using a linear geometry	*Mesh* *Digitizing* toolbar

14.5.2 Exploring the Z value assignment logic

When a mesh layer is turned into edit mode, a *Vertex Z value* widget opens at the top right of the map canvas. By default, its value corresponds to the *Default Z value* set in *Settings* ► *Options* ► *Digitizing* tab. When there are selected vertices, the widget displays the average Z value of the selected vertices.

During editing, the *Vertex Z value* is assigned to new vertices. It is also possible to set a custom value: edit the widget, press `Enter` and you will override the default value and make use of this new value in the digitizing process. Click the icon in the widget to reset its value to the Options default value.

Rules of assignment

This is configured in the drop-down menu of the *Digitize mesh elements* tool as *New vertex Z value* option. Following methods are available:

Tab. 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.

Tab. 14.6: Matrix of Z value assignment to new vertex

Vertex creation	Vertices selected?	Source of assigned value	Assigned Z Value
„Free“ vertex, not connected to any face or edge of a face	Nein	*Vertex Z value*	Default or user defined
		Advanced Digitizing Panel (if *z* widget is in Locked state)	*z* widget if in Locked state
	Ja	*Vertex Z value*	Average of the selected vertices or user-defined
Vertex on an edge	—	Mesh layer	Interpolated from the edge’s vertices
Vertex on a face	—	Mesh layer	Interpolated from the face’s vertices
Vertex snapped to a 2D vector feature	—	*Vertex Z value*	Default or user defined
Vertex snapped to a 3D vector vertex	—	Vector layer	Vertex
Vertex snapped to a 3D vector segment	—	Vector layer	Interpolated along the vector segment

Bemerkung

The *Vertex Z value* widget is deactivated if the *Advanced Digitizing Panel* is enabled and no mesh element is selected. The latter’s *z* widget then rules the Z value assignment.

Modifying Z value of existing vertices

To modify the Z value of vertices, the most straightforward way is:

1. Select one or many vertices. The *Vertex Z value* widget will display the average height of the selection.
2. Change the value in the widget.
3. Press `Enter`. The entered value is assigned to the vertices and becomes the default value of next vertices.

Another way to change the Z value of a vertex is to move and snap it on a vector layer feature with the Z value capability. If more than one vertex are selected, the Z value can't be changed in this way.

The *Transform mesh vertices* dialog also provides means to modify the Z value of a selection of vertices (along with their X or Y coordinates).

14.5.3 Selecting mesh elements

To select mesh elements, you can use the following tools:

- Digitize Mesh Elements to select different mesh elements, see more at *Using Digitize Mesh Elements*.
- Select Mesh Elements by Polygon to select different mesh elements by polygon, see more at *Using Select Mesh Elements by Polygon*.
- 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 *Using Select Mesh Elements by Expression*.
- 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 *Digitize Mesh Elements*

Activate the Digitize Mesh Elements tool. Hover over an element and it gets highlighted, allowing you to select it.

- Click on a vertex, and it is selected.
- Click on the small square at the center of a face or an edge, and it gets selected. Connected vertices are also selected. Conversely, selecting all the vertices of an edge or a face also selects that element.
- Drag a rectangle to select overlapping elements (a selected face comes with all their vertices). Press `Alt` key if you want to select only completely contained elements.
- To add elements to a selection, press `Shift` while selecting them.
- To remove an element from the selection, press `Ctrl` and reselect it. A deselected face will also deselect all their vertices.

Using *Select Mesh Elements by Polygon*

Activate the Select Mesh Elements by Polygon tool and:

- Draw a polygon (left-click to add vertex, `Backspace` to undo last vertex, `Esc` to abort the polygon and right-click to validate it) over the mesh geometries. Any partially overlapping vertices and faces will get selected. Press `Alt` key while drawing if you want to select only completely contained elements.
- Right-click over the geometry of a vector layer's feature, select it in the list that pops up and any partially overlapping vertices and faces of the mesh layer will get selected. Use `Alt` while drawing to select only completely contained elements.

- To add elements to a selection, press `Shift` while selecting them.
- To remove an element from the selection, press `Ctrl` while drawing over the selection polygon.

Using *Select Mesh Elements by Expression*

Another tool for mesh elements selection is Select Mesh Elements by Expression. When pressed, the tool opens the mesh *expression selector dialog* from which you can:

1. Select the method of selection:
   - *Select by vertices*: applies the entered expression to vertices, and returns matching ones and their eventually associated edges/faces
   - *Select by faces*: applies the entered expression to faces, and returns matching ones and their associated edges/vertices
2. Write the expression of selection. Depending on the selected method, available functions in the *Meshes group* will be filtered accordingly.
3. Run the query by setting how the selection should behave and pressing:
   - *Select*: replaces any existing selection in the layer
   - *Add to current selection*
   - *Remove from current selection*

14.5.4 Modifying mesh elements

Adding vertices

To add vertices to a mesh layer:

1. Press the Digitize mesh elements button
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. Then double-click:
   - outside a face: adds a „free vertex“, that is a vertex not linked to any face. This vertex is represented by a red dot when the layer is in editing mode.
   - on the edge of existing face(s): adds a vertex on the edge, splits the touching face(s) into triangles connected to the new vertex.
   - inside a face: splits the face into triangles whose edges connect the surrounding vertices to the new vertex.

     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.

Adding faces

To add faces to a mesh layer:

1. Press the Digitize mesh elements button
2. A *Vertex Z value* widget appears on the top right corner of the map canvas. Set this value to the Z coordinate you would like to assign to the subsequent vertices.
3. Hover over a vertex and click the small triangle that appears next to it.
4. Move the cursor to the next vertex position; you can snap to existing vertex or left-click to add a new one.
5. Proceed as above to add as many vertices as you wish for the face. Press `Backspace` button to undo the last vertex.
6. While moving the mouse, a rubberband showing the shape of the face is displayed. If it is shown in green, then the expected face is valid and you can right-click to add it to the mesh. If it is red, the face is not valid (e.g. because it self-intersects, overlaps an existing face or vertex, creates a hole, …) and can't be added. You'd need to undo some vertices and fix the geometry.
7. Press `Esc` to abort the face digitizing.
8. Right-click to validate the face.

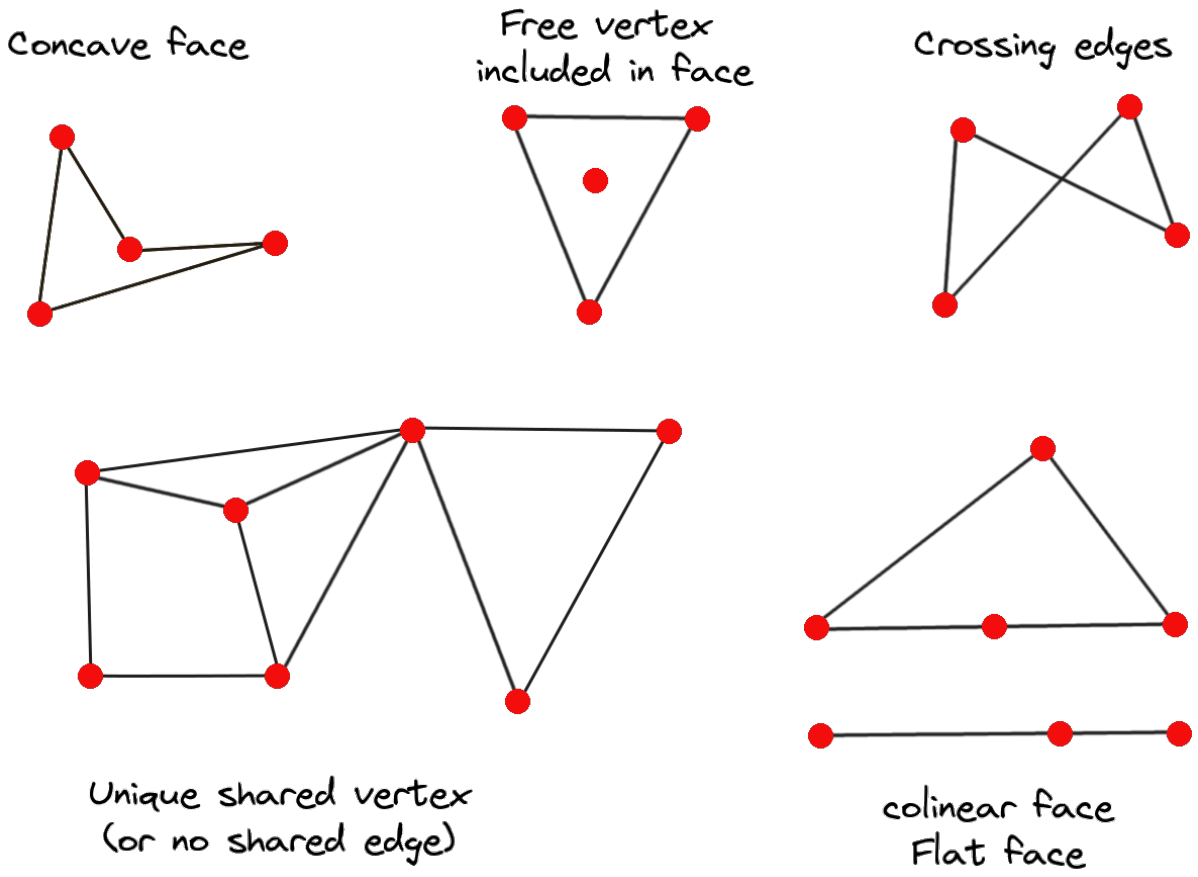


Abb. 14.17: Examples of invalid mesh

Removing mesh elements

1. *Select the target elements*
2. Enable the Digitize mesh elements tool
3. Right-click and select:
   - *Remove Selected Vertices and Fill Hole(s)* or press `Ctrl+Del`: removes vertices and linked faces and fills the hole(s) by triangulating from the neighbor vertices
   - *Remove Selected Vertices Without Filling Hole(s)* or press `Ctrl+Shift+Del`: removes vertices and linked faces and do not fill hole(s)
   - *Remove Selected Face(s)* or press `Shift+Del`: removes faces but keeps the vertices

These options are also accessible from the contextual menu when hovering over a single item without selecting.

Moving mesh elements

To move vertices and faces of a mesh layer:

1. *Select the target elements*
2. Enable the Digitize mesh elements tool
3. To start moving the element, click on a vertex or the centroid of a face/edge
4. Move the cursor to the target location (snapping to vector features is supported).
5. If the new location does not generate an *invalid mesh*, the moved elements appear in green. Click again to release them at this location. Faces whose vertices are all selected are translated, their neighbors are reshaped accordingly.

Transforming mesh vertices

The Transform Vertices Coordinates tool gives a more advanced way to move vertices, by editing their X, Y and/or Z coordinates thanks to expressions.

1. Select the vertices you want to edit the coordinates
2. Press Transform Vertices Coordinates. A dialog opens with a mention of the number of selected vertices. You can still add or remove vertices from the selection.
3. Depending on the properties you want to modify, you need to check the *X coordinate*, *Y coordinate* and/or *Z value*.
4. Then enter the target position in the box, either as a numeric value or an expression (using the Expression dialog)
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 *Preview Transform* to simulate the vertices new location and preview the mesh with transformation.
   - If the preview is green, transformed mesh is valid and you can apply the transformation.
   - If the preview is red, the transformed mesh is invalid and you can not apply the transformation until it is corrected.
7. Press *Apply Transform* to modify the selected coordinates for the set of vertices.

Reshaping mesh geometry

The contextual menu

1. Enable the Digitize mesh elements
2. Select mesh item(s), or not
3. Hover over a mesh element, it gets highlighted.
4. Right-click and you can:
   - *remove the item(s)*
   - *Split Selected Face(s)* (*Split Current Face*): splits the face you are hovering over or each selected quad mesh faces into two triangles
   - *Delaunay Triangulation with Selected vertices*: builds triangular faces using selected free vertices.
   - *Refine Selected Face(s)* (*Refine Current Face*): splits the face into four faces, based on vertices added at the middle of each edge (a triangle results into triangles, a quad into quads). Also triangulates adjacent faces connected to the new vertices.

The edge markers

When the Digitize mesh elements is active and you hover over an edge, the edge is highlighted and it is possible to interact with it. Depending on the context, following markers may be available:

- a **square**, at the center of the edge: click on it to select extremity vertices.
- a **cross** if the two faces on either side can be merged: click on it to delete the edge and merge the faces.
- a **circle** if the edge is between two triangles: Click on it to flip the edge, i.e. connect it instead to the two other „free“ vertices of the faces

The *Force by Selected Geometries* tool

The Force by Selected Geometries tool provides advanced ways to apply break lines using lines geometry. A break line will force the mesh to have edges along the line. Note that the break line will not be considered persistent once the operation is done; resulting edges will not act as constraints anymore and can be modified like any other edge. This can be used for example to locally modify a mesh layer with accurate lines, as river banks or border of road embankments.

1. Enable the Force by Selected Geometries tool
2. Indicate the geometry to use as „forcing line“; it can be:
   - picked from a line or polygon feature in the map canvas: right-click over the vector feature and select it from the list in the contextual menu.
   - a virtual line drawn over the mesh frame: left-click to add vertices, right-click for validation. Vertices Z value is set through the *Vertex Z value* widget or the *z* widget if the *Advanced Digitizing Panel* is on. If the line is snapped to a mesh vertex or a 3D vector feature's vertex or segment, the new vertex takes the snapped element Z value.

Mesh faces that overlap the line geometry or the polygon's boundary will be affected in a way that depends on options you can set from the Force by Selected Geometries tool drop-down menu:

- *Add new vertex on intersecting edges*: with this option, a new vertex is added each time the forcing line intersect an edge. This option leads to split along the line each encountered faces.

Without this option, encountered faces are removed and replaced by faces coming from a triangulation with only the existing vertices plus the vertices of the forcing lines (new vertices are also added on the boundary edge intersecting the forcing lines).

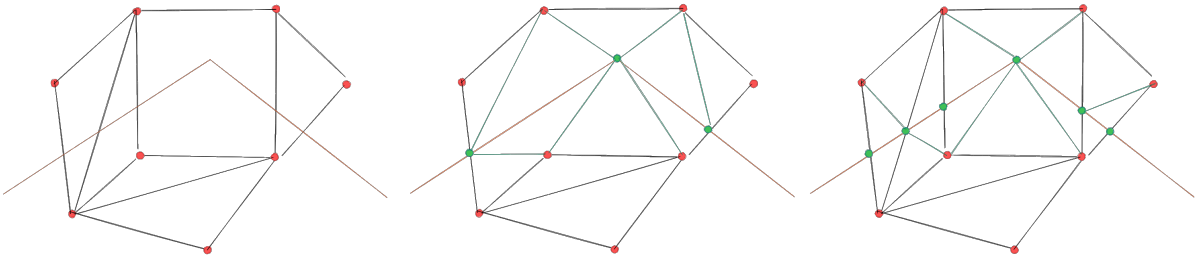


Abb. 14.18: Force Mesh using a line geometry - Results without (middle) and with (right) new vertex on edges intersection

- *Interpolate Z value from*: set how the new vertices Z value is calculated. It can be from:
  - the *Mesh* itself: the new vertices Z value is interpolated from vertices of the face they fall within
  - or the *Forcing line*: if the line is defined by a 3D vector feature or a drawn line then the new vertices Z value is derived from its geometry. In case of 2D line feature, the new vertices Z value is the *Vertex Z value*.
- *Tolerance*: when an existing mesh vertex is closer to the line than the tolerance value, do not create new vertex on the line but use the existing vertex instead. The value can be set in *Meters at Scale* or in *Map Units* (more details at *Einheiten Auswahl*).

14.5.5 Reindexing meshes

During edit, and in order to allow quick undo/redo operations, QGIS keeps empty places for deleted elements, which may lead to growing memory use and inefficient mesh structuring. The *Mesh* ► *Reindex Faces and Vertices* tool is designed to remove these holes and renumber the indices of faces and vertices so that they are continuous and somewhat reasonably ordered. This optimizes relation between faces and vertices and increases the efficiency of calculation.

Bemerkung

The *Reindex Faces and Vertices* tool saves the layer and clear the undo/redo stacks, disabling any rollback.

14.6 Mesh Calculator

The *Mesh Calculator* tool from the top *Mesh* menu allows you to perform arithmetic and logical calculations on existing dataset groups to generate a new dataset group (see Abb. 14.19).

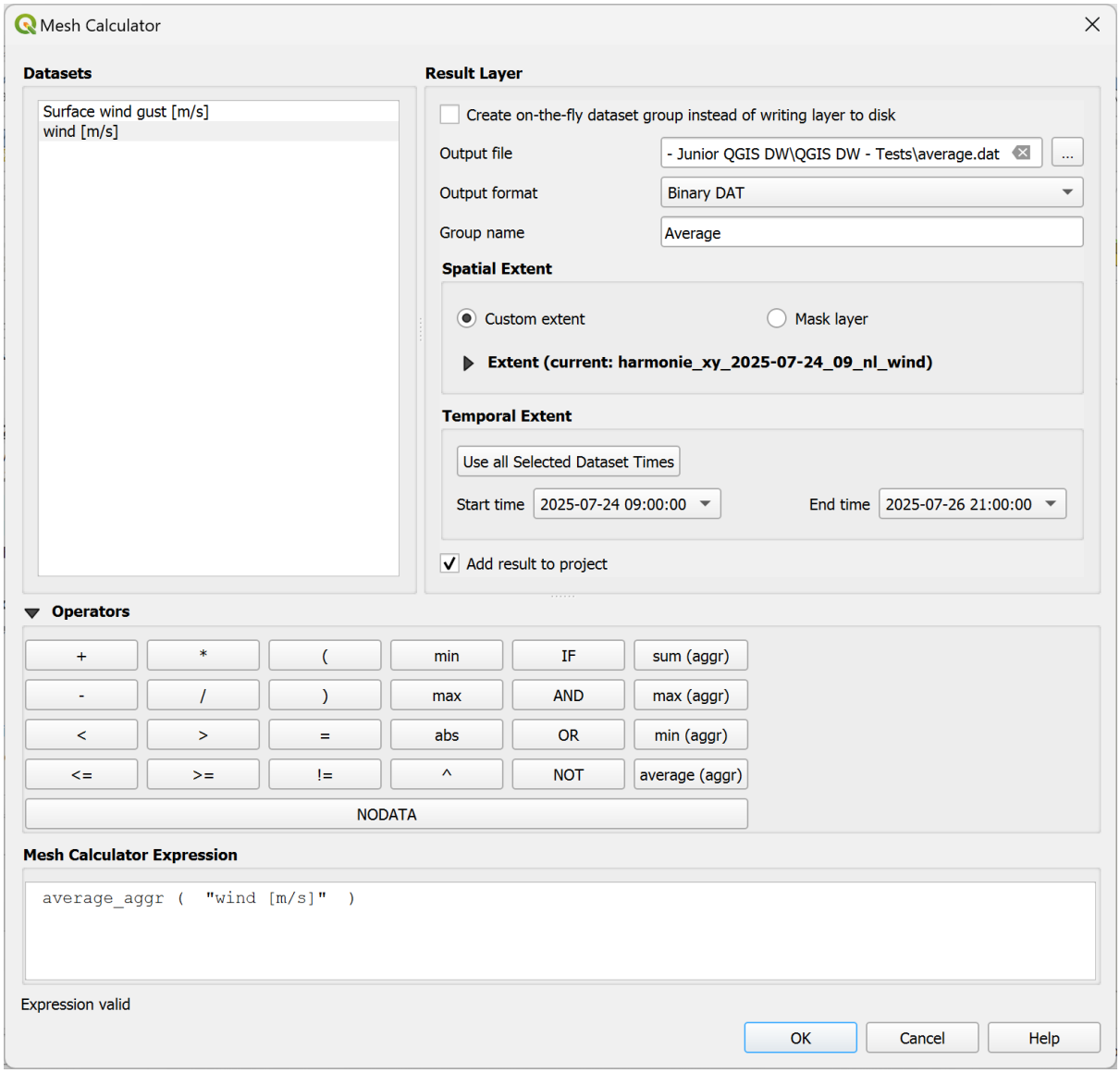


Abb. 14.19: Mesh Calculator

The *Datasets* list contains all dataset groups in the active mesh layer. To use a dataset group in an expression, double click its name in the list and it will be added to the *Mesh calculator expression* field. You can then use the operators to construct calculation expressions, or you can just type them into the box.

The *Result Layer* helps you configure properties of the output layer:

- *Create on-the-fly dataset group instead of writing layer to disk*:
  - If unchecked, the output is stored on disk as a new plain file. An *Output File* path and an *Output Format* are required.
  - If checked, a new dataset group will be added to the mesh layer. Values of the dataset group are not stored in memory but each dataset is calculated when needed with the formula entered in the mesh calculator. That virtual dataset group is saved with the project, and if needed, it can be removed or made persistent in file from the layer *Source* properties tab.

  In either case, you should provide a *Group Name* for the output dataset group.
- The *Spatial extent* to consider for calculation can be:
  - defined as a *Custom extent* using the *extent selector* widget.

  - defined by a polygon layer (*Mask layer*) of the project: the polygon features geometry are used to clip the mesh layer datasets

- The *Temporal extent* to take into account for datasets can be set with the *Start time* and *End time* options, selected from the existing dataset groups timesteps. They can also be filled using the *Use all selected dataset times* button to take the whole range.

- 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.

The *Operators* section contains all available operators. To add an operator to the mesh calculator expression box, click the appropriate button. Mathematical calculations (`+`, `-`, `*`, ...) and statistical functions (`min`, `max`, `sum (aggr)`, `average (aggr)`, ...) are available. Conditional expressions (`=`, `!=`, `<`, `>=`, `IF`, `AND`, `NOT`, ...) return either 0 for false and 1 for true, and therefore can be used with other operators and functions. The `NODATA` value can also be used in the expressions.

The *Mesh Calculator Expression* widget shows and lets you edit the expression to execute.

KAPITEL 15

Arbeiten mit Vektorkacheln (Vector Tiles)

15.1 Was sind Vektorkacheln?

Bei Vektorkacheln handelt es sich um Pakete mit geographischen Daten, die in vordefinierte ungefähr quadratische „Kacheln“ gepackt wurden, um sie über das Internet zu übertragen. Sie kombinieren vorgerenderte Raster- und Vektorkartenkacheln. Der Vektorkachelserver schickt anstelle eines vorgerenderten Kartenbilds vektorbasierte Kartendaten, die auf die Grenzen der Kacheln zugeschnitten wurden. Die zugeschnittenen Kacheln stellen die Maßstabsebenen des Vektorkacheldienstes dar und werden über einen Pyramidenansatz abgeleitet. Mit dieser Datenstruktur wird die Menge der zu übertragenden Daten gegenüber einem ungekachelten Vektordatensatz verringert. Es müssen nämlich nur die Daten des aktuellen Kartenausschnitts im aktuellen Maßstab übermittelt werden. Gegenüber einem gekachelten Rasterdatensatz wird die Datenübertragungsrate auch erheblich reduziert, weil Vektordaten typischerweise deutlich kleiner als eine Bilddatei sind. Vektorkacheln haben keine Darstellungsinformationen, so dass QGIS einen Darstelllungsstil anwenden muss, um die Daten darzustellen.

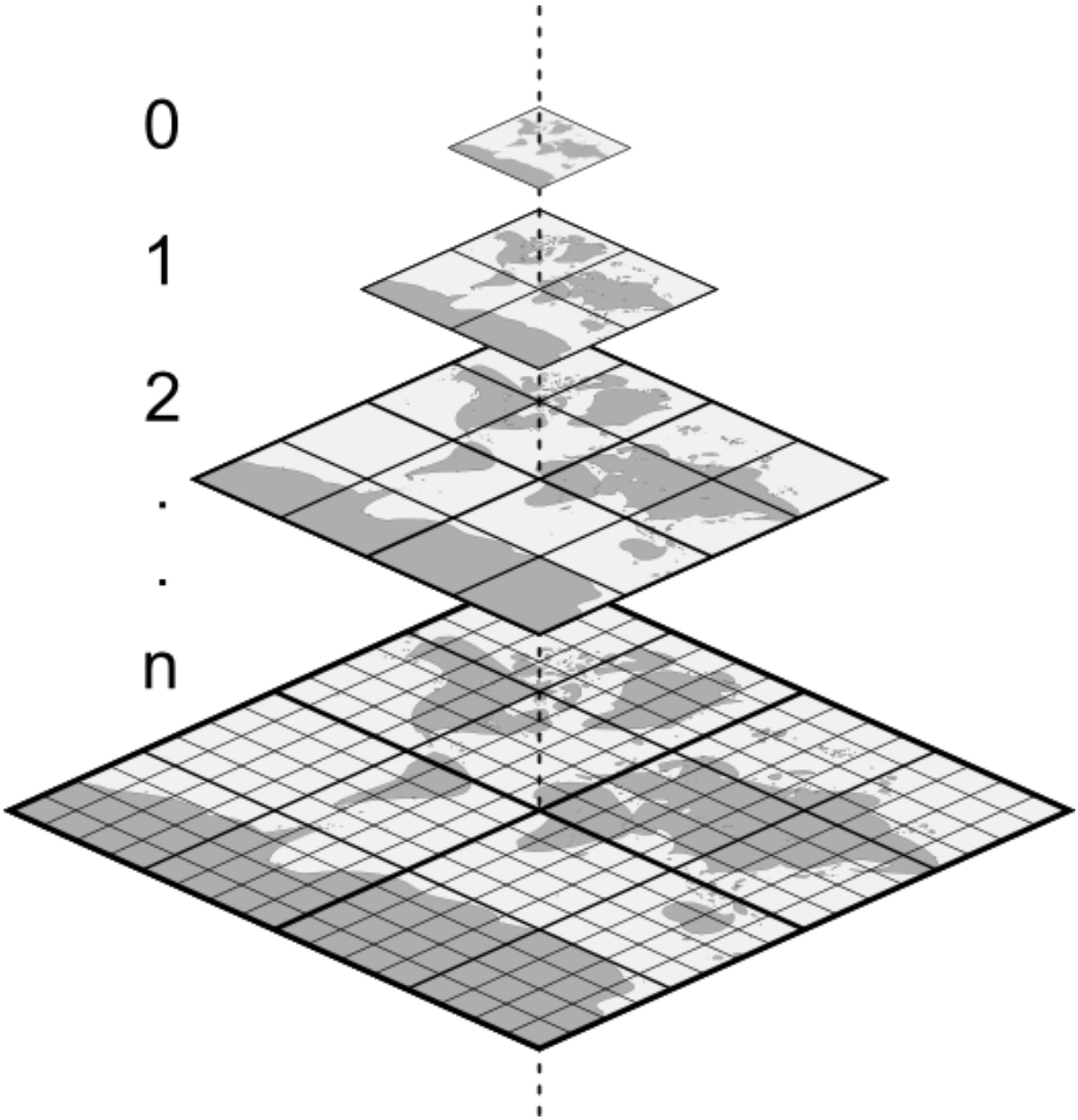


Abb. 15.1: Pyramidenstruktur eines Vektorkacheldatensatzes mit Maßstabsebenen

15.2 Unterstützte Formate

Vektorkacheln werden in folgenden Formaten unterstützt:

- entfernte Quellen (HTTP/S) - im XYZ-Muster - `type=xyz&url=http://example.com/{z}/{x}/{y}.pbf`
- lokale Dateien - im XYZ-Muster - z.B. `type=xyz&url=file:///path/to/tiles/{z}/{x}/{y}.pbf`
- eine lokale MBTile-Datenbank - z.B. `type=mbtiles&url=file:///path/to/file.mbtiles`

Sie laden einen Vektorkacheldatensatz in QGIS über den Tab *Vector Tile* im Dialog *Datenquellenverwaltung* . Lesen Sie *Verwendung von Vector Tiles-Diensten (Vektorkacheln)* wie Sie dabei im Detail vorgehen.

15.3 Eigenschaften von Vektorkacheldatensätzen

Der Dialog *Layereigenschaften* bietet für Vektorkacheln die folgenden Bereiche:

15.3.1 Information

Der Tab *Information* kann nicht editiert werden und ist ein guter Platz, um schnell zusammengefasste Informationen und Metadaten des aktuellen Layers nachzuschlagen. Hier erhalten Sie folgende Informationen:

- auf der Grundlage des Providers des Layers: Name, URI, Typ und Pfad der Datenquelle, Anzahl der Maßstabsebenen
- Benutzerdefinierte Eigenschaften, die verwendet werden, um zusätzliche Informationen über den Layer im aktiven Projekt zu speichern. Weitere Eigenschaften können mit PyQGIS erstellt und verwaltet werden, insbesondere mit der Methode setCustomProperty().
- the Coordinate Reference System: name, units, method, accuracy, reference (i.e., whether it is static or dynamic)
- von dem was bei den *Metadaten* ausgefüllt wurde: Zugang, Ausdehnung, Links, Kontakt, Protokoll…

15.3.2 Quelle

Auf dem Reiter *Quelle* werden grundlegende Informationen über die ausgewählte Vektorkachelquelle angezeigt, darunter:

- der *Layername*, der im *Layer*-Bedienfeld angezeigt wird;
- das *Koordinatenreferenzsystem*: Zeigt das *Koordinatenreferenzsystem (KBS)* des Layers an. Sie können das KBS des Layers ändern, indem Sie ein kürzlich verwendetes KBS aus der Dropdown-Liste auswählen oder auf die Schaltfläche KBS wählen klicken (siehe *Auswahl des Koordinatenbezugssystems*). Verwenden Sie dieses Verfahren nur, wenn das KBS falsch oder nicht angegeben ist.
- 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*

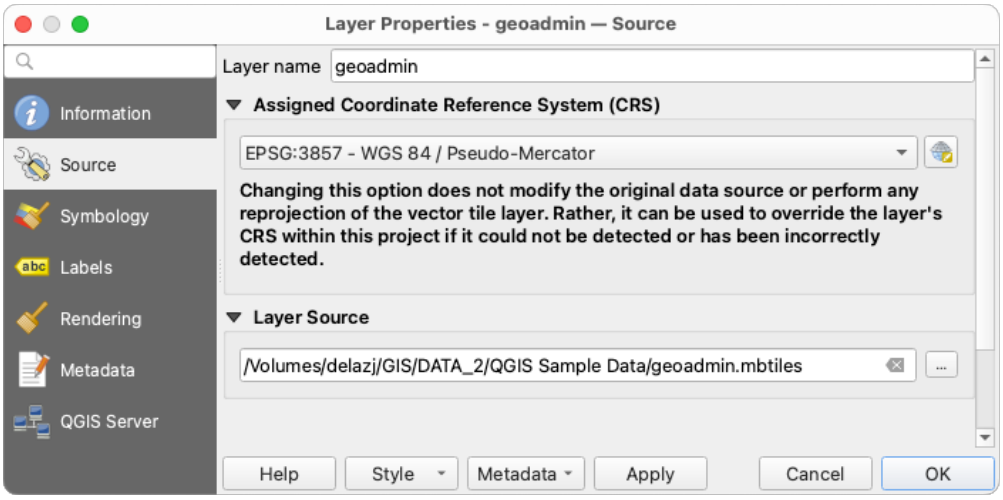


Abb. 15.2: Vektorkachellayereigenschaften - Quelle-Reiter

15.3.3 Eigenschaften der Reiter Symbolisierung und Beschriftung

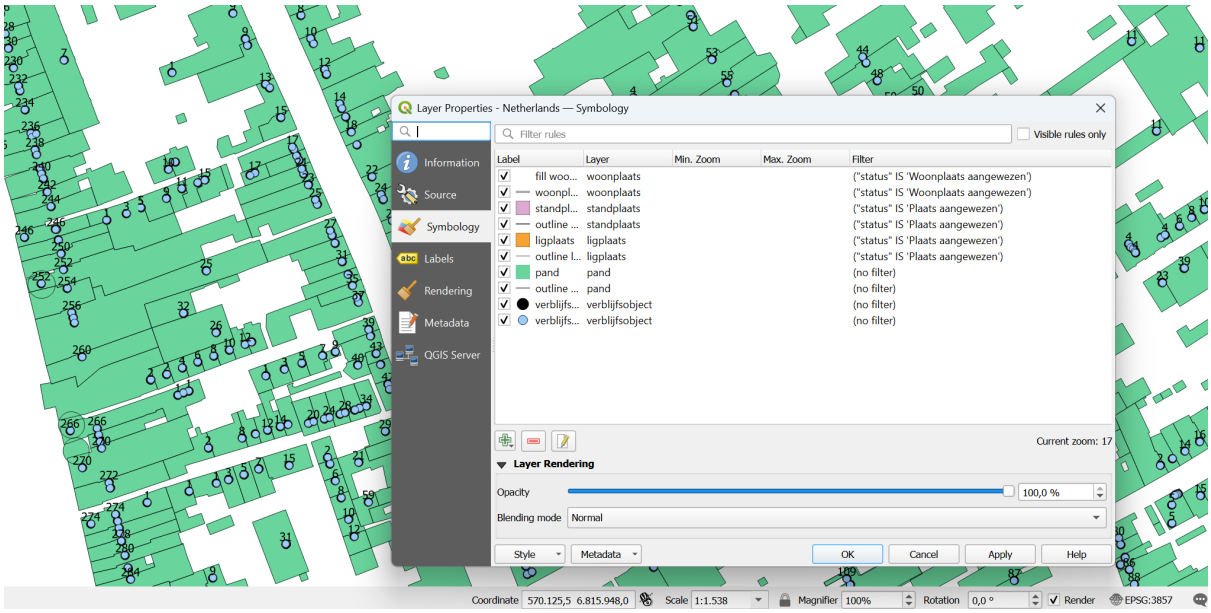


Abb. 15.3: Symbolisierung von Vektorkachellayern

Regeln definieren

Da Vektorkacheln aus Punkt-, Linien- und Polygongeometrien bestehen sind die jeweiligen Symbolisierungen verfügbar. Um eine kartografische Darstellung anzuwenden (mit Symbolisierung und/oder Beschriftung) können Sie entweder:

- Eine *Stil URL* benutzen, während Sie eine *Vektorkachelverbindung* herstellen. Die Symbolisierung wird sofort im Reiter *Symbolisierung* angezeigt, nachdem der Layer in QGIS geladen wurde.
- Oder Sie machen Ihre eigene Symbolisierung und Beschriftung in den entsprechenden Reitern der Layereigenschaften. Standardmäßig weist QGIS den Objekten auf Basis ihres Geometrietyps das selbe Symbol zu.

In beiden Fällen muss eine Darstellung für Vektorkacheln einen *regelbasierten Renderer* auf die Objekte anwenden, der folgendes enthält:

- ein *Symbol* oder eine *Beschriftung* für Vektoren
- eine *Beschriftung*, also einen Titel, um die Regel verständlich zu identifizieren
- den Namen eines bestimmten *Layers*, auf den die Regel angewendet werden soll, falls nicht `(alle Layer)` benutzt werden soll
- *Min. Zoom* und *Max. Zoom*, für den Darstellungsbereich. Symbolisierung und Beschriftung können vom Zoom abhängen.
- ein *Filter*, also ein QGIS-Ausdruck, um die Objekte festzulegen, auf die der Stil angewendet wird

Sie fügen eine Regel hinzu, indem Sie die Schaltfläche Regel hinzufügen drücken und den Symboltyp (*Markierung*, *Linie*, *Füllung*) entsprechend des Geoemtrietyps auswählen. Sie können auch die Gewählte Regel löschen oder die Aktuelle Regel bearbeiten.

Unten im Fenster sehen Sie den *Aktuellen Zoom*. Haken Sie die Option *Nur sichtbare Regeln* oben im Dialog an, um die Regelliste auf diejenigen einzuschränken, die im aktuellen Zoom sichtbar sind. Das erleichtert es Ihnen, mit komplexen Vektorstilen zu arbeiten und hilft Ihnen, problematische Regeln zu identifizieren. Die Eingabezeile *Filterregeln* hilft Ihnen dabei, schnell eine bestimmte Regel zu finden, indem sie die Felder *Beschriftung*, *Layer* und *Filter* durchsucht.

In der Abbildung Abb. 15.3 machen wir einen Stil für den Layer `landuse` von OpenStreetMap. Für das bessere Verständnis sind die meisten Regeln ausgeschaltet.

Layerdarstellung

Auf der Registerkarte *Symbolisierung* können Sie auch einige Optionen einstellen, die sich immer auf alle Objekte des Layers auswirken:

- *Deckkraft*: Mit diesem Werkzeug können Sie den darunter liegenden Layer in der Kartenansicht sichtbar machen. Verwenden Sie den Schieberegler, um die Sichtbarkeit Ihres Beschriftungslayers an Ihre Bedürfnisse anzupassen. Sie können den Prozentsatz der Sichtbarkeit auch im Menü neben dem Schieberegler genau festlegen.
- *Mischmodus*: Mit diesen Werkzeugen können Sie spezielle Rendering-Effekte erzielen, die Sie bisher nur von Graphikprogrammen her kennen. Die Pixel von sich überlagernden Layern werden durch die verwendete Einstellungen gemischt, wie in *Mischmodi* beschrieben.

Stile

Auf den meisten Reitern finden Sie unten die Menüauswahl *Stil* ►, die einen schnellen Weg bietet, Stile zu speichern, zu laden, zu erzeugen sowie zu wechseln um sie auf die Vektorkacheln anzuwenden. Vektorkachelstile können von QGIS als *QML*-Dateien gespeichert werden und sie können aus folgenden Formaten geladen werden:

- *QML*-Dateien (*QML - Das QGIS-Stil-Dateiformat*)
- *MapBox GL Json* Stildateien

Weitere Details unter *Layer-Eigenschaften speichern und teilen*.

15.3.4 Darstellung

Unter *Maßstabsabhängige Sichtbarkeit* können Sie die Maßstäbe *Maximum (inklusive)* und *Minimum (exklusive)* festlegen und damit einen Bereich von Maßstäben definieren, in dem die Objekte des Layers sichtbar sind. Außerhalb dieses Bereichs werden sie ausgeblendet. Die Schaltfläche Auf aktuellen Kartenmaßstab setzen hilft Ihnen, den aktuellen Maßstab der Kartenansicht als Grenze für den Sichtbarkeitsbereich zu verwenden. Siehe *Einstellen des Sichtbarkeitsmaßstabs* für weitere Informationen.

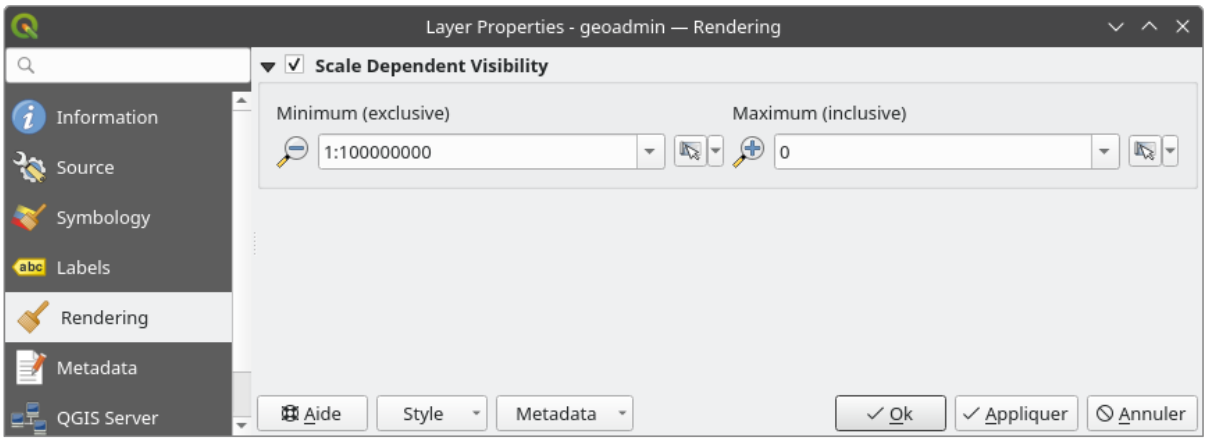


Abb. 15.4: Vektorkachellayereigenschaften - Darstellungs-Reiter

15.3.5 Metadaten

Der Reiter *Metadata* bietet Ihnen Optionen zum Erstellen und Bearbeiten eines Metadatenberichts zu Ihrem Layer. Siehe *Metadaten* für weitere Informationen.

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*

KAPITEL 16

Arbeiten mit Punktwolken

16.1 Einführung in Punktwolken

Was ist eine Punktwolke?

Eine Punktwolke ist ein dreidimensionales Bild eines Raums, das aus vielen einzelnen Datenpunkten (bis zu Milliarden oder sogar Billionen) besteht. Jeder dieser Punkte hat eine x-, y- und z-Koordinate. Je nach Erfassungsmethode verfügen Punktwolken in der Regel auch über zusätzliche Attribute, die aus der Erfassung stammen, wie beispielsweise Farbwerte oder Intensität. Diese Attribute können beispielsweise dazu verwendet werden, Punktwolken in verschiedenen Farben darzustellen. In QGIS kann eine Punktwolke verwendet werden, um ein dreidimensionales Bild der Landschaft (oder eines anderen Raums) zu erzeugen.

Unterstützte Formate

QGIS unterstützt die Datenformate Entwine Point Tile (EPT) und LAS/LAZ. Für die Arbeit mit Punktwolken speichert QGIS die Daten immer im EPT-Format. EPT ist ein Speicherformat, das aus mehreren Dateien besteht, die in einem gemeinsamen Ordner gespeichert sind. Um einen schnellen Zugriff auf die Daten zu ermöglichen, verwendet EPT eine Indizierung. Weitere Informationen zum EPT-Format finden Sie auf der Entwine-Homepage <https://entwine.io/en/latest/entwine-point-tile.html>_

Wenn die Daten im LAS- oder LAZ-Format vorliegen, konvertiert QGIS sie beim ersten Laden in das EPT-Format. Je nach Größe der Datei kann dies einige Zeit in Anspruch nehmen. Dabei wird in dem Ordner, in dem sich die LAS/LAZ-Datei befindet, ein Unterordner nach folgendem Schema erstellt: `ept_ + name_LAS/LAZ_file`. Wenn ein solcher Unterordner bereits vorhanden ist, lädt QGIS die EPT-Datei sofort (was zu einer verkürzten Ladezeit führt).

Gut zu wissen

In QGIS ist es (noch) nicht möglich, Punktwolken zu bearbeiten. Wenn Sie Ihre Punktwolke bearbeiten möchten, können Sie CloudCompare verwenden, ein Open-Source-Tool zur Bearbeitung von Punktwolken. Auch die Point Data Abstraction Library (PDAL – ähnlich wie GDAL) bietet Ihnen Optionen zur Bearbeitung von Punktwolken (PDAL ist nur über die Befehlszeile verfügbar).

Aufgrund der großen Anzahl von Datenpunkten ist es nicht möglich, eine Attributtabelle von Punktwolken in QGIS anzuzeigen. Das Werkzeug Objekte abfragen unterstützt jedoch Punktwolken, sodass Sie alle Attribute anzeigen können, auch die eines einzelnen Datenpunkts.

Wenn Sie aus einem vorhandenen Punktwolken-Layer einen neuen Layer im gleichen oder einem anderen unterstützten Format erstellen möchten, lesen Sie *Erstellen neuer Layer aus einem vorhandenen Layer*.

16.2 Punktwolkenlayer-Eigenschaften

Der *Layereigenschaften* Dialog für eine Punktwolkenebene enthält allgemeine Einstellungen für die Ebene und ihre Darstellung. Außerdem werden Informationen zur Ebene angezeigt.

Sie können den *Layereigenschaften* Dialog öffnen, indem Sie:

- Doppelklicken Sie im Bedienfeld *Layer* auf den Layer oder klicken Sie mit der rechten Maustaste darauf und wählen Sie im Kontextmenü *Eigenschaften…* aus.
- in der Menüleiste *Layer* ► *Layereigenschaften …* auswählen wenn der Layer aktiv ist

die Punktwolken-Layer *Layereigenschaften* enthalten die folgenden Abschnitte:

Bemerkung

Die meisten Eigenschaften eines Punktwolken-Layers können über das Menü *Stil* am unteren Rand des Eigenschaftendialogs in einer `.qml`-Datei gespeichert oder aus dieser geladen werden. Weitere Details finden Sie unter *Layer-Eigenschaften speichern und teilen*.

16.2.1 Information

Die Registerkarte |Metadaten| *Information* ist nur zum Lesen vorgesehen und stellt einen interessanten Ort dar, um schnell zusammengefasste Informationen und Metadaten über den aktuellen Layer zu erhalten. Die bereitgestellten Informationen sind:

- Allgemeine Angaben wie Name im Projekt, Quellpfad, Zeitpunkt der letzten Speicherung und Größe, verwendeter Anbieter
- Benutzerdefinierte Eigenschaften, die verwendet werden, um zusätzliche Informationen über den Layer im aktiven Projekt zu speichern. Zu den standardmäßigen benutzerdefinierten Eigenschaften können *Layer-Notizen* gehören. Weitere Eigenschaften können mit PyQGIS erstellt und verwaltet werden, insbesondere mit der Methode setCustomProperty().
- Basierend auf dem Anbieter des Layers: Ausdehnung und Anzahl der Punkte
- Das Koordinatenreferenzsystem: Name, Einheiten, Methode, Genauigkeit, Referenz (d. h. ob es statisch oder dynamisch ist)
- Vom Anbieter bereitgestellte Metadaten: Erstellungsdatum, Version, Datenformat, Maßstab X/Y/Z, …
- Entnommen aus der Registerkarte *Metadaten* (wo sie bearbeitet werden können): Zugriff, Umfang, Links, Kontakte, Verlauf…

[1] Also available in the *Layer styling panel*

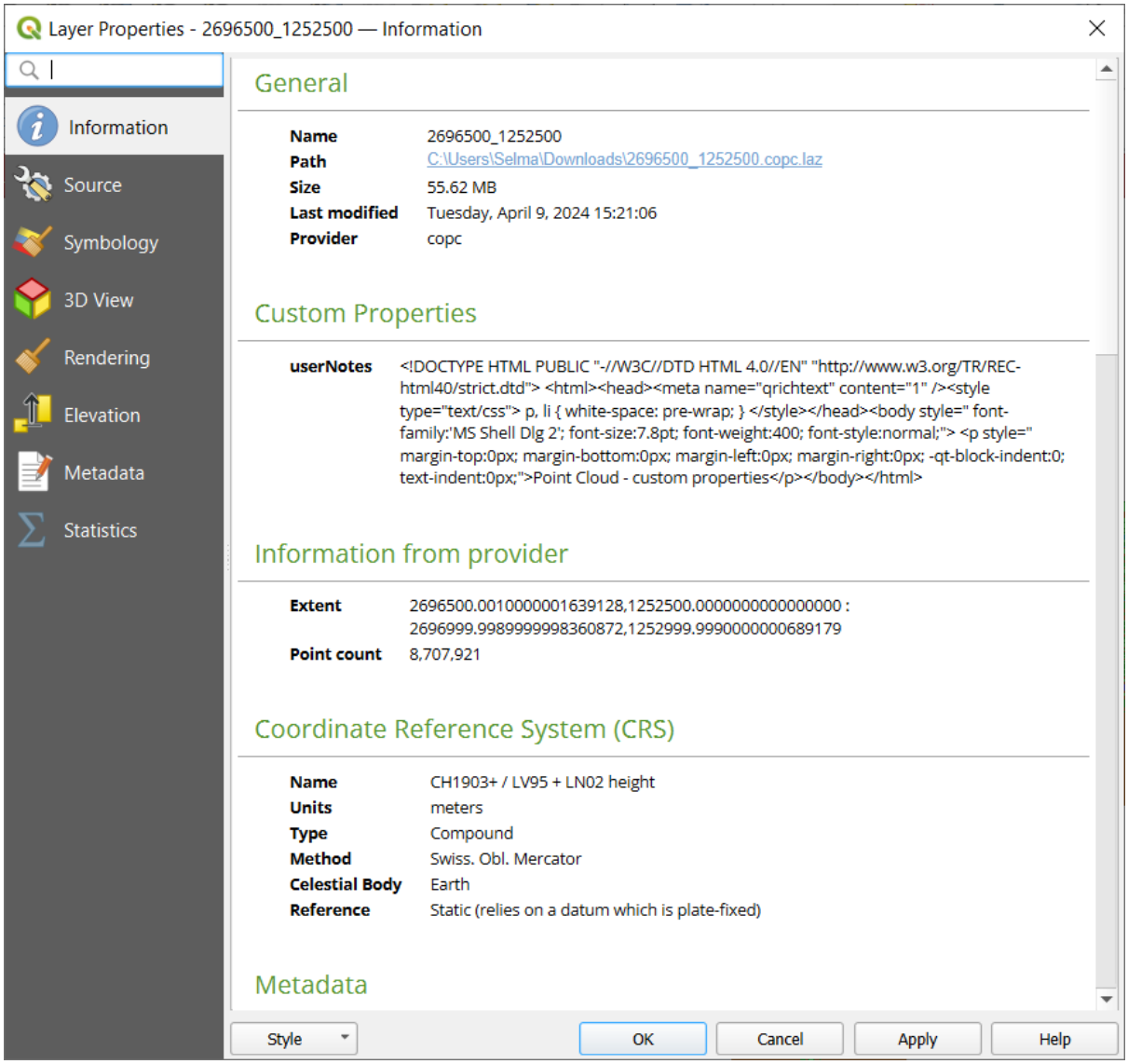


Abb. 16.1: Der Informationen-Reiter von Punktwolken

16.2.2 Quelle

Auf der Registerkarte :guilabel:‚Quelle‘ können Sie grundlegende Informationen zum Punktwolken-Layer anzeigen und bearbeiten:

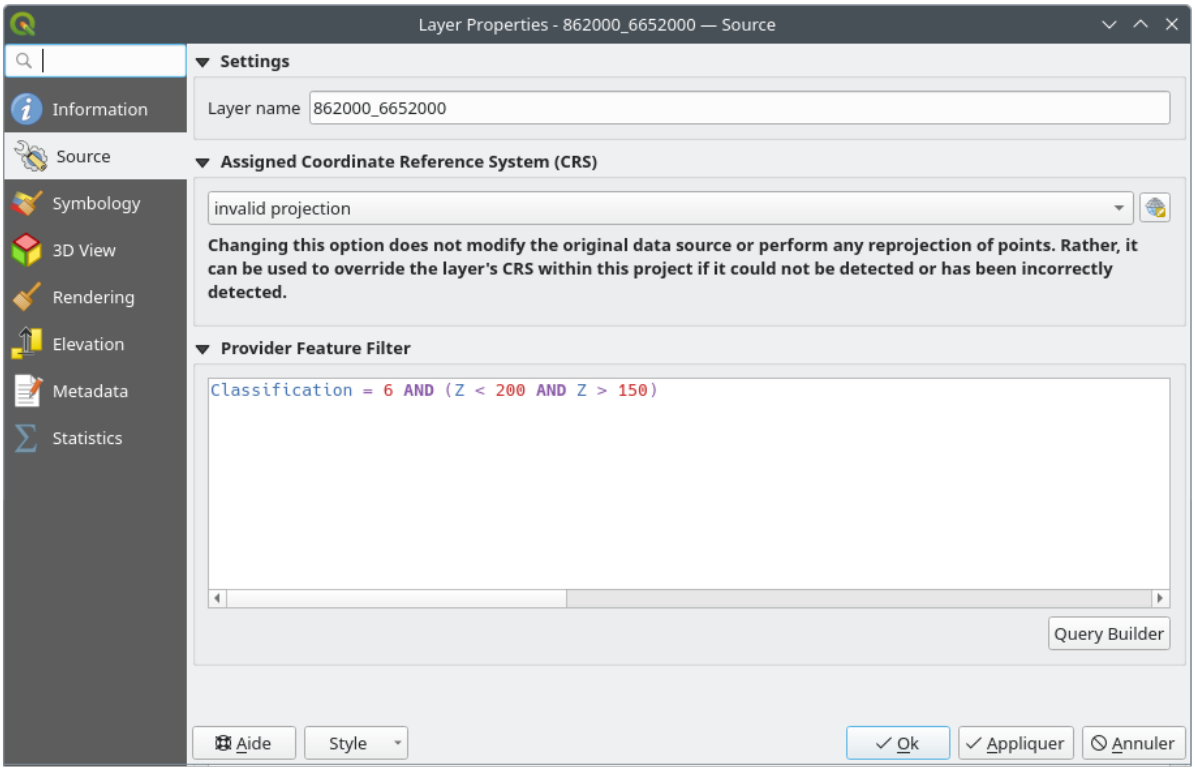


Abb. 16.2: Der Quelle-Reiter von Punktwolken

- *Einstellungen*: Legen Sie einen Layer-Name fest, der sich vom Dateinamen des Layers unterscheidet und zur Identifizierung des Layers im Projekt verwendet wird (im Layer-Bedienfeld, mit Ausdrücken, in der Drucklayout-Legende usw.).
- *Zugewiesenes Koordinatenbezugssystem (KBS)*: Hier können Sie das dem Layer zugewiesene *Koordinatenbezugssystem* ändern, indem Sie ein kürzlich verwendetes System aus der Dropdown-Liste auswählen oder mit der Schaltfläche Projektion festlegen, das KBS auswählen (siehe *Auswahl des Koordinatenbezugssystems*). Wenden Sie dieses Verfahren nur an, wenn das auf den Layer angewendete KBS falsch ist oder wenn kein KBS zugewiesen wurde.
- *Provider Feature Filter*: Ermöglicht es, die im aktuellen Projekt zu verwendenden Daten mithilfe von Funktionen und Ausdrücken auf PDAL-Datenanbieterebene auf eine Teilmenge zu beschränken. Klicken Sie unten auf die Schaltfläche *Ausdruckseditor*, um mit der Einstellung des Filters zu beginnen.

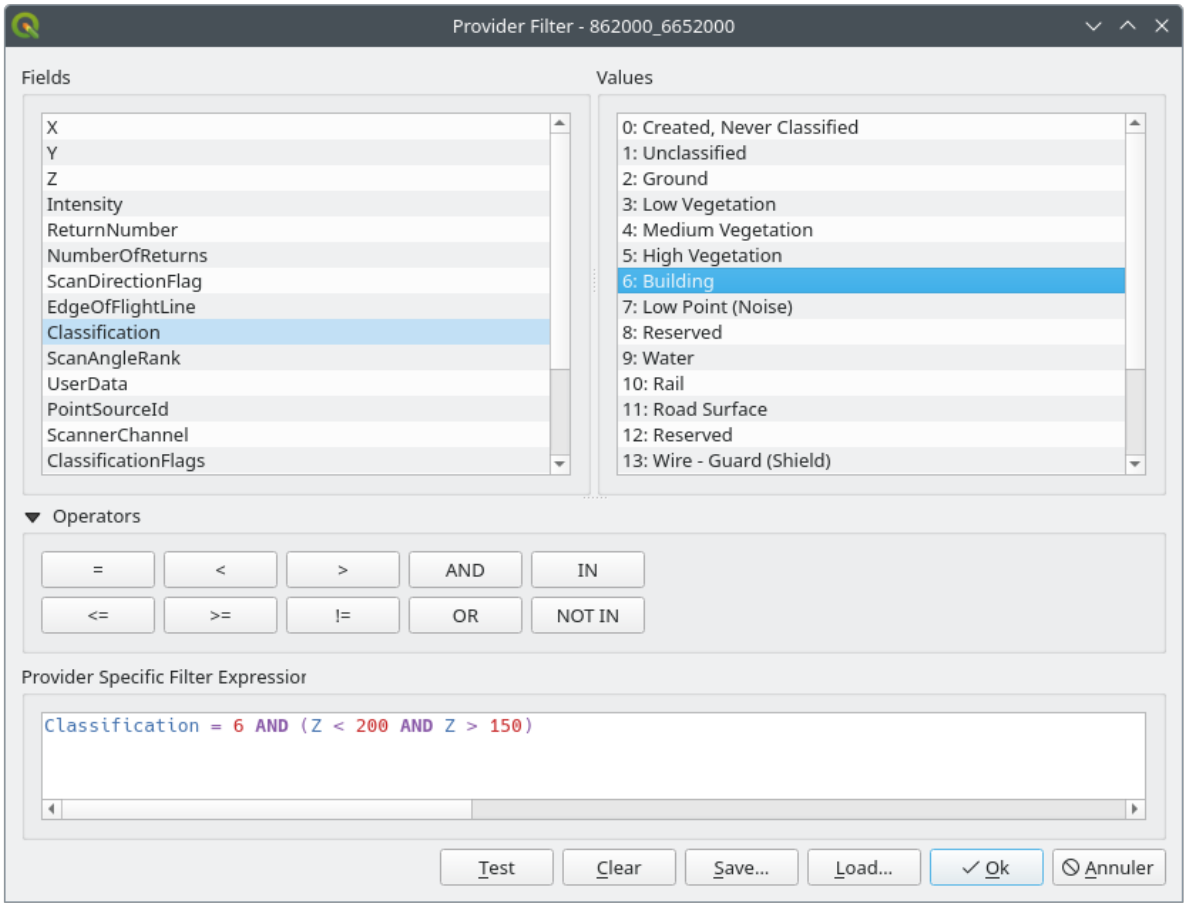


Abb. 16.3: Filterung des Punktwolken-Layers auf Gebäude in einer bestimmten Höhe

Im unteren Teil des Dialogfelds können Sie einen *Anbieterspezifischen Filterausdruck* eingeben. Ein solcher Ausdruck kann dank folgender Funktionen erstellt werden:

- *Felder*: Die Liste enthält alle Attribute des Punktwolken-Layers. Um ein Attribut zum Ausdruck hinzuzufügen, doppelklicken Sie auf dessen Namen oder geben Sie ihn einfach in das Textfeld ein.
- *Werte*: Der Bereich listet Werte oder Statistiken des aktuell ausgewählten Attributs als `Schlüssel:Wert`-Paare auf. Um einen Wert zum Ausdrucksfeld hinzuzufügen, doppelklicken Sie auf dessen Namen in der Liste: Je nach Art des Attributs wird entweder der Schlüssel oder der Wert zum Ausdruck hinzugefügt. Oder geben Sie den Wert in das Ausdrucksfeld ein.
- *Operatoren*: Diese Symbolleiste enthält alle verwendbaren Operatoren. Um einen Operator zum Ausdrucksfeld hinzuzufügen, klicken Sie auf die entsprechende Schaltfläche. Es stehen relationale Operatoren (= , > , …) und logische Operatoren (`AND`, `OR`, …) zur Verfügung.

Mit der Schaltfläche *Test* können Sie die Syntax Ihrer Abfrage überprüfen. Verwenden Sie die Schaltfläche *Löschen*, um die Abfrage zu löschen und die Layer in ihren ursprünglichen Zustand zurückzusetzen (d. h. alle Punkte in der Layer vollständig zu laden). Es ist möglich, die Abfrage als `.QQF`-Datei zu speichern (*Speichern…*) oder die Abfrage aus einer Datei in das Dialogfeld zu laden (*Laden…*).

Wenn ein Filter angewendet wird, behandelt QGIS die resultierende Teilmenge so, als wäre sie der gesamte Layer. Wenn Sie beispielsweise den oben gezeigten Filter zum Filtern von Gebäuden angewendet haben, können Sie keine Punkte anzeigen, abfragen, speichern oder bearbeiten, die zur Vegetationsklassifizierung gehören, da diese nicht Teil der Teilmenge sind.

Tipp

Gefilterte Layer werden im „ Layer“ Bedienfeld gekennzeichnet

Im *Layer*-Bedienfeld wird ein gefilterter Layer mit einem Filter-Symbol daneben aufgelistet, das die verwendete Abfrage anzeigt, wenn Sie mit der Maus über das Symbol fahren. Doppelklicken Sie auf das Symbol, um das Dialogfeld *Ausdruckseditor* zum Bearbeiten zu öffnen. Dies kann auch über das Menü *Layer* ► *Filter…* erfolgen.

16.2.3 Symbolisierung

Auf der Registerkarte *Symbologie* werden die Einstellungen für die Darstellung der Punktwolke vorgenommen. Im oberen Bereich befinden sich die Einstellungen der verschiedenen Objekt-Renderer. Im unteren Bereich gibt es Abschnitte, in denen allgemeine Einstellungen für die gesamte Ebene vorgenommen werden können, die über die Objekt-Renderer hinweg gelten.

Objekt-Rendering-Typen

Es gibt verschiedene Optionen für die Darstellung von Punktwolken, die über das Dropdown-Menü oben auf der Registerkarte *Symbologie* ausgewählt werden können (siehe Abb. 16.4):

- *Nur Ausdehnung*: Es wird nur ein Begrenzungsrahmen für die Ausdehnung der Daten angezeigt; praktisch, um sich einen Überblick über die Datenausdehnung zu verschaffen. Wie an anderer Stelle hilft Ihnen das *Symbol Widget* dabei, alle gewünschten Eigenschaften (Farbe, Strichstärke, Deckkraft, Unterebenen usw.) für den Rahmen zu konfigurieren.
- *Attribute nach Farbverlauf*: Die Daten werden über einen Farbverlauf dargestellt. Siehe *Attribute nach Farbverlauf*
- *RGB*: Zeichnen Sie die Daten unter Verwendung von Rot-, Grün- und Blau-Farbwerten. Siehe *RGB Renderer*
- *Klassifizierung*: Die Daten werden für verschiedene Klassen in unterschiedlichen Farben dargestellt. Siehe *Klassifizierungs-Renderer*

Wenn eine Punktwolke geladen wird, wählt QGIS anhand einer bestimmten Logik den besten Renderer aus:

- Wenn der Datensatz Farbinformationen enthält (Rot-, Grün-, Blau-Attribute), wird der RGB-Renderer verwendet.
- Wenn nicht, aber der Datensatz ein Attribut `Classification` enthält, wird der klassifizierte Renderer verwendet.
- Andernfalls wird auf die Darstellung basierend auf dem Z-Attribut zurückgegriffen.

Wenn Sie die Attribute der Punktwolke nicht kennen, bietet der :guilabel:Statistik *Reiter* einen guten Überblick darüber, welche Attribute in der Punktwolke enthalten sind und in welchen Bereichen sich die Werte befinden.

Für jeden Renderer können Sie die Datenanzeige optimieren, indem Sie die *Punktsymbolgröße* anpassen oder die *Oberflächentriangulation* aktivieren.

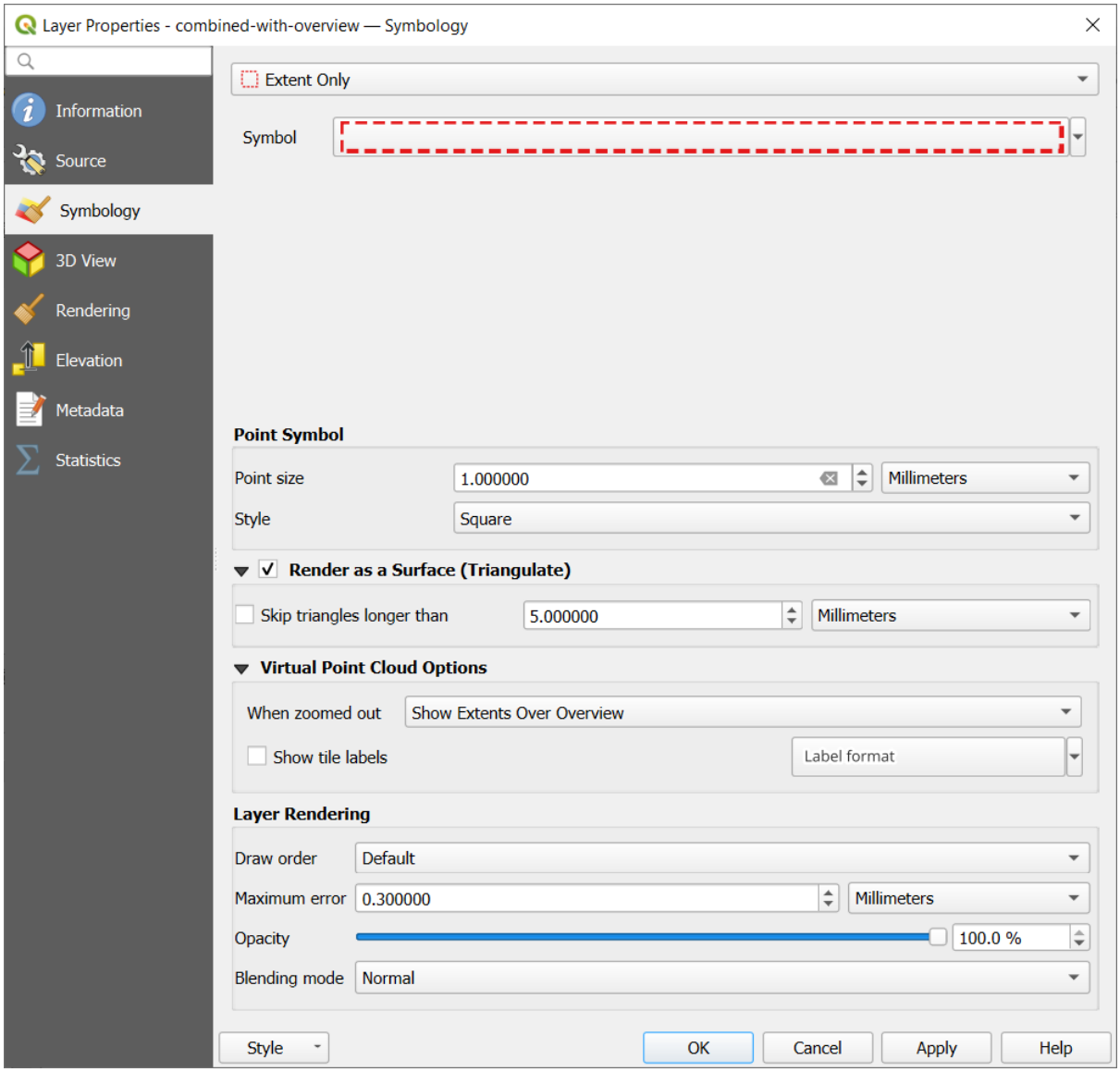


Abb. 16.4: Der Symbologie-Reiter von Punktwolken

Attribute nach Farbverlauf

Mit *Attribute nach Farbverlauf* können die Daten anhand von numerischen Werten über einen Farbverlauf dargestellt werden. Solche numerischen Werte können beispielsweise ein vorhandenes Intensitätsattribut oder der Z-Wert sein. Abhängig von einem Minimal- und einem Maximalwert werden die anderen Werte durch Interpolation auf den Farbverlauf verteilt. Die einzelnen Werte und ihre Zuordnung zu einer bestimmten Farbe werden als „Farbkarte" bezeichnet und sind in der Tabelle aufgeführt. Es gibt verschiedene Einstellungsoptionen, die unterhalb der Abbildung beschrieben werden.

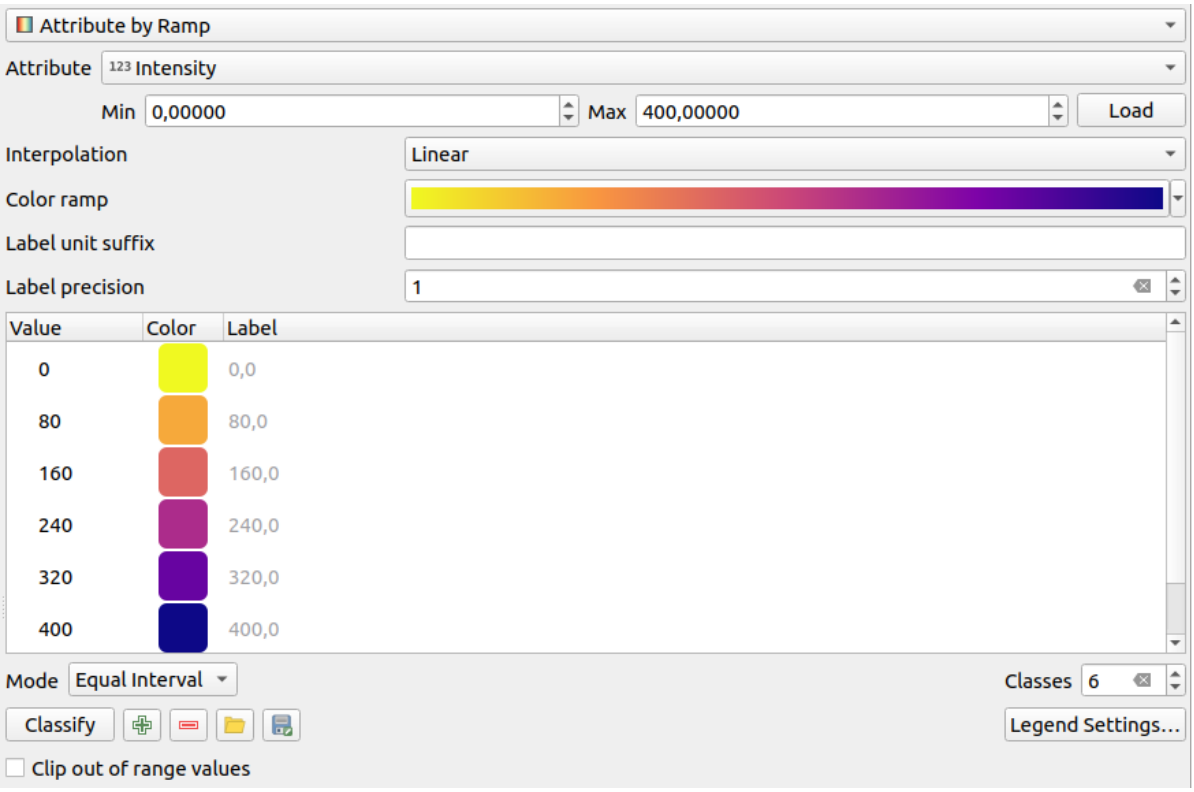


Abb. 16.5: Der Symbologie-Reiter von Punktwolken: Attribute nach Farbverlauf

- *Min* und *Max* definieren den Bereich, der auf die Farbskala angewendet wird: Der Wert *Min* steht für das linke Ende der Farbskala, der Wert *Max* für das rechte Ende, die Werte dazwischen werden interpoliert. Standardmäßig erkennt QGIS das Minimum und Maximum aus dem ausgewählten Attribut, diese Werte können aber geändert werden. Nachdem Sie die Werte geändert haben, können Sie die Standardeinstellungen wiederherstellen, indem Sie auf die Schaltfläche *Laden* klicken.
- Der Eintrag *Interpolation* definiert, wie den Werten ihre Farbe zugewiesen wird:
  - *Diskret* (ein <= Symbol erscheint im Kopf der *Wert* Spalte): Die Farbe wird vom nächstgelegenen Farbkarteneintrag mit gleichem oder höherem Wert übernommen
  - *Linear* Die Farbe wird linear aus den Einträgen der Farbtabelle oberhalb und unterhalb des Pixelwerts interpoliert, was bedeutet, dass jedem Datensatzwert eine eindeutige Farbe zugeordnet ist.
  - *Genau* (ein = Sybol erscheint im Kopf über der *Wert* Spalte): Nur Pixel mit einem Wert, der einem Eintrag in der Farbkarte entspricht, werden mit einer Farbe versehen; andere werden nicht dargestellt.
- Das Widget *Farbverlauf* hilft Ihnen bei der Auswahl des Farbverlaufs, der dem Datensatz zugewiesen werden soll. Wie bei *diesem Widget* üblich, können Sie einen neuen Farbverlauf erstellen und den aktuell ausgewählten bearbeiten oder speichern.
- Mit *Beschriftungseinheitssuffix* wird eine Beschriftung hinter dem Wert in der Legende hinzugefügt, und mit *Beschriftungsgenauigkeit* wird die Anzahl der anzuzeigenden Dezimalstellen festgelegt.

Mit dem Klassifizierungs *Modus* können Sie festlegen, wie die Werte auf die Klassen verteilt werden:

- *Kontinuierlich*: Die Anzahl und Farbe der Klassen werden aus den Farbverlaufsstopps übernommen; die Grenzwerte werden entsprechend der Verteilung der Stopps im Farbverlauf festgelegt (weitere Informationen zu Stopps finden Sie unter *Einstellen einer Farbrampe*).
- *Gleiches Intervall*: Die Anzahl der Klassen wird durch das Feld *Klassen* am Ende der Zeile festgelegt; die Grenzwerte werden so definiert, dass alle Klassen die gleiche Größe haben.

Die Klassen werden automatisch ermittelt und in der Farbtabelle angezeigt. Sie können diese Klassen jedoch auch manuell bearbeiten:

- Durch Doppelklicken auf einen *Wert* in der Tabelle können Sie den Klassenwert ändern.
- Durch Doppelklicken in die Spalte *Farbe* wird das Widget :ref:„Farbauswahl“ geöffnet, in dem Sie eine Farbe auswählen können, die für diesen Wert angewendet werden soll.
- Doppelklicken Sie in der Spalte *Beschriftung*, um die Bezeichnung der Klasse zu ändern.
- Ein Rechtsklick auf ausgewählte Zeilen in der Farbtabelle zeigt ein Kontextmenü mit den Optionen *Farbe ändern…* und :guilabel:` Transparenz ändern…` für die Auswahl an.

Unterhalb der Tabelle befinden sich die Optionen zum Wiederherstellen der Standardklassen mit *Klassifizieren* oder zum manuellen Hinzufügen von Werten oder Entfernen ausgewählter Werte aus der Tabelle.

Da ein benutzerdefinierter Farbverlauf sehr komplex sein kann, besteht auch die Möglichkeit, ein vorhandenen Farbverlauf zu Laden oder zu Speichern, um ihn in anderen Layern zu verwenden (Format ist `txt`-Datei).

Wenn Sie für *Interpolation* die Option *Linear* ausgewählt haben, können Sie auch Folgendes konfigurieren:

- *Werte außerhalb des Bereichs ausschließen* Standardmäßig weist die lineare Methode den Werten im Datensatz, die unter dem festgelegten Wert *Min*, bzw. über dem festgelegten Wert *Max* liegen, die Farbe der ersten Klasse, bzw. der letzten Klasse zu. Aktivieren Sie diese Einstellung, wenn Sie diese Werte nicht rendern möchten.
- *Legenden-Einstellungen*, zur Anzeige im *Layer*-Bedienfeld und in der *Layout-Legende*. Die Anpassung funktioniert genauso wie bei einem Raster-Layer (weitere Details finden Sie unter *Anpassen der Rasterlegende*).

RGB Renderer

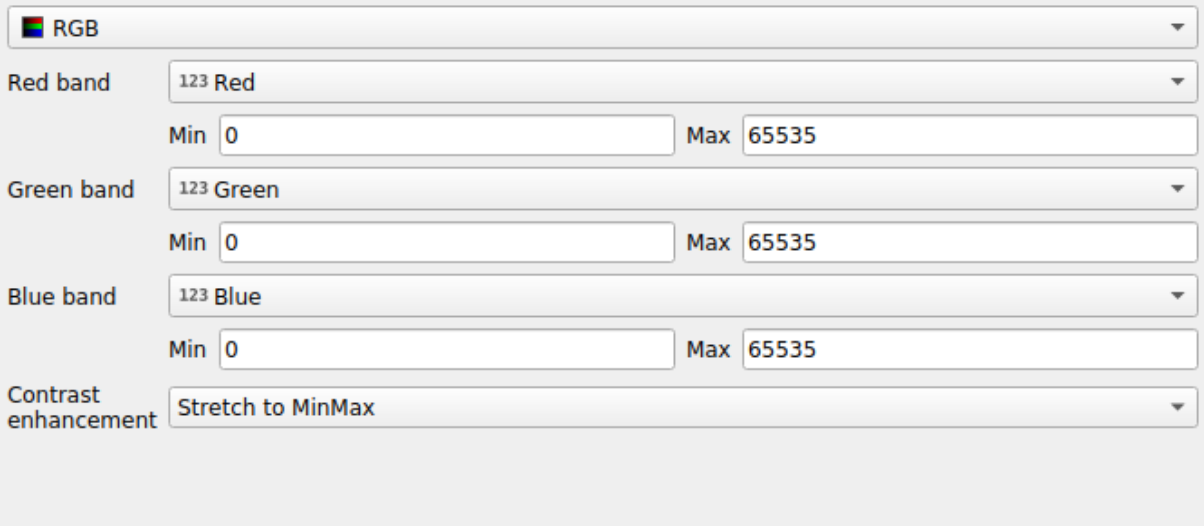


Abb. 16.6: Der Punktwolken-RGB-Renderer

Mit dem Renderer *RGB* werden drei ausgewählte Attribute aus der Punktwolke als Rot-, Grün- und Blaukomponente verwendet. Wenn die Attribute entsprechend benannt sind, wählt QGIS sie automatisch aus, ruft die Werte *Min* und *Max* für jedes Band ab und skaliert die Farbgebung entsprechend. Es ist jedoch auch möglich, die Werte manuell zu ändern.

Zur *Kontrastverbesserung*-Methode können die folgenden Methoden angewendet werden: *Keine Verbesserung*, *Auf MinMax strecken*, *Auf MinMax strecken und beschneiden* und *Auf MinMax beschneiden*

Bemerkung

Das Werkzeug *Kontrastverbesserung* befindet sich noch in der Entwicklung. Wenn Sie Probleme damit haben, sollten Sie die Standardeinstellung *Auf MinMax strecken* verwenden.

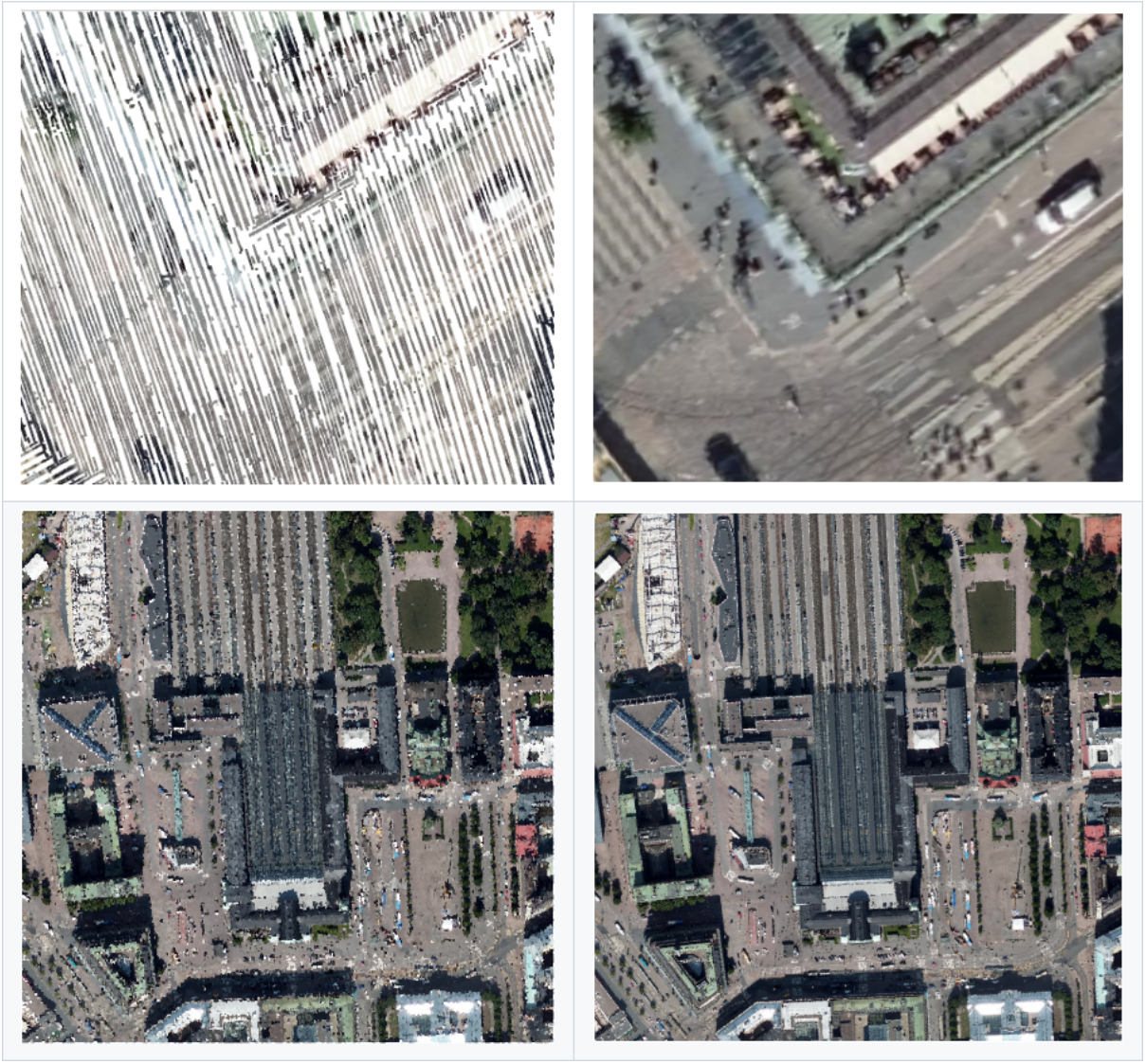


Abb. 16.7: Beispiel für einen RGB-Renderer (links) in Kombination mit der Option zur Oberflächentriangulation (rechts)

Klassifizierungs-Renderer

In der *Classification*-Darstellung wird die Punktwolke anhand eines Attributs farblich differenziert dargestellt. Es kann jede Art von Attribut verwendet werden (numerisch, Zeichenfolge, …). Punktwolkendaten enthalten oft ein Feld namens `Classification`. Dieses enthält in der Regel Daten, die automatisch durch Nachbearbeitung ermittelt wurden, z. B. über die Vegetation. Mit *Attribute* können Sie das Feld aus der Attributtabelle auswählen, das für die Klassifizierung verwendet werden soll. Standardmäßig verwendet QGIS die Definitionen der LAS-Spezifikation (siehe Tabelle „ASPRS Standard Point Classes" im PDF auf der ASPRS-Homepage). Die Daten können jedoch von diesem Schema abweichen. Im Zweifelsfall müssen Sie die Person oder Institution, von der Sie die Daten erhalten haben, nach den Definitionen fragen.

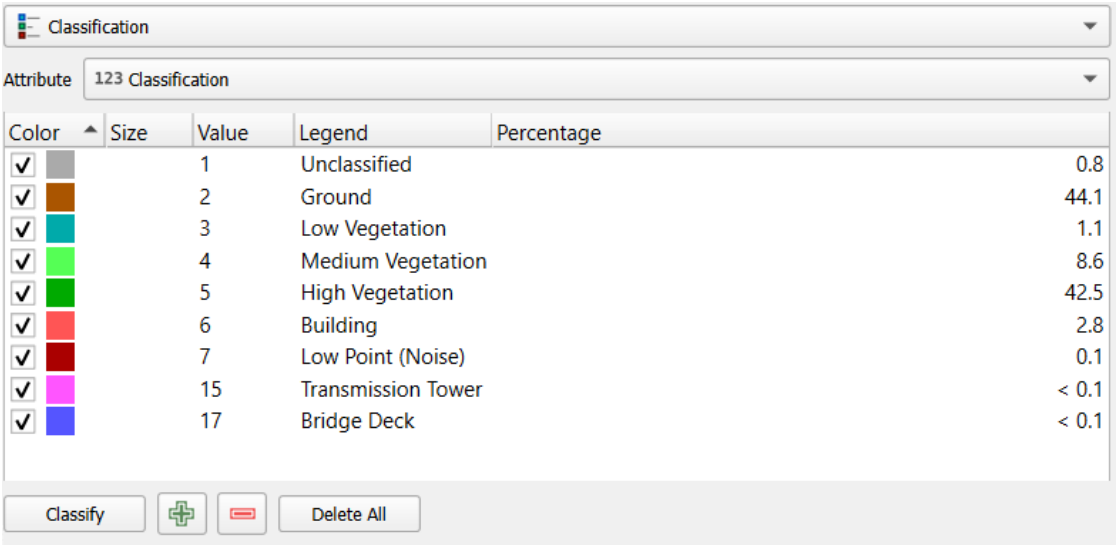


Abb. 16.8: Der Punktwolkenklassifizierungs-Renderer

In der Tabelle werden alle verwendeten Werte mit der entsprechenden Farbe und Legende angezeigt. Am Anfang jeder Zeile befindet sich ein Kontrollkästchen. Wenn dieses Kontrollkästchen deaktiviert ist, wird dieser Wert nicht mehr auf der Karte angezeigt. Mit einem Doppelklick in der Tabelle können Sie Folgendes ändern:

- die *Farbe*: das Farbauswahl-Widget wird geöffnet
- *Größe*: Wenn Sie einer Kategorie die Größe `0` zuweisen, wird die für den Layer festgelegte *Standardpunktgröße* wieder verwendet.
- the *Wert*
- the *Legende*
- Die Spalte *prozentualer Anteil* zeigt Ihnen den Anteil der Kategorie innerhalb des Layers an.

Sie können auch mit der rechten Maustaste auf eine oder mehrere Zeilen klicken, um ein Kontextmenü mit Optionen zum Ändern der Farbe, Transparenz oder Punktgröße zu öffnen.

Unterhalb der Tabelle befinden sich Schaltflächen, mit denen Sie die von QGIS generierten Standardklassen ändern können:

- Mit der Schaltfläche *Klassifizieren* können die Daten automatisch klassifiziert werden: Alle Werte, die in den Attributen vorkommen und noch nicht in der Tabelle vorhanden sind, werden hinzugefügt.
- Mit Hinzufügen und Löschen können Werte manuell hinzugefügt oder entfernt werden.
- *Alle löschen* entfernt alle Werte aus der Tabelle.

Hinweis

Im *Layer*-Bedienfeld können Sie mit der rechten Maustaste auf einen Klasseneintrag eines Layers klicken, um schnell die Sichtbarkeit der entsprechenden Objekte zu konfigurieren.

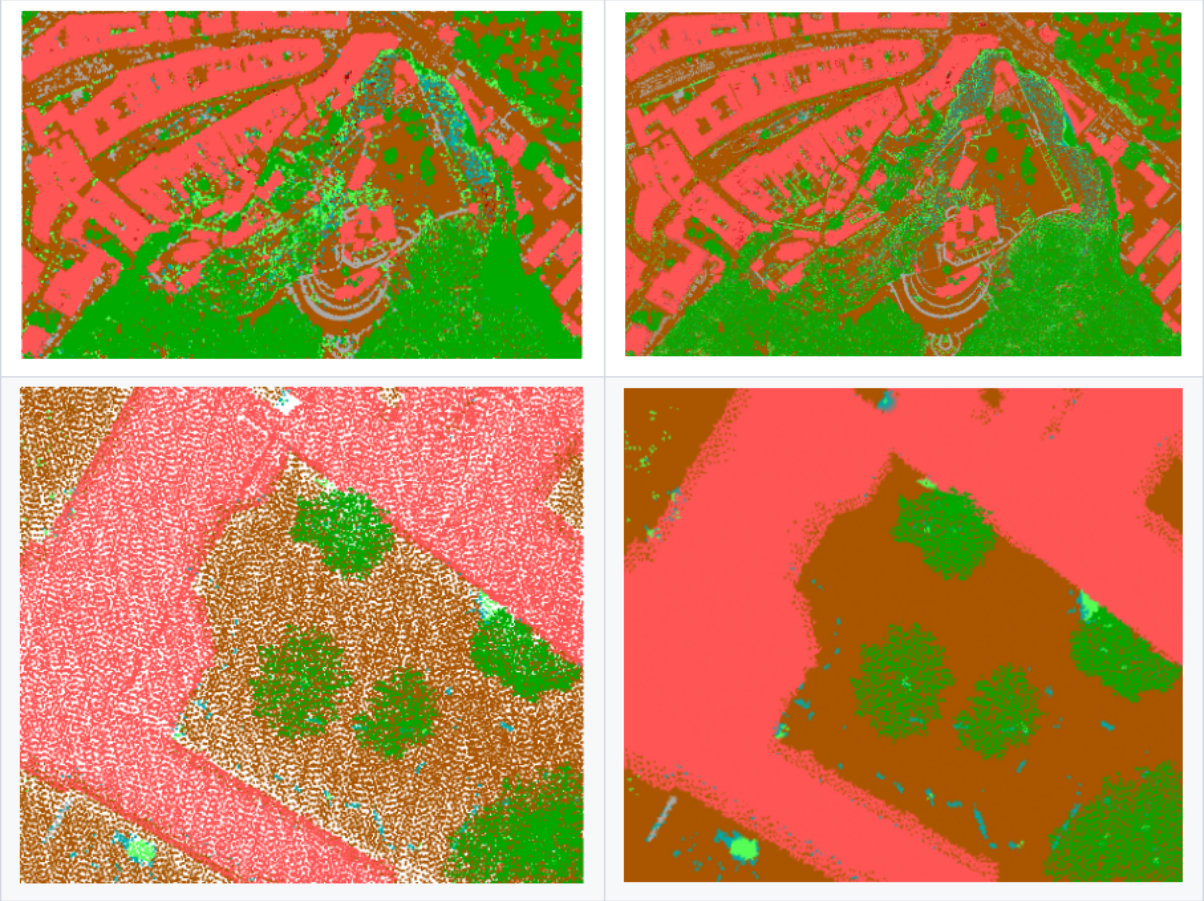


Abb. 16.9: Beispiel für einen Klassifizierungs-Renderer (links) in Kombination mit der Option zur Oberflächentriangulation (rechts)

Punktsymbol

Unter *Punktsymbol* können die Größe und die Einheit (z. B. Millimeter, Pixel, Zoll) eingestellt werden, mit der jeder Datenpunkt angezeigt wird. Als Stil für die Punkte kann entweder *Kreis* oder *Quadrat* ausgewählt werden.

Als Fläche rendern (Triangulieren)

Aktivieren Sie das Kontrollkästchen „Als Fläche rendern (Triangulieren)“, um die Triangulierung des Punktwolken-Layers in der 2D-Ansicht zu aktivieren. Mit dieser Option können Dreiecke anstelle von Punkten gerendert werden. Jeder Punkt behält seine Farbe für die Interpolation im Dreieck bei. Sie können die horizontale Länge der berechneten Dreiecke steuern:

Durch Aktivieren des Kontrollkästchens „Dreiecke mit einer Länge von mehr als“ und Festlegen des Schwellenwerts können Sie die maximale Seitenlänge der Dreiecke festlegen, die im horizontalen Grundriss berücksichtigt werden sollen. Dies kann besonders nützlich sein, wenn Sie tatsächliche Lücken in den Daten identifizieren möchten.

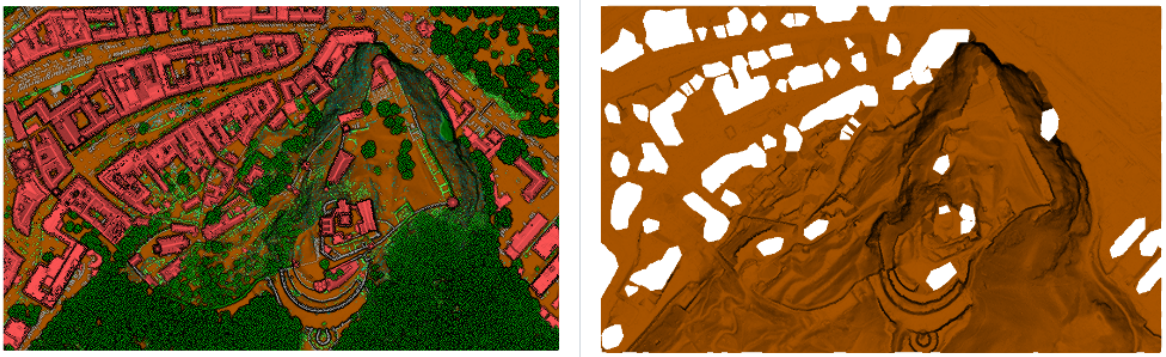


Abb. 16.10: Darstellung von Daten als Oberfläche mit Kartenschattierung (links) und mit Kartenschattierung und Filterung großer Dreiecke (rechts)

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.

Layer-Rendering

Im Abschnitt *Layer Rendering* haben Sie die folgenden Optionen, um die Darstellung der Layer zu ändern:

- *Zeichnungsreihenfolge*: Ermöglicht die Steuerung, ob die Reihenfolge der Darstellung von Punktwolken in der 2D-Kartenansicht von ihrem Z-Wert abhängen soll. Es ist möglich, Folgendes darzustellen:
  - mit der *Standard*-Reihenfolge, in der die Punkte in der Ebene gespeichert sind,
  - von *Von unten nach oben* (Punkte mit größeren Z-Werten überdecken niedrigere Punkte, wodurch das Aussehen eines echten Orthofotos entsteht),
  - oder *Von oben nach unten*, wobei die Szene von unten betrachtet erscheint.
- *Maximaler Fehler*: Punktwolken enthalten in der Regel mehr Punkte, als für die Anzeige erforderlich sind. Mit dieser Option legen Sie fest, wie dicht oder weitläufig die Anzeige der Punktwolke sein soll (dies kann auch als „maximal zulässiger Abstand zwischen den Punkten" verstanden werden). Wenn Sie einen großen Wert (z. B. 5 mm) einstellen, sind zwischen den Punkten Lücken sichtbar. Ein niedriger Wert (z. B. 0,1 mm) kann dazu führen, dass eine unnötige Anzahl von Punkten gerendert wird, was das Rendern verlangsamt (es können verschiedene Einheiten ausgewählt werden).
- *Transparenz*: Mit diesem Werkzeug können Sie die darunterliegende Layer in der Kartenansicht sichtbar machen. Verwenden Sie den Schieberegler, um die Sichtbarkeit Ihres Layers an Ihre Bedürfnisse anzupassen. Sie können auch den Prozentsatz der Sichtbarkeit im Menü neben dem Schieberegler genau festlegen.
- *Mischmodus*: Mit diesem Werkzeug können Sie spezielle Rendering-Effekte erzielen. Die Pixel Ihrer überlagernden und darunterliegenden Layer werden anhand der unter *Mischmodi* beschriebenen Einstellungen gemischt.

- *Eye Dome Lighting*: Wendet Schattierungseffekte auf die Kartenansicht an, um eine bessere Tiefenwiedergabe zu erzielen. Die Wiedergabequalität hängt von der Eigenschaft :ref:„Zeichnungsreihenfolge 1“ ab. Die Zeichnungsreihenfolge :guilabel:„Standard“ kann zu suboptimalen Ergebnissen führen. Folgende Parameter können gesteuert werden:
  - *Stärke*: erhöht den Kontrast und ermöglicht eine bessere Tiefenwahrnehmung
  - *Abstand*: stellt den Abstand der verwendeten Pixel vom mittleren Pixel dar und bewirkt, dass die Ränder dicker werden.

16.2.4 3D-Ansicht Eigenschaften

Auf der Registerkarte *3D-Ansicht* können Sie die Einstellungen für die Darstellung der Punktwolke in 3D-Karten vornehmen.

3D-Rendering-Modi

Die folgenden Optionen können aus dem Dropdown-Menü oben auf der Registerkarte ausgewählt werden:

- *Keine Darstellung*: Daten werden nicht angezeigt
- *2D-Symbologie folgen*: Synchronisiert die Darstellung von Features in 3D mit der *in 2D zugewiesenen Symbologie*.
- *Einzelne Farbe*: Alle Punkte werden unabhängig von ihren Attributen in derselben *Farbe* angezeigt.
- :guilabel:„Attribute nach Farbverlauf“: Interpoliert ein bestimmtes Attribut über einen Farbverlauf und weist den Objekten die passende Farbe zu. Siehe *Attribute nach Farbverlauf*.
- *RGB*: Verwenden Sie verschiedene Attribute der Objekte, um die ihnen zuzuweisenden Rot-, Grün- und Blau-Farbkomponenten festzulegen. Siehe *RGB Renderer*.
- *Klassifizierung*: Unterscheidet Punkte anhand eines Attributs über ihre Farbe. Siehe *Klassifizierungs-Renderer*.

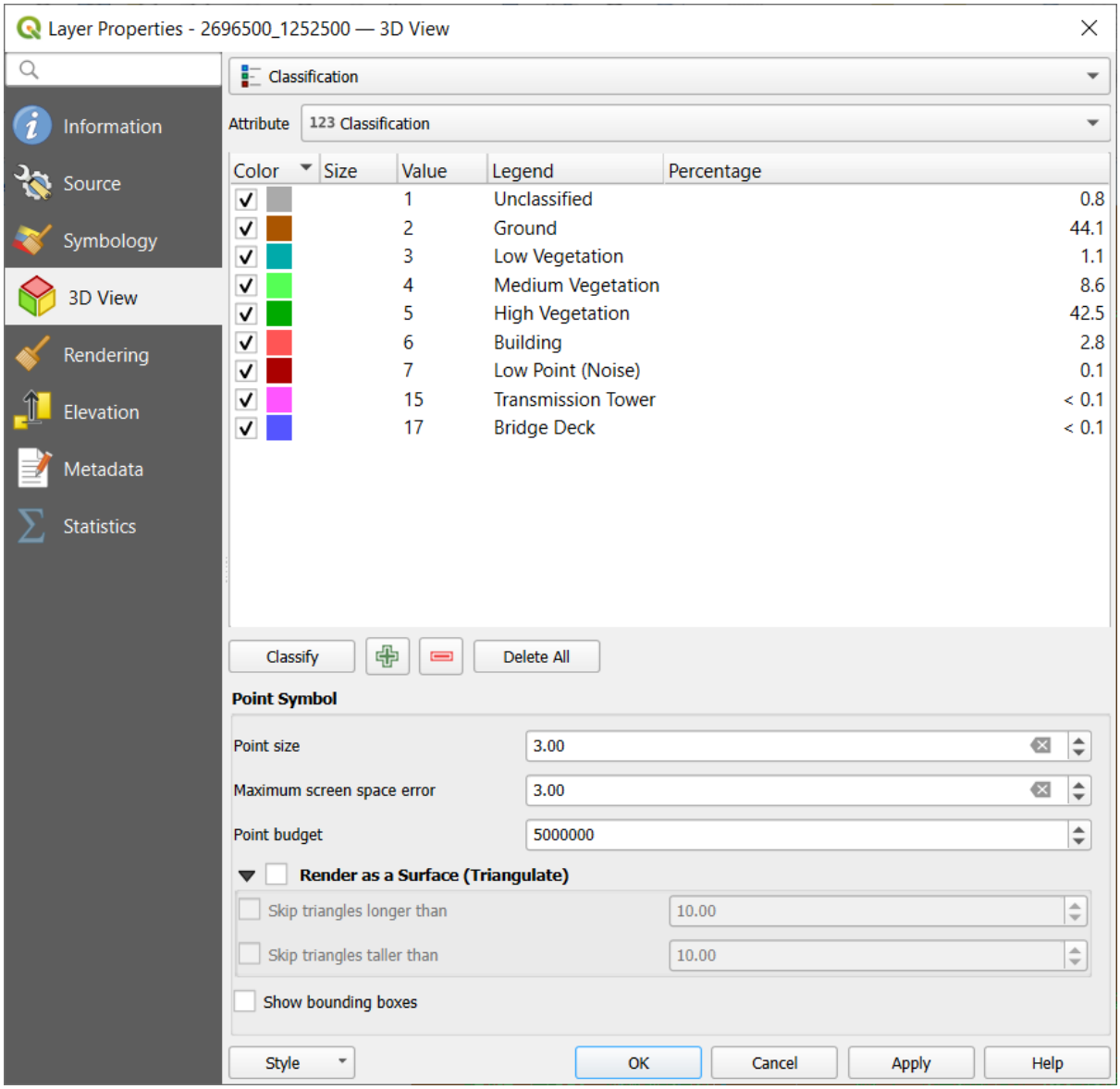


Abb. 16.11: Der Reiter 3D-Ansicht von Punktwolken mit dem Klassifizierungs-Renderer

3D-Punktsymbol

Im unteren Teil der Registerkarte *3D-Ansicht* finden Sie den Abschnitt *Punktsymbol*. Hier können Sie allgemeine Einstellungen für den gesamten Layer vornehmen, die für alle Renderer gleich sind. Es gibt folgende Optionen:

- *Punktgröße*: Die Größe (in Pixeln), mit der jeder Datenpunkt angezeigt wird, kann eingestellt werden.
- *Maximaler Bildschirmfehler*: Mit dieser Option legen Sie fest, wie dicht oder weitläufig die Punktwolke angezeigt wird (in Pixeln). Wenn Sie einen hohen Wert (z. B. 10) einstellen, entstehen sichtbare Lücken zwischen den Punkten; ein niedriger Wert (z. B. 0) kann dazu führen, dass unnötig viele Punkte gerendert werden müssen, was das Rendern verlangsamt (weitere Details finden Sie unter *Symbology Maximaler Fehler*).
- *Punktbudget*: Um lange Renderzeiten zu vermeiden, können Sie die maximale Anzahl der zu rendernden Punkte festlegen.
- Aktivieren Sie das Kontrollkästchen *Als Fläche rendern (Triangulieren)*, um den Punktwolken-Layer in der 3D-Ansicht mit einer durch Triangulation erhaltenen festen Fläche zu rendern. Sie können die Abmessungen der berechneten Dreiecke steuern:

  - *Dreiecke überspringen, die länger sind als* legt ein Schwellenwert in der horizontalen Ebene fest, der die maximale Länge einer Seite der zu berücksichtigenden Dreiecke beschreibt.
  - *Dreiecke überspringen, die höher als* ein Schwellenwert sind: Legt in der vertikalen Ebene die maximale Höhe einer Seite der zu berücksichtigenden Dreiecke fest.

- *Begrenzungsrahmen anzeigen*: Besonders nützlich für die Fehlersuche, zeigt Begrenzungsrahmen von Knoten in der Hierarchie an.

16.2.5 Darstellung

Unter dem Gruppenfeld *Massstabsabhängige Sichtbarkeit* können Sie den *Maximalwert (inklusive)* und den *Minimalwert (exklusiv)* festlegen und so einen Massstabsbereich definieren, in dem Objekte sichtbar sind. Außerhalb dieses Bereichs werden sie ausgeblendet. Mit der Schaltfläche :sup:„Auf aktuellen Kartenmaßstab setzen“ können Sie den aktuellen Maßstab der Kartenansicht als Grenze für den Sichtbarkeitsbereich verwenden. Weitere Informationen finden Sie unter *Einstellen des Sichtbarkeitsmaßstabs*.

Bemerkung

Sie können die skalierungsabhängige Sichtbarkeit eines Layers auch über das Bedienfeld *Layer* aktivieren: Klicken Sie mit der rechten Maustaste auf den Layer und wählen Sie im Kontextmenü *Maßstabsabhängige Sichtbarkeit des Layers setzen*.

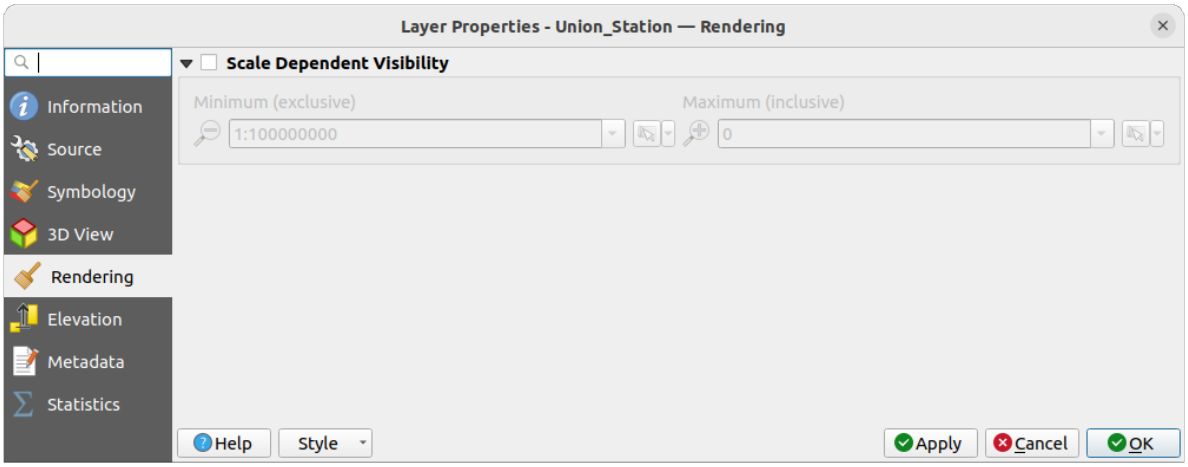


Abb. 16.12: Der Punktwolken-Rendering-Reiter

16.2.6 Höhe

Auf dem Reiter *Höhe* können Sie Korrekturen für die Z-Werte der Daten vornehmen. Dies kann erforderlich sein, um die Höhe der Daten in 3D-Karten und deren Darstellung in den *Höhen-Diagrammen* anzupassen. Es stehen folgende Optionen zur Verfügung:

- *Vertikales Referenzsystem*: Wenn das KBS Ihres Punktewolken-Layers ein zusammengesetztes KBS ist (einschließlich einer Z-Dimension), wird das für den Layer verwendete vertikale KBS automatisch aus der vertikalen Komponente des KBS des Layers abgeleitet. In diesem Fall können Sie kein anderes vertikales KBS manuell festlegen, und die Option zum Ändern wird deaktiviert. Wenn Ihr Punktwolken-Layer ein horizontales (2D) KBS verwendet (was jedoch selten vorkommt), können Sie manuell ein bestimmtes vertikales KBS auswählen, indem Sie auf KBS auswählen klicken. Vertikale Referenzsysteme werden für Punktwolken-Layer unterstützt in:

  - |newElevationProfile| *Geländehöhenprofilen <label_elevation_profile_view>*
  - *Objekte-abfragen-Werkzeug*
  - *3D-Kartenansichten*

- Unter der Gruppe *Höhe*:
  - können Sie eine *Skalierung* festlegen: Wenn Sie hier den Wert `10` eingeben, wird ein Punkt mit dem Wert Z = `5` in einer Höhe von `50` angezeigt.
  - Ein *Offset* zum Z-Level kann eingegeben werden. Dies ist nützlich, um verschiedene Datenquellen in ihrer Höhe aufeinander abzustimmen. Standardmäßig wird der niedrigste in den Daten enthaltene Z-Wert als dieser Wert verwendet. Dieser Wert kann auch mit der Schaltfläche Aktualisieren am Ende der Zeile wiederhergestellt werden.
- Unter *Genauigkeit des Profil-Diagramms* hilft Ihnen die Option *Maximaler Fehler* dabei, zu steuern, wie dicht oder verstreut die Punkte im Höhenprofil dargestellt werden. Größere Werte führen zu einer schnelleren Generierung mit weniger Punkten.
- Unter *Darstellung des Profil-Diagramms* können Sie die Punktanzeige steuern:
  - *Punktgröße*: Die Größe, mit der die Punkte gerendert werden sollen, in unterstützten Einheiten (Millimeter, Karteneinheiten, Pixel, …)
  - *Stil*: Gibt an, ob die Punkte als *Kreis* oder *Quadrat* dargestellt werden sollen.
  - Verwenden Sie eine einzelne *Farbe* für alle Punkte, die in der Profilansicht sichtbar sind.
  - Aktivieren Sie das Kontrollkästchen *Farben des Layers berücksichtigen* , um stattdessen die Punkte mit der Farbe anzuzeigen, die ihnen über ihre 2D-Symbologie 1 zugewiesen wurde.
  - *Transparenz anhand der Entfernung vom Kurveneffekt anwenden*, wodurch die Transparenz von Punkten reduziert wird, die weiter von der Profilkurve entfernt sind

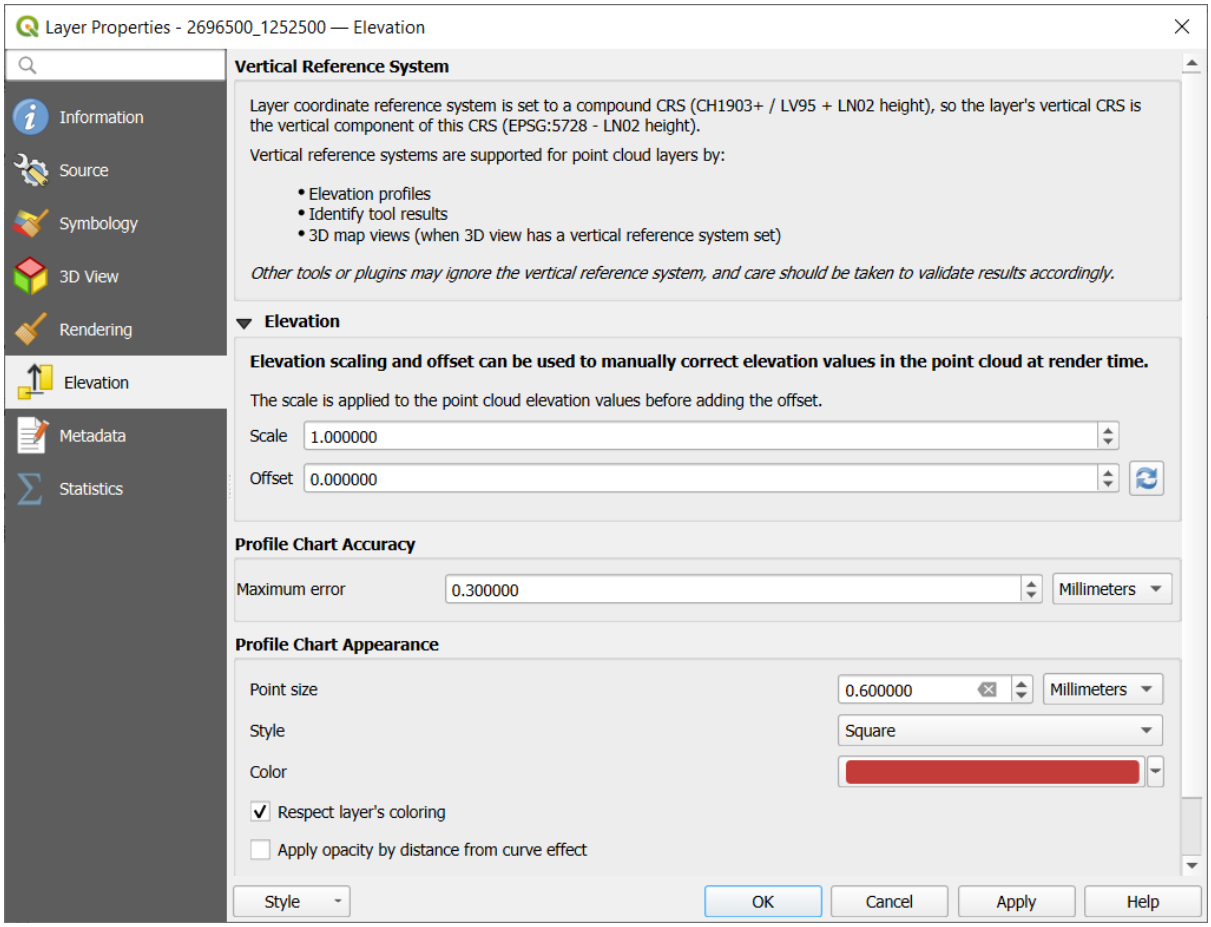


Abb. 16.13: Der Punktwolken-Höhe-Reiter

16.2.7 Metadaten

Der Reiter *Metadata* bietet Ihnen Optionen zum Erstellen und Bearbeiten eines Metadatenberichts zu Ihrem Layer. Siehe *Metadaten* für weitere Informationen.

16.2.8 Statistik

Auf der Registerkarte *Statistik* erhalten Sie einen Überblick über die Attribute Ihrer Punktwolke und deren Verteilung.

Oben finden Sie den Abschnitt *Attributstatistik*. Hier sind alle in der Punktwolke enthaltenen Attribute sowie einige ihrer statistischen Werte aufgeführt: *Minimum*, *Maximum*, *Mittelwert*, *Standardabweichung*

Wenn es ein Attribut *Classification* gibt, dann gibt es im unteren Bereich eine weitere Tabelle. Hier werden alle im Attribut enthaltenen Werte sowie ihre absolute *Anzahl* und relative % -Häufigkeit aufgelistet.

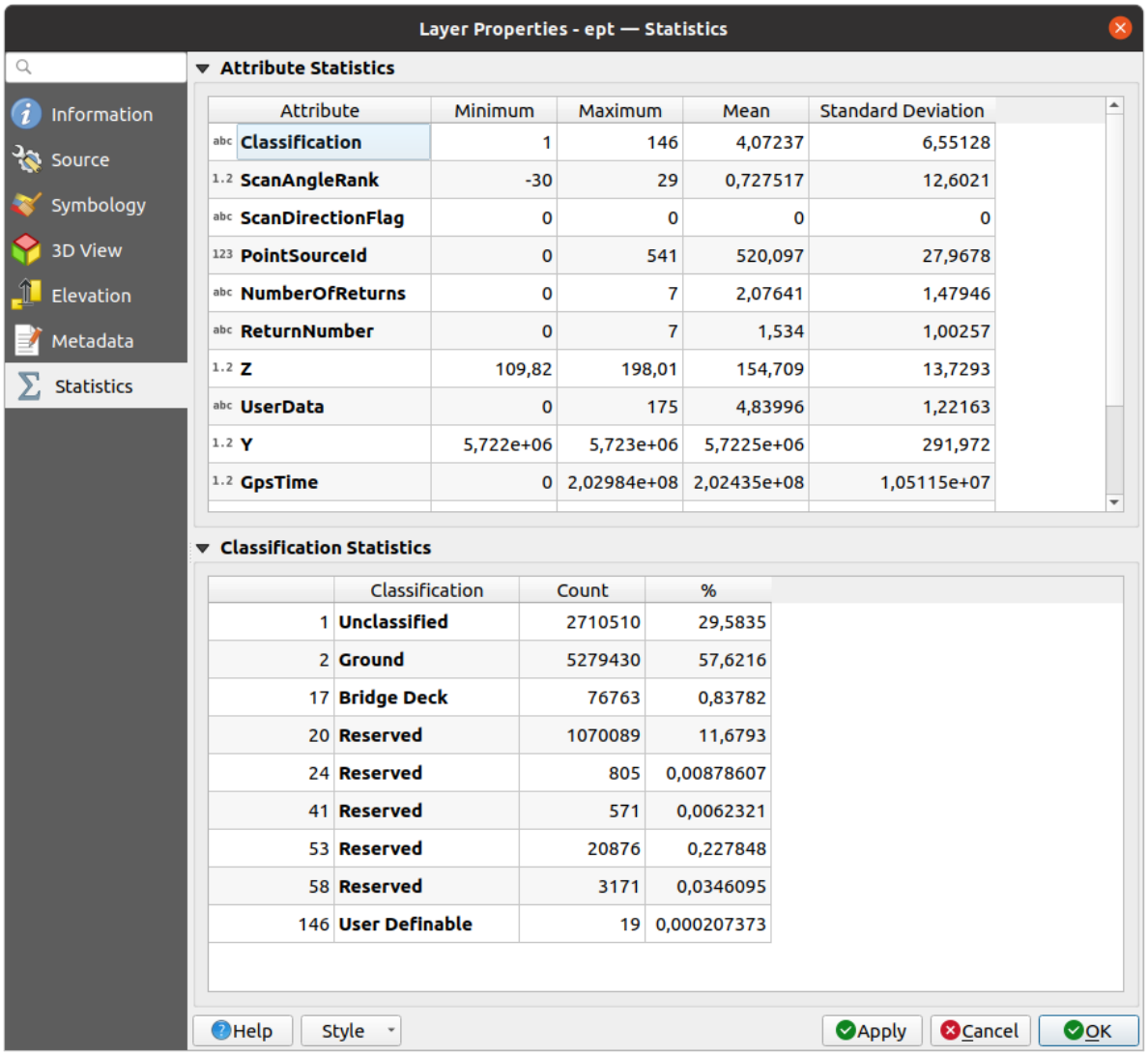


Abb. 16.14: Der Punktwolken-Statistik-Reiter

16.3 Virtuelle Punktwolke

Lidar-Vermessungen größerer Gebiete sind oft mehrere Terabyte große Datensätze mit vielen Milliarden Punkten. Solch große Datensätze als einzelne Punktwolken-Datei darzustellen, ist aufgrund der Schwierigkeiten bei der Speicherung, Übertragung, Anzeige und Analyse nicht praktikabel. Punktwolken-Daten werden daher in der Regel in quadratische Kacheln (z. B. `1 km x 1 km`) aufgeteilt gespeichert und verteilt, wobei jede Kachel eine besser handhabbare Dateigröße hat (z. B. ~200 MB im komprimierten Zustand).

Das Kacheln von Daten löst zwar das Problem der Datengröße, führt jedoch zu Problemen bei der Verarbeitung oder Anzeige eines Bereichs von Interesse, der nicht vollständig in eine einzelne Kachel passt. Benutzer müssen Workflows entwickeln, die mehrere Kacheln berücksichtigen, und es muss besonders darauf geachtet werden, wie mit Daten in der Nähe der Kachelränder umgegangen wird, um unerwünschte Artefakte in den Ausgaben zu vermeiden. Ebenso wird es bei der Anzeige von Punktwolkendaten mühsam, viele einzelne Dateien zu laden und dieselbe Symbolik anzuwenden.

Hier ist ein Beispiel für mehrere in QGIS geladene Punktwolkenkacheln. Jede Kachel wird basierend auf den minimalen/maximalen Z-Werten der Kachel gestaltet, wodurch sichtbare Artefakte an den Kachelkanten entstehen. Die Gestaltung muss für jede Ebene separat angepasst werden:

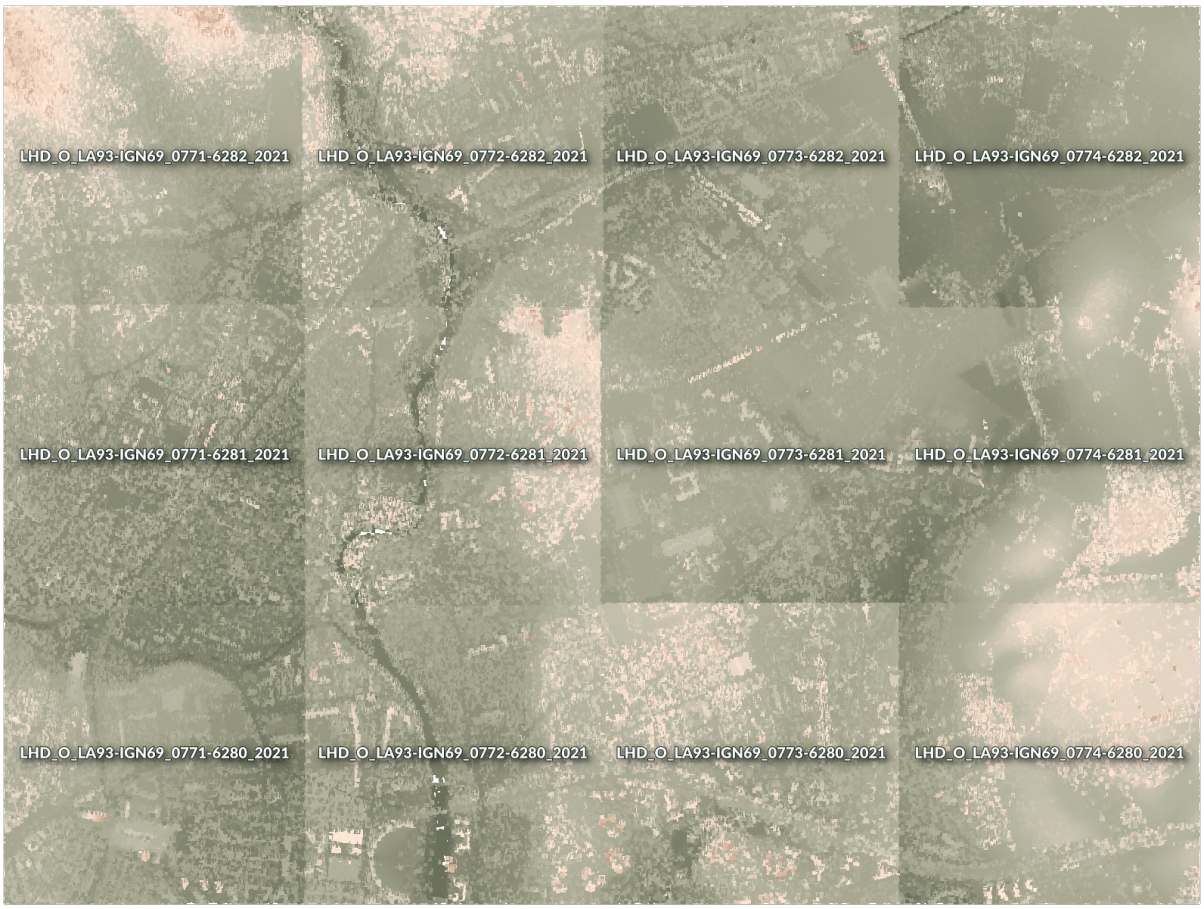


Abb. 16.15: Verschiedene Punktwolken-Kacheln mit Artefakten an den Kanten

In der Welt der GIS-Software sind viele Nutzer mit dem Konzept der virtuellen Raster vertraut. Ein virtuelles Raster ist eine Datei, die lediglich auf andere Rasterdateien mit tatsächlichen Daten verweist. Auf diese Weise behandelt die GIS-Software den gesamten Datensatz, der aus vielen Dateien besteht, als einen einzigen Raster-Layer, was die Anzeige und Analyse aller in der virtuellen Datei aufgeführten Raster erheblich vereinfacht.

Das Konzept der virtuellen Raster wurde von GDAL für Punktwolken übernommen. **Virtuelle Punktwolken (VPC)** sind ein Dateiformat, das auf andere Punktwolken-Dateien verweist. Software, die virtuelle Punktwolken unterstützt, behandelt den gesamten gekachelten Datensatz als eine einzige Datenquelle.

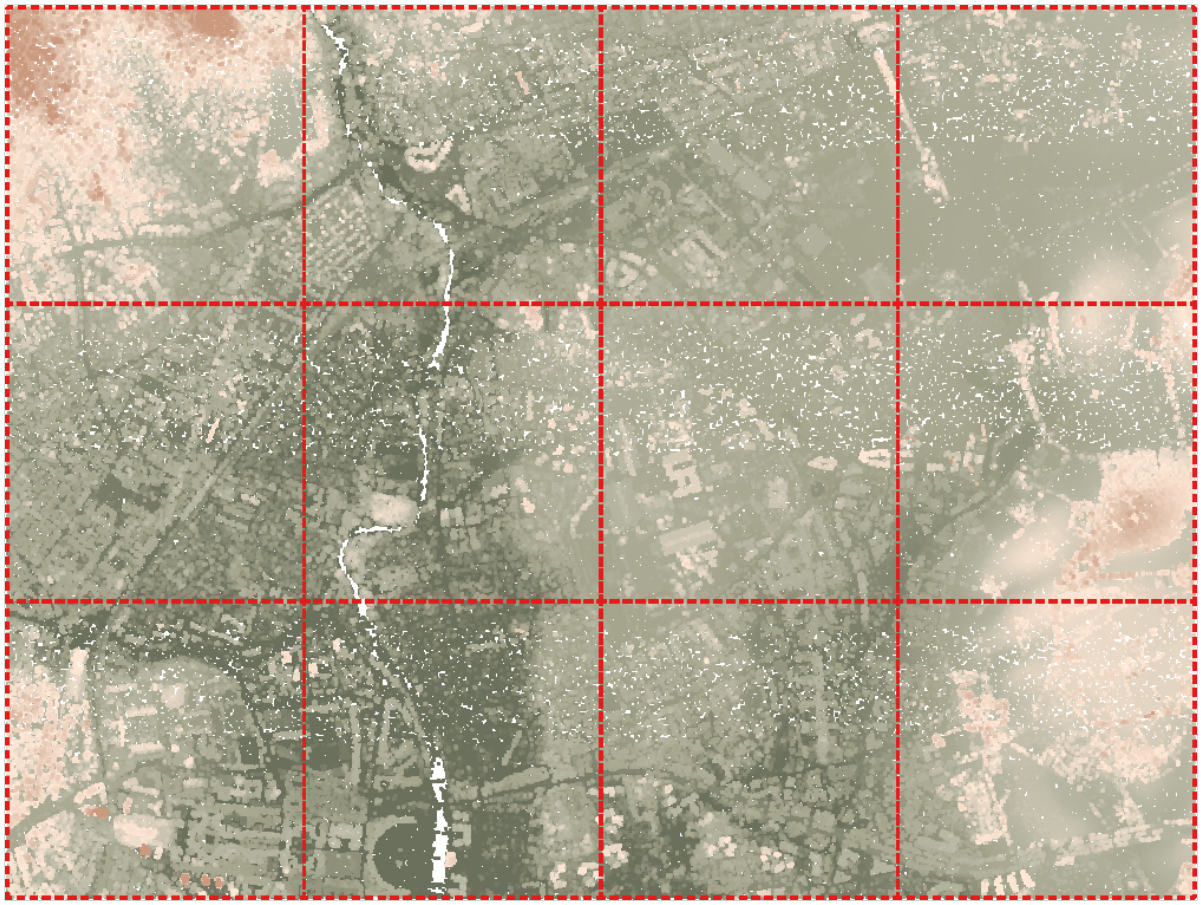


Abb. 16.16: Eine virtuelle Punktwolke

Die Anzeige und Bearbeitung virtueller Punktwolken ist wesentlich flüssiger und einfacher.

Im Kern ist eine virtuelle Punktwolken-Datei eine einfache JSON-Datei mit der Erweiterung `.vpc`, die Verweise auf tatsächliche Datendateien (z. B. `.LAS`-, `.LAZ`- oder `.COPC`-Dateien) und zusätzliche Metadaten enthält, die aus den Dateien extrahiert wurden. Auch wenn es möglich ist, VPC-Dateien von Hand zu schreiben, wird dringend empfohlen, sie mit einem automatisierten Tool zu erstellen, wie z. B.

- Der Verarbeitungsalgorithmus *Virtuelle Punktwolke aufbauen (VPC)*
- Der Befehl `build_vpc` des Tools PDAL wrench

Weitere Informationen finden Sie in der VPC-Spezifikation, die auch Best Practices und optionale Erweiterungen (z. B. Übersichten) enthält.

KAPITEL 17

Arbeiten mit 3D Tiles

17.1 Was sind 3D Tiles?

3D Tiles sind eine Spezifikation für das Streamen und Visualisieren von großformatigen 3D Geodatensätzen. Grundlegend ist eine hierarchische Datenstruktur, die es möglich macht 3D Daten effizient zu verwalten und darzustellen. Die Performance wird optimiert durch das dynamische nachladen von entsprechenden Detailstufen. Diese Technologie wird häufig in den Bereichen Stadtplanung, Architektur, Simulation, Gaming und Virtual Reality eingesetzt und bietet eine standardisierte und interoperable Lösung für die Visualisierung komplexer geografischer Daten.

Abb. 17.1: Beispiele für 3D Tiles

Siehe *Verwendung von 3D-Kachelszenendiensten* für Erläuterungen wie Sie diese Datenquellen in QGIS hinzufügen.

17.1.1 State of 3D Tiles Support

Momentan unterstützt QGIS zwei Formate von 3D Tiles:

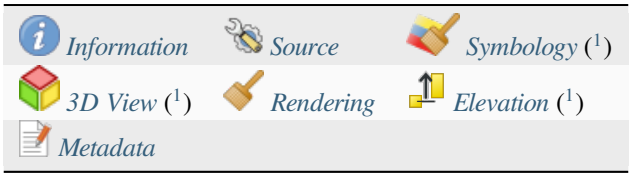
- „Cesium“ 3D Tiles are used primarily for complex 3D models of buildings or whole cities. Such datasets can be provided by cloud-based platforms such as Cesium Ion or Google (*Photorealistic 3D Tiles*).
- Quantized Mesh tiles are used for terrain elevation data.

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 3D Tiles Eigenschaften

Der 3D Tiles *Layer Properties* Dialog umfasst die folgenden Bereiche:



17.2.1 Information

Der Tab *Information* kann nicht editiert werden und ist ein guter Platz, um schnell zusammengefasste Informationen und Metadaten des aktuellen Layers nachzuschlagen. Hier erhalten Sie folgende Informationen:

- je nach Datenanbieter des Layers: Name, URL, Quelltyp und -pfad, Anzahl der Zoomstufen
- Benutzerdefinierte Eigenschaften, die verwendet werden, um zusätzliche Informationen über den Layer im aktiven Projekt zu speichern. Weitere Eigenschaften können mit PyQGIS erstellt und verwaltet werden, insbesondere mit der Methode setCustomProperty().
- das Koordinatenbezugssystem: Name, Einheiten, Methode, Genauigkeit, Referenz (d.h., ob es statisch oder dynamisch ist)
- aus den *Metadaten* werden weitere Informationen gezogen: Zugriff, Ausdehnung, weiterführende Links, Kontaktangaben und Versionen

17.2.2 Quelle

Der *Source* Tab zeigt Basisinformationen über das ausgewählte 3D Tile, darunter:

- der *Layername*, der im *Layer*-Bedienfeld angezeigt wird;
- das *Koordinatenreferenzsystem*: Zeigt das *Koordinatenreferenzsystem (KBS)* des Layers an. Sie können das KBS des Layers ändern, indem Sie ein kürzlich verwendetes KBS aus der Dropdown-Liste auswählen oder auf die Schaltfläche KBS wählen klicken (siehe *Auswahl des Koordinatenbezugssystems*). Verwenden Sie dieses Verfahren nur, wenn das KBS falsch oder nicht angegeben ist.

[1] Also available in the *Layer styling panel*

17.2.3 Symbolisierung

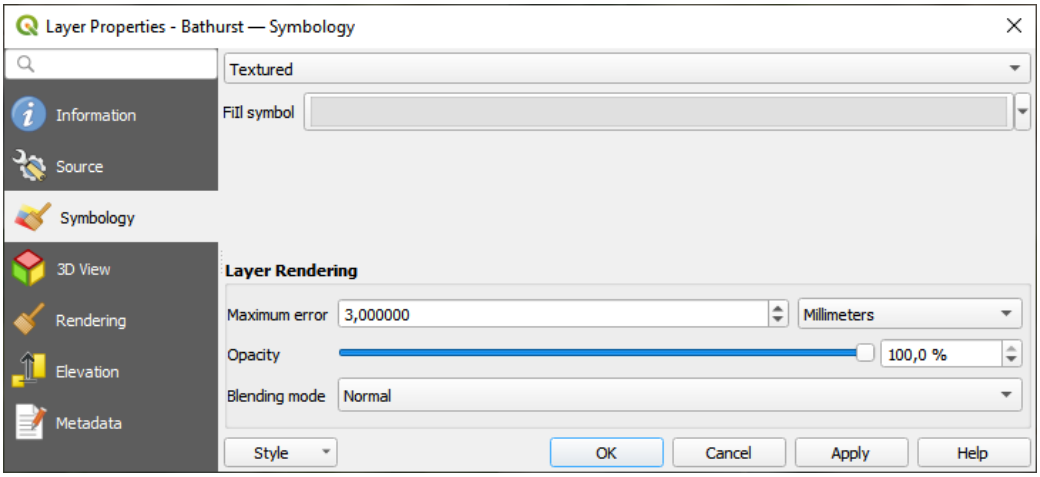


Abb. 17.2: 3D Tile Layer Symbology

Standardmäßig wird die Ebene mit einer Textur gestaltet, aber Sie können dies ändern, um das Drahtgitter hinter der Szene anzuzeigen, indem Sie im Dropdown-Menü *Wireframe* auswählen. Sie können auch die Füllung und die Liniensymbole des Gitters ähnlich wie bei den Vektorpolygonen ändern. Wenn Sie das Kontrollkästchen *Use texture colors* aktivieren, wird jedes Netz-Element mit dem Durchschnittswert der gesamten Textur gerendert. Dies ist eine gute Option, wenn Sie mit einem großen Datensatz arbeiten und sich einen schnellen Überblick über die Daten verschaffen möchten.

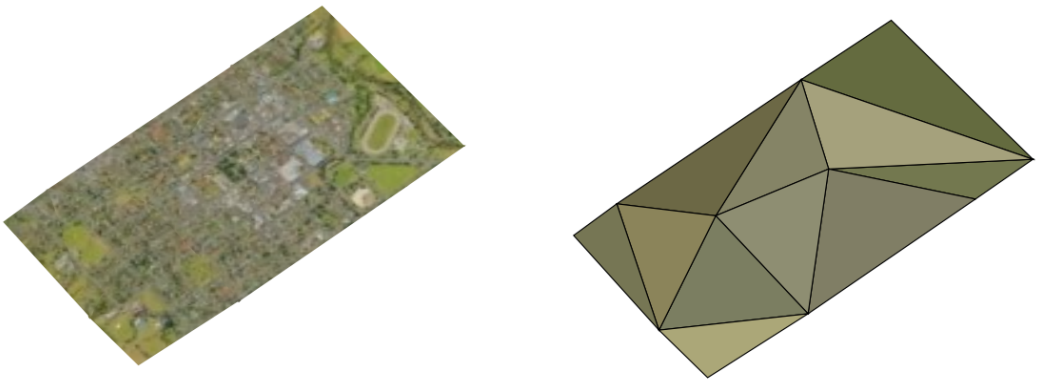


Abb. 17.3: 3D Tiles - textured and wireframe

Auf der Registerkarte *Symbolisierung* können Sie auch einige Optionen einstellen, die sich immer auf alle Objekte des Layers auswirken:

- *Maximum error*: This parameter determines the level of detail displayed in the 3D model. Similar to point clouds, 3D tiles often contain more information than necessary for visual representation. By adjusting this setting, you control the balance between display density and rendering speed. A larger value (e.g., 5 mm) may introduce noticeable gaps between elements, while a smaller value (e.g., 0.1 mm) could lead to the rendering of an excessive number of details, potentially slowing down the rendering process. Different units can be selected to tailor the setting to your specific needs.

- *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.
- *Mischmodus*: Mit diesen Werkzeugen können Sie spezielle Rendering-Effekte erzielen, die Sie bisher nur von Graphikprogrammen her kennen. Die Pixel von sich überlagernden Layern werden durch die verwendete Einstellungen gemischt, wie in *Mischmodi* beschrieben.

17.2.4 3D-Ansicht Eigenschaften

- *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).

To view the data you can open *New 3D map view*.

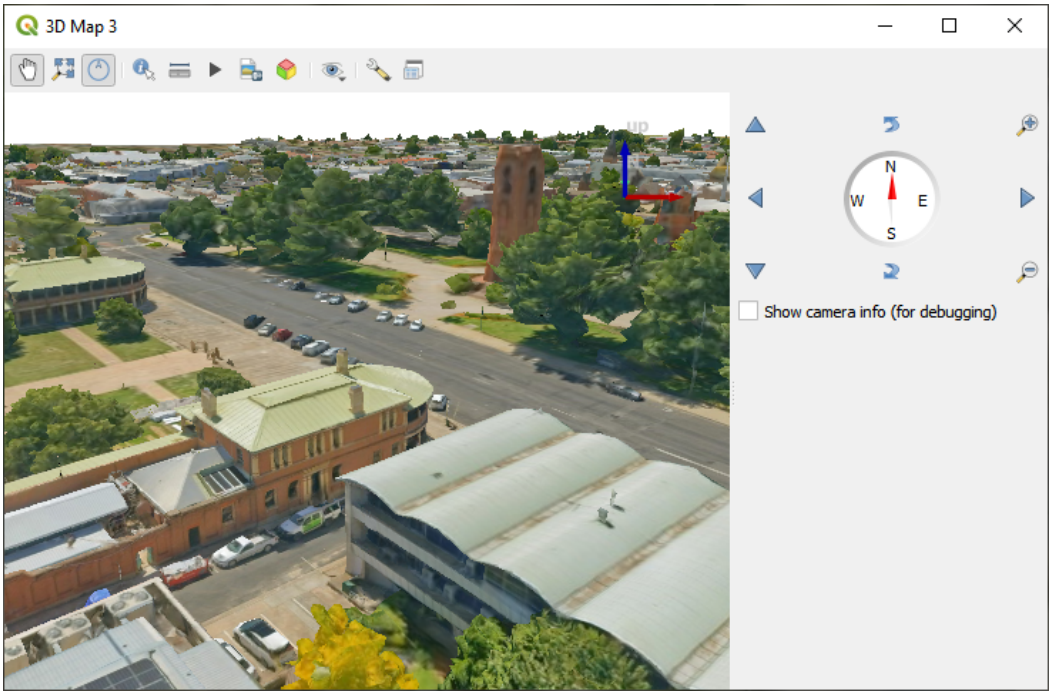


Abb. 17.4: 3D-Kartenansicht

17.2.5 Darstellung

Unter *Maßstabsabhängige Sichtbarkeit* können Sie die Maßstäbe *Maximum (inklusive)* und *Minimum (exklusive)* festlegen und damit einen Bereich von Maßstäben definieren, in dem die Objekte des Layers sichtbar sind. Außerhalb dieses Bereichs werden sie ausgeblendet. Die Schaltfläche Auf aktuellen Kartenmaßstab setzen hilft Ihnen, den aktuellen Maßstab der Kartenansicht als Grenze für den Sichtbarkeitsbereich zu verwenden. Siehe *Einstellen des Sichtbarkeitsmaßstabs* für weitere Informationen.

17.2.6 Höhe

The *Elevation* tab provides options to control the layer elevation properties within a *3D map view*. Specifically, you can set:

- *Elevation Surface*: how the 3D layer vertices Z values should be interpreted as terrain elevation. You can apply a *Scale* factor and an *Offset*.

17.2.7 Metadaten

Der Reiter *Metadata* bietet Ihnen Optionen zum Erstellen und Bearbeiten eines Metadatenberichts zu Ihrem Layer. Siehe *Metadaten* für weitere Informationen.

KAPITEL 18

Kartenlayout

Mit Drucklayouts und Berichten können Sie Karten und Kartenserien (Atlas) erzeugen und sie entweder ausdrucken oder als Bild, PDF- oder SVG-Datei speichern.

18.1 Überblick über das Druck-Layout

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 Beispielsitzung für Einsteiger

Bevor Sie mit dem Druck-Layout arbeiten, müssen Sie einige Raster- oder Vektor-Layer in die QGIS Kartenansicht laden und deren Eigenschaften nach Ihren Wünschen anpassen. Nachdem alles nach Ihren Wünschen dargestellt und symbolisiert wurde, klicken Sie auf das Symbol Neues Drucklayout in der Werkzeugleiste *Projekt* oder wählen Sie *Projekt* ► *Neues Drucklayout*. Sie werden aufgefordert, einen Titel für das neue Layout zu wählen.

Um zu zeigen, wie man eine Karte erstellt, folgen Sie den nächsten Anweisungen.

1. Wählen Sie auf der linken Seite der Werkzeugleiste die Schaltfläche „Karte hinzufügen“ und ziehen Sie mit gedrückter linker Maustaste ein Rechteck auf die Kartenansicht. Innerhalb des gezeichneten Rechtecks wird die Kartenansicht aus dem Hauptfenster im Layout angezeigt.

2. Wählen Sie die Schaltfläche Maßstab hinzufügen und klicken Sie mit der linken Maustaste auf die Kartenansicht des Drucklayouts. Eine Maßstabsleiste wird der Kartenansicht hinzugefügt.

3. Wählen Sie in der Symbolleiste die Schaltfläche Legende hinzufügen und ziehen Sie mit gedrückter linker Maustaste ein Rechteck auf die Kartenansicht. Innerhalb des gezeichneten Rechtecks wird die Legende gezeichnet.

4. Mit dem Element auswählen/verschieben Werkzeug können Sie die hinzugefügten Elemente im Karten-Layout auswählen und dann verschieben.
5. Wenn ein Kartenelement ausgewählt ist, können Sie auch dessen Größe verändern. Dazu klicken Sie in eins der kleinen weißen Rechtecke am Rand des Kartenelements, die dann zu sehen sind, halten die linke Maustaste gedrückt und verschieben Sie das Rechteck an einen neuen Ort.
6. Klicken Sie auf das Bedienfeld *Elementeigenschaften* auf der rechten Seite unten und suchen Sie den Bereich *Haupteigenschaften*. Ändern Sie den Wert der Einstellung *Kartendrehung* auf ‚15.00°'. Sie sollten sehen, dass sich die Ausrichtung der Inhalte im Kartenelement ändert.
7. Jetzt können Sie Ihr Drucklayout mit den Exportwerkzeugen im Menü *Layout* ausdrucken oder in ein Bildformat, nach PDF oder SVG exportieren.
8. Schließlich können Sie Ihr Drucklayout mit der Schaltfläche Projekt speichern innerhalb der Projektdatei speichern.

Sie können dem Drucklayout mehrere Elemente hinzufügen. Es ist auch möglich, mehr als eine Kartenansicht, Legende oder Maßstabsleiste in der Kartenansicht des Drucklayouts auf einer oder mehreren Seiten zu haben. Jedes Element hat seine eigenen Eigenschaften und, im Falle der Karte, seinen eigenen Ausschnitt. Wenn Sie Elemente aus der Kartenansicht entfernen möchten, können Sie dies mit der Taste `Entfernen` oder der Taste `Löschen` tun.

18.1.2 Layoutverwaltung

Die *Layout-Verwaltung* ist das Hauptfenster, um Drucklayouts innerhalb des Projekts zu verwalten. Sie gibt Ihnen eine Übersicht über die vorhandenen Layouts und Berichte im Projekt und bietet folgende Werkzeuge:

- nach einem Layout suchen;
- ein neues Drucklayout oder einen neuen Bericht auf der Basis der leeren oder einer bestimmten Vorlage erzeugen oder ein bereits vorhandenes Element duplizieren;
- ein Element umbenennen oder löschen;
- ein Element öffnen.

So öffnen Sie den Layout-Manager-Dialog:

- Im QGIS-Hauptfenster wählen Sie im Menü *Projekt* ► *Layout-Verwaltung…* oder klicken Sie auf die Schaltfläche Layout-Verwaltung in der *Projektwerkzeugleiste*;
- im Drucklayout oder im Berichtsfenster wählen Sie im Menü *Layout* ► *Layout-Verwaltung…* oder klicken Sie auf die Schaltfläche Layouverwaltung in der *Layout-Werkzeugleiste*.

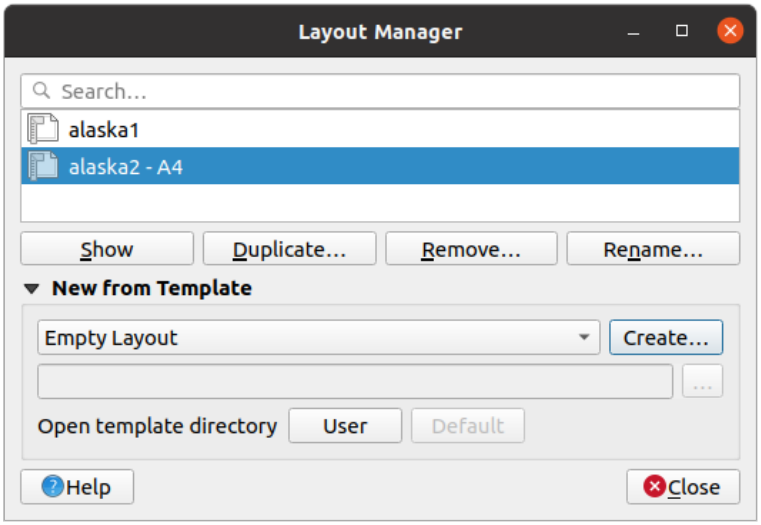


Abb. 18.1: Die Layout-Verwaltung

Die Layout-Verwaltung listet im oberen Teil alle im Projekt verfügbaren Drucklayouts oder Berichte auf und bietet folgende Werkzeuge:

- *Anzeigen*: Sie können mehrere Drucklayouts und/oder Berichte auswählen und sie mit einem Klick öffnen. Ein Doppelklick auf einen Namen öffnet das Element ebenfalls;
- *Duplizieren…* des ausgewählten Drucklayout oder den ausgewählten Bericht (setzt voraus, dass ein einzelner Eintrag ausgewählt ist): erzeugt ein neues Fenster und benutzt das ausgewählte Element als Vorlage. Sie werden dazu aufgefordert, einen neuen Titel zu vergeben;
- *Umbenennen…* des ausgewählten Drucklayouts oder des ausgewählten Berichts (setzt voraus, dass ein einzelner Eintrag ausgewählt ist): Sie werden dazu aufgefordert, einen neuen Titel zu vergeben;
- *Entfernen…* das (oder die) ausgewählte(n) Layout(s) oder Bericht(e) wird (werden) aus dem Projekt gelöscht.

Im unteren Teil ist es möglich, neue Drucklayouts oder Berichte von Grund auf oder mit einer Vorlage zu erstellen. Standardmäßig sucht QGIS nach Vorlagen im *Benutzerprofil* Ordner und in den Verzeichnissen der Anwendungsvorlagen (erreichbar über die beiden Schaltflächen am unteren Rand des Rahmens), aber auch in jedem Ordner, der unter *Einstellungen* ► *Optionen* ► *Layouts* im Bereich *Pfad(e) um nach zusätzlichen Druckvorlagen zu suchen* angegeben ist. Die gefundenen Vorlagen werden in der Combobox aufgelistet. Wählen Sie eine Vorlage aus und drücken Sie die Schaltfläche *Erstellen*, um einen neuen Bericht oder ein neues Drucklayout zu erstellen.

Sie können auch Vorlagen aus einem anderen Verzeichnus benutzen; in diesem Fall wählen Sie *Bestimmtes* aus dem Auswahlfeld, gehen im Dateisystem zur Vorlage und drücken danach *Erzeugen…*.

Tipp

So erzeugen Sie Drucklayouts im Browser-Bedienfeld aus Vorlagen heraus

Das Einfügen einer Vorlagendatei `.qpt` per drag and drop aus dem Dateimanager in das Kartenfenster oder ein Doppelklick darauf im *Browser-Bedienfeld* erzeugt ein neues Drucklayout aus der Vorlagendatei.

18.1.3 Menüs, Werkzeuge und Bedienfelder des Drucklayouts

Wenn Sie ein neues, leeres Drucklayout öffnen, sehen Sie zunächst nur eine leere Karte, die die Abgrenzungen der Fläche darstellt, auf der Elemente beim Export bzw. Druck dargestellt werden. Auf der linken Seite finden Sie Schaltflächen zum Hinzufügen von Drucklayoutelementen: 2D- oder 3D-Karten, Textbeschriftungen, Bilder, Legenden, Maßstabsbalken, Formen, Pfeile, Attribut- oder einfache Tabellen, HTML-Rahmen, Höhenprofile,… In dieser Symbolleiste finden Sie auch Schaltflächen zum Navigieren, Vergrößern eines Bereichs und Schwenken der Ansicht im Layout sowie Schaltflächen zum Auswählen eines beliebigen Layoutelements und zum Verschieben der Ausmaße der Kartenelemente.

Abb. 18.2 zeigt die anfängliche Ansicht des Drucklayouts, bevor irgendwelche Elemente hinzugefügt werden.

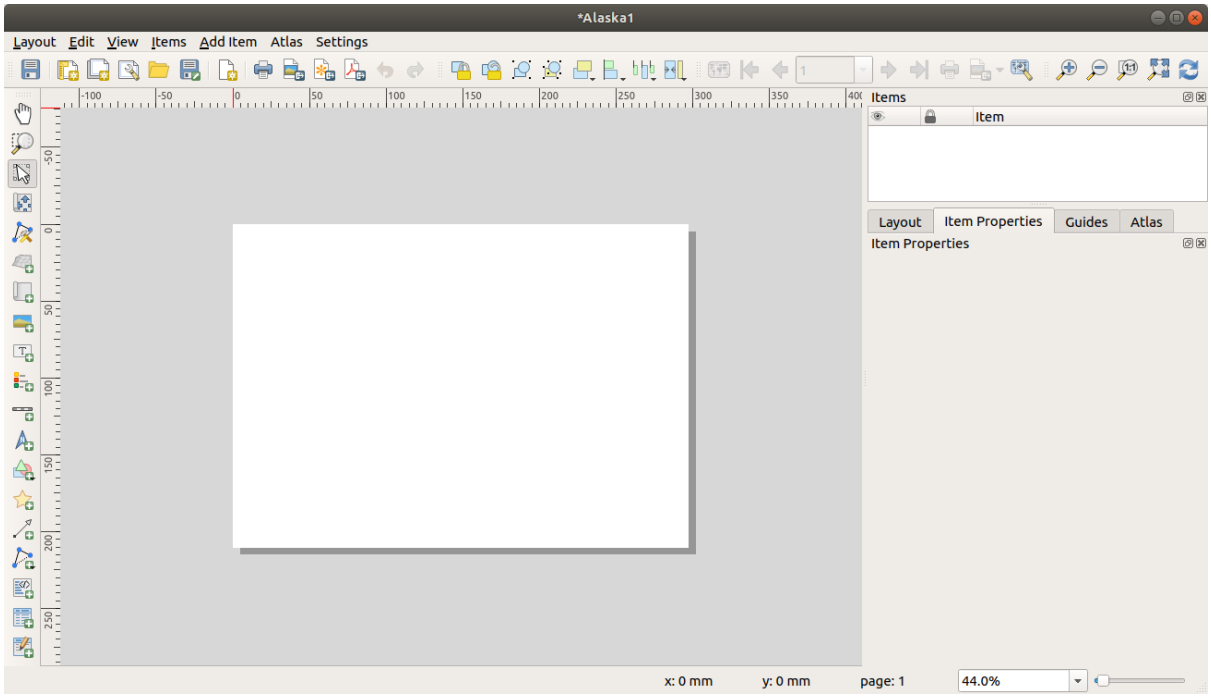


Abb. 18.2: Ein leeres Druck-Layout

Rechts von der Zeichenfläche finden Sie zwei Bedienfelder. Das obere enthält die Reiter *Elemente* und *Rücknahmeprotokoll* und das untere Bedienfeld enthält die Reiter *Layout*, *Elementeigenschaften* und *Führungen*.

- Der Reiter *Elemente* enthält eine Liste aller Elemente, die der Zeichenfläche hinzugefügt wurden (siehe *Das Elemente-Bedienfeld* für weitere Informationen).
- Das Bedienfeld *Rücknahmeprotokoll* zeigt die Historie aller am Layout vorgenommenen Änderungen an. Mit einem Mausklick ist es möglich, Layoutschritte rückgängig zu machen und bis zu einem bestimmten Status wiederherzustellen.
- Im Reiter *Layout* können Sie allgemeine Parameter einstellen, die auf das Layout angewendet werden, wenn Sie es exportieren oder darin arbeiten (siehe *Das Layout-Bedienfeld* für eine genauere Beschreibung);
- Der Reiter *Elementeigenschaften* enthält die Eigenschaften für das ausgewählte Element. Klicken Sie auf die Schaltfläche Element auswählen/verschieben, um ein Element in der Zeichenfläche auswählen zu können (z.B. eine Legende, einen Maßstab oder eine Beschriftung). Klicken Sie dann den Reiter *Elementeigenschaften* an und passen Sie die Einstellungen für das ausgewählte Element an (siehe *Layout-Elemente* für eine genauere Beschreibung über die Einstellungen jedes Elements).
- Der Reiter *Atlas* ermöglicht es Ihnen, einen Atlas für das aktuelle Drucklayout zu erzeugen und gibt Ihnen Zugang zu seinen Einstellungen (siehe *Einen Atlas erzeugen* für eine genauere Beschreibung der Erzeugung eines Atlas).

Im unteren Teil des Drucklayoutfensters können Sie eine Statusleiste mit der Mausposition, der aktuellen Seitenanzahl, ein Auswahlfeld zum Einstellen der Zoomstufe (plus Schieberegler), die Anzahl der gewählten Objekte (falls eine Auswahl vorhanden ist) und im Falle einer Atlaserzeugung, die Anzahl der Atlasobjekte, finden.

Im oberen Teil des Drucklayoutfensters finden Sie Menüs und Werkzeugleisten. Alle Werkzeuge des Drucklayouts sind über Menüs und Schaltflächen in der Werkzeugleiste verfügbar.

Die Werkzeugleisten und Bedienfelder können mit einem rechten Mausklick auf eine beliebige Symbolleiste oder über *Ansicht* ► *Werkzeugleisten* ► oder *Ansicht* ► *Bedienfelder* ► aus- und eingeschaltet werden.

Menüs und Werkzeuge

Das Layout-Menü

Mit den Einträgen im Menü *Layout* können Sie das Layout verwalten:

- Speichern Sie die Projektdatei direkt aus dem Drucklayout heraus.
- Erzeugen Sie mit dem Eintrag *Neues Layout…* ein neues leeres Drucklayout.
- *Layout duplizieren…* : Erzeugen Sie ein neues Drucklayout als Duplikat des aktuellen.
- Löschen Sie mit dem Eintrag *Layout löschen…* das aktuelle Layout.
- Öffnen Sie die Layout-Verwaltung mit *Layout-Verwaltung…*.
- Öffnen Sie ein vorhandenes Drucklayout mit *Layouts* ►.

Wenn Sie das Layout fertig gestaltet haben, können Sie den aktuellen Zustand mit *Als Vorlage speichern* in eine `.qpt` Vorlagendatei speichern und dann die Elemente aus dieser gespeicherten Vorlage mit *Elemente zur Vorlage hinzufügen…* in einem anderen Drucklayout wieder hineinzufügen.

Im Menü *Layout* gibt es auch nützliche Werkzeuge, um Layouts zu exportieren, sodass sie in Berichte eingebunden oder publiziert werden können. Diese Werkzeuge sind *Als Bild exportieren…*, *Als PDF exportieren…*, *Als SVG exportieren…* und *Drucken…*.

In der nachstehenden Tabelle sind aller verfügbaren Werkzeuge im Layout-Menü aufgeführt.

Tab. 18.1: „Layout“ Menu Tools

Werkzeug	Tastenkürzel	Werkzeugleiste	Referenz
Projekt speichern	`Strg+S`	*Layout*	*Vorstellung von QGIS-Projekten*
Neues Layout	`Ctrl+N`	*Layout*	*Layoutverwaltung*
Layout duplizieren		*Layout*	*Layoutverwaltung*
Layout löschen			
Layout-Verwaltung…		*Layout*	*Layoutverwaltung*
Layouts ►			
Layouteigenschaften…			*Das Layout-Bedienfeld*
Layout umbenennen…			
Seiteneigenschaften…			*Arbeiten mit den Seiteneigenschaften*
Seiten hinzufügen…		*Layout*	*Arbeiten mit den Seiteneigenschaften*
Elemente zur Vorlage hinzufügen		*Layout*	*Erstellen eines Layout-Elements*
Als Vorlage speichern…		*Layout*	*Layoutverwaltung*

Fortsetzung auf der nächsten Seite

Tab. 18.1 – Fortsetzung der vorherigen Seite

Werkzeug	Tastenkürzel	Werkzeugleiste	Referenz
Als Bild exportieren…		*Layout*	*Exportieren als Rasterbild*
Als SVG exportieren…		*Layout*	*Als SVG exportieren*
Als PDF exportieren…		*Layout*	*Als PDF exportieren*
Druckseiteneinstellungen…	`Strg+Shift+P`		
Drucken…	`Strg+P`	*Layout*	*Eine Ausgabe erzeugen*
Schließen	`Strg+Q`		

Bearbeiten-Menü

Das Menü *Bearbeiten* bietet Ihnen Werkzeuge, um Drucklayouts zu bearbeiten. Das beinhaltet gängige Aufgaben wie Auswahlwerkzeuge, Kopieren/Ausschneiden/Einfügen und die Funktionen Rückgängig/Wiederherstellen (siehe *Das Rücknahmeprotokoll-Bedienfeld: Aktionen rückgängig machen und wiederherstellen*) für Elemente im Layout.

Wenn Sie die Aktion Einfügen benutzen, werden die Elemente an der aktuellen Position des Mauszeigers eingefügt. Wenn Sie den Menüpunkt *Bearbeiten* ► *Durch Einfügen ersetzen* oder das Tastenkürzel `Strg+Umschalt+V` drücken, werden die Elemente auf die aktuelle Seite an der Position eingefügt, an der sie auch auf ihrer Originalseite platziert waren. Damit kann sichergestellt werden, dass kopierte Elemente von Seite zu Seite an genau derselben Stelle eingefügt werden.

In der nachstehenden Tabelle sind aller verfügbaren Werkzeuge im Layout-Menü aufgeführt.

Tab. 18.2: „Edit“ Menu Tools

Werkzeug	Tastenkürzel	Werkzeug	Referenz
(…) rückgängig machen (letzte Änderung)	`Strg+Z`	*Layout*	*Das Rücknahmeprotokoll-Bedienfeld: Aktionen rückgängig machen und wiederherstellen*
(…) wiederherstellen (der letzten rückgängig gemachten Änderung)	`Strg+Y`	*Layout*	*Das Rücknahmeprotokoll-Bedienfeld: Aktionen rückgängig machen und wiederherstellen*
Löschen	`Entf`		
Ausschneiden	`Strg+X`		
Kopieren	`Strg+C`		
Einfügen	`Strg+V`		
Durch Einfügen ersetzen	`Strg+Shift+V`		
Alles wählen	`Strg+A`		
Alles abwählen	`Strg+Umschalt`		
Auswahl umkehren			
Nächstes Elemente darunter wählen	`Strg+Alt+[`		
Nächstes Elemente darüber wählen	`Strg+Alt+]`		
Layout verschieben	`P`	*Werkzeugki*	
Zoom	`Z`	*Werkzeugki*	
Eintrag auswählen/verschieben	`V`	*Werkzeugki*	*Interaktion mit Layout-Elementen*
Inhalt verschieben	`C`	*Werkzeugki*	*Das Karten-Element*

Fortsetzung auf der nächsten Seite

Tab. 18.2 – Fortsetzung der vorherigen Seite

Werkzeug	Tastenkürzel	Werkzeug	Referenz
Knotenelement bearbeiten		*Werkzeugki*	*Das Knoten-Element*

Menü Ansicht

Das Menü *Ansicht* ermöglicht den Zugriff auf Navigationswerkzeuge und hilft bei der Konfiguration des allgemeinen Verhaltens des Drucklayouts. Neben den üblichen Zoom-Werkzeugen stehen Ihnen folgende Möglichkeiten zur Verfügung:

- Ansicht aktualisieren (wenn Sie die Ansicht in einem inkonsistenten Zustand vorfinden);
- Aktivieren Sie ein *Gitter*, an dem Sie Elemente einrasten lassen können, wenn Sie sie verschieben oder erstellen. Die Einstellung der Raster erfolgt in *Einstellungen* ► *Layoutoptionen…* oder im *Layout Bedienfeld*;
- aktivieren Sie *Führungen*, an denen Sie Elemente beim Verschieben oder Erstellen einrasten lassen können. Führungen sind rote Linien, die Sie erstellen können, indem Sie in das Lineal (oben oder auf der linken Seite des Layouts) klicken und an die gewünschte Stelle ziehen;
- *Schlaue Führungen*: verwenden andere Layoutelemente als Hilfslinien, an denen sie dynamisch einrasten, wenn Sie ein Element verschieben oder umgestalten;
- *Führungen löschen* um alle aktuellen Führungen zu entfernen;
- *Umgrenzungsrahmen anzeigen* um die Elemente herum, um Ihre Auswahl besser zu erkennen;
- *Lineale anzeigen* um das Layout herum;
- *Seiten anzeigen* oder Seiten auf transparent setzen. Oft wird Layout verwendet, um nicht-druckbare Layouts zu erstellen, z.B. für Präsentationen oder andere Dokumente, und es ist wünschenswert, die ZUsammenstellung mit einem völlig transparenten Hintergrund zu exportieren. In anderen Bearbeitungsprogrammen wird dies manchmal als „ unbegrenzte Kartenansicht“ bezeichnet.

Im Drucklayout können Sie die Zoomstufe mit dem Mausrad oder dem Schieberegler und dem Kombinationsfeld in der Statusleiste ändern. Wenn Sie während der Arbeit im Layoutbereich in den Schwenkmodus wechseln müssen, können Sie die `Leertaste` oder das Mausrad gedrückt halten. Mit `Strg+Leertaste` können Sie vorübergehend in den Vergrößerungsmodus und mit `Strg+Alt+Leertaste` in den Verkleinerungsmodus wechseln.

Bedienfelder und Werkzeugleisten können über das Menü *Ansicht* ► aktiviert werden. Um den verfügbaren Platz für die Interaktion mit einer Zusammenstellung zu maximieren, können Sie das Kontrollkästchen *Ansicht* ► *Bedienfeldsichtbarkeit umschalten* aktivieren oder `Strg+Tab` drücken; alle Bedienfelder werden ausgeblendet und nur zuvor sichtbare Bedienfelder werden wiederhergestellt, wenn Sie das Kontrollkästchen deaktivieren.

Es ist auch möglich, in den Vollbildmodus zu wechseln, um mehr Platz für Interaktionen zu haben, indem Sie `F11` drücken oder *Ansicht* ► *Vollbildmodus umschalten* verwenden.

Tab. 18.3: „View“ Menu Tools

Werkzeug	Tastenkürzel	Werkzeugleiste	Referenz
Aktualisieren	`F5`	*Navigation*	
Voransicht ►			
Hineinzoomen	`Strg++`	*Navigation*	
Hinaus zoomen	`Strg+-`	*Navigation*	
Zoomen auf 100%	`Strg+1`	*Navigation*	
Volle Ausdehnung	`Strg+0`	*Navigation*	
Auf Breite zoomen			

Fortsetzung auf der nächsten Seite

Tab. 18.3 – Fortsetzung der vorherigen Seite

Werkzeug	Tastenkürzel	Werkzeugleiste	Referenz
Gitter anzeigen	`Strg+'`		*Führungen und Gitter*
Am Gitter ausrichten	`Strg+Umschalt+'`		*Führungen und Gitter*
Führung anzeigen	`Strg+;`		*Führungen und Gitter*
An Führung ausrichten	`Strg+Umschalt+;`		*Führungen und Gitter*
Schlaue Führung	`Strg+Alt+;`		
Führungen verwalten…			*Das Führungen-Bedienfeld*
Führung löschen			*Das Führungen-Bedienfeld*
Lineale anzeigen	`Strg+R`		
Umgrenzungsrahmen anzeigen	`Strg+Umschalt+B`		
Seiten anzeigen			
Werkzeugleiste ►			*Bedienfelder und Werkzeugkästen*
Bedienfelder ►			*Bedienfelder und Werkzeugkästen*
Vollbildmodus umschalten	`F11`		*Sicht (View)*
Bedienfeldsichtbarkeit umschalten	`Strg+Tab`		*Sicht (View)*

Elemente-Menü

Mit dem Menü *Elemente* können Sie die Position der Elemente im Layout und die Beziehungen zwischen ihnen konfigurieren (siehe auch *Interaktion mit Layout-Elementen*).

Tab. 18.4: „Items“ Menu Tools

Werkzeug	Tastenkürzel	Werkzeugleiste	Referenz
Gruppieren	`Strg+G`	*Aktionswerkzeugleiste*	*Elemente gruppieren*
Auflösen	`Strg+Umschalt+G`	*Aktionswerkzeugleiste*	*Elemente gruppieren*
Hervorheben	`Strg+]`	*Aktionswerkzeugleiste*	*Ausrichtung und Verteilung*
Tiefer setzen	`Strg+[`	*Aktionswerkzeugleiste*	*Ausrichtung und Verteilung*
In den Vordergrund holen	`Strg+Umschalt+]`	*Aktionswerkzeugleiste*	*Ausrichtung und Verteilung*
In den Hintergrund schicken	`Strg+Umschalt+[`	*Aktionswerkzeugleiste*	*Ausrichtung und Verteilung*
Gewählte Elemente fixieren	`Strg+L`	*Aktionswerkzeugleiste*	*Fixieren von Elementen*
Alle Fixierungen aufheben	`Strg+Umschalt+L`	*Aktionswerkzeugleiste*	*Fixieren von Elementen*
Elemente ausrichten ►		*Aktionswerkzeugleiste*	*Ausrichtung und Verteilung*
Elemente verteilen ►		*Aktionswerkzeugleiste*	*Elemente verschieben und in der Größe ändern*
Größe ändern ►		*Aktionswerkzeugleiste*	*Elemente verschieben und in der Größe ändern*

Element-hinzufügen-Menü

In diesem Menü finden Sie Werkzeuge zum Erstellen von Kartenelementen. Jedes dieser Werkzeuge wird im Kapitel *Layout-Elemente* ausführlich beschrieben.

Tab. 18.5: „Add Item“ Menu Tools

Werkzeug	Werkzeugleiste	Referenz
Karte hinzufügen	*Werkzeugkiste*	*Das Karten-Element*
3D-Karte hinzufügen	*Werkzeugkiste*	*Das 3D Kartenelement*
Bild hinzufügen	*Werkzeugkiste*	*Das Bild-Element*
Beschriftung hinzufügen	*Werkzeugkiste*	*Das Beschriftungs-Element*
Dynamischen Text hinzufügen ►		*Das Beschriftungselement*
Legende hinzufügen	*Werkzeugkiste*	*Das Legenden-Element*
Maßstab hinzufügen	*Werkzeugkiste*	*Das Maßstabs-Element*
Nordpfeil hinzufügen	*Werkzeugkiste*	*Das Nordpfeil-Element*
Form hinzufügen ►	*Werkzeugkiste*	*Das Rechteck-Element*
► *Rechteck hinzufügen*	*Werkzeugkiste*	*Das Rechteck-Element*
► *Ellipse hinzufügen*	*Werkzeugkiste*	*Das Rechteck-Element*
► *Dreieick hinzufügen*	*Werkzeugkiste*	*Das Rechteck-Element*
Markierung hinzufügen	*Werkzeugkiste*	*Das Markierungs-Element*
Pfeil hinzufügen	*Werkzeugkiste*	*Das Pfeilelement*
Knotenelement hinzufügen ►	*Werkzeugkiste*	*Das Knoten-Element*
► *Polygon hinzufügen*	*Werkzeugkiste*	*Das Knoten-Element*
► *Polylinie hinzufügen*	*Werkzeugkiste*	*Das Knoten-Element*
HTML hinzufügen	*Werkzeugkiste*	*Das HTML-Rahmen Element*
Attributtabelle hinzufügen	*Werkzeugkiste*	*Das Attributtabellen-Element*
Feste Tabelle hinzufügen	*Werkzeugkiste*	*Das Feste Tabellenelement*
Geländehöhenprofil hinzufügen	*Werkzeugkiste*	*Das Geländehöhenprofil-Element*

Atlas-Menü

Tab. 18.6: „Atlas“ Menu Tools

Werkzeug	Tastenkürzel	Werkzeugleiste	Referenz
Atlas-Vorschau anzeigen	`Strg+ALt+/`	*Atlas*	*Vorschau und Erstellung eines Atlas*
Erstes Objekt	`Strg+<`	*Atlas*	*Vorschau und Erstellung eines Atlas*
Vorheriges Objekt	`Strg+,`	*Atlas*	*Vorschau und Erstellung eines Atlas*
Nächstes Objekt	`Strg+.`	*Atlas*	*Vorschau und Erstellung eines Atlas*
Letztes Objekt	`Strg+>`	*Atlas*	*Vorschau und Erstellung eines Atlas*

Fortsetzung auf der nächsten Seite

Tab. 18.6 – Fortsetzung der vorherigen Seite

Werkzeug	Tastenkürzel	Werkzeugleiste	Referenz
Atlas drucken…		*Atlas*	*Vorschau und Erstellung eines Atlas*
Atlas als Bilder exportieren…		*Atlas*	*Vorschau und Erstellung eines Atlas*
Atlas als SVG exportieren…		*Atlas*	*Vorschau und Erstellung eines Atlas*
Atlas als PDF exportieren…		*Atlas*	*Vorschau und Erstellung eines Atlas*
Atlas-Einstellungen		*Atlas*	*Einen Atlas erzeugen*

Einstellungen-Menü

Das Menü *Einstellungen* ► *Layoutoptionen…* ist eine Verbindung zum Menü *Einstellungen* ► *Optionen* ► *Layouts* im QGIS Hauptfenster. Hier können Sie einige Optionen einstellen, die als Standard für jedes neue Drucklayout verwendet werden:

- Mit *Layoutvorgaben* können Sie die zu verwendende Standardschriftart festlegen;
- Mit *Gitterdarstellung* können Sie den Gitterstil und die Farbe des Gitters festlegen. Es gibt drei Arten von Gittern: **Punkte**, **Ausgefüllt** und **Kreuze**;
- *Gitter- und Führungsvoreinstellungen* definiert Abstand, Versatz und Toleranz des Gitters (siehe *Führungen und Gitter* für weitere Details);
- *Layoutpfade*: zur Verwaltung der Liste der benutzerdefinierten Pfade zur Suche von Druckvorlagen.

Das Menü *Einstellungen* ► *Tastaturkürzel…* ermöglicht es Ihnen, die *Tastenkürzel-Verwaltung* vom Drucklayout-Fenster zu erreichen.

Kontext-Menüs

Je nachdem, wo Sie im Drucklayoutdialog mit der rechten Maustaste klicken, öffnen Sie ein Kontextmenü mit verschiedenen Optionen:

- Klicken Sie mit der rechten Maustaste auf die Menüleiste oder eine beliebige Symbolleiste, und Sie erhalten eine Liste der Bedienfelder und Werkzeugleisten, die Sie mit einem Klick aktivieren oder deaktivieren können.
- Wenn Sie mit der rechten Maustaste auf ein Lineal klicken, haben Sie verschiedene Optionen für die Führungen: *Führungen anzeigen*, *An Führungen ausrichten*, *Führungen verwalten…* oder *Führungen löschen* öffnen. Es ist auch möglich, die Lineale ein- und auszublenden.
- Klicken Sie mit der rechten Maustaste in die Kartenansicht des Drucklayouts und:
  - Sie können die letzten Änderungen *(…) rückgängig machen* und *Wiederherstellen* oder ein Element *Kopieren*, *Ausschneiden* (nur verfügbar, wenn ein Element ausgewählt ist) oder *Einfügen* (nur verfügbar, wenn kein Element ausgewählt ist).
  - Wenn Sie auf eine Seite klicken, können Sie zusätzlich die entsprechenden *Seiteneigenschaften* aufrufen oder die *Seite löschen*.
  - Wenn Sie auf ein ausgewähltes Element klicken, können Sie auch die *Elementeigenschaften* öffnen.
  - Wenn mehrere Elemente ausgewählt sind, können Sie sie entweder gruppieren und/oder die Gruppierung aufheben, wenn mindestens eine Gruppe bereits in der Auswahl enthalten ist.
- Wenn Sie mit der rechten Maustaste in ein Textfeld oder ein Spinbox-Widget eines beliebigen Bedienfelds klicken, können Sie dessen Inhalt bearbeiten.

Das Layout-Bedienfeld

Im Bedienfeld *Layout* können Sie die globalen Einstellungen für Ihr Drucklayout festlegen.

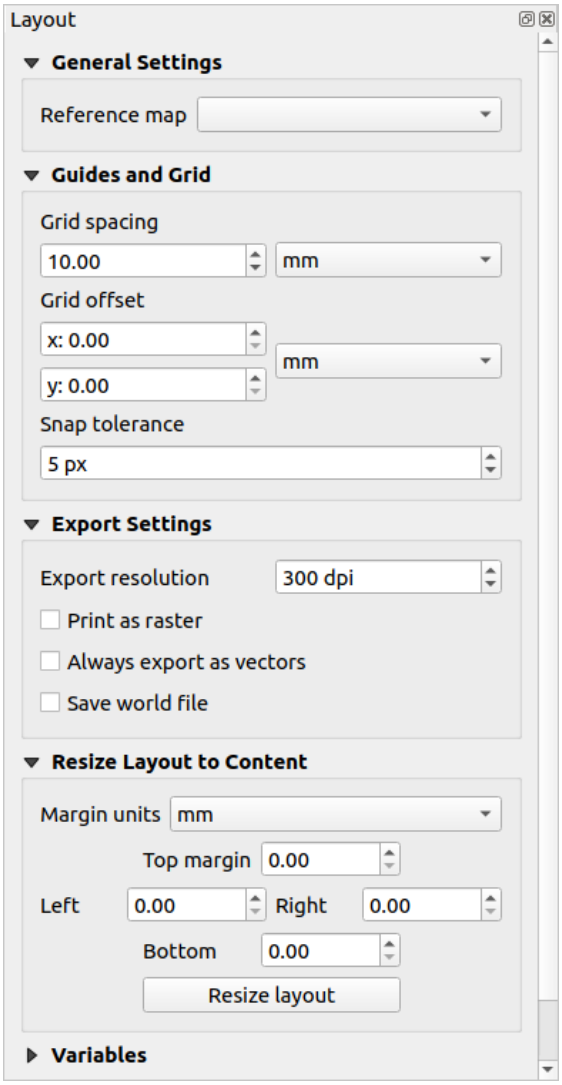


Abb. 18.3: Das Layout-Bedienfeld im Drucklayout

Allgemeine Einstellungen

In einem Drucklayout können Sie mehr als ein Karten-Element verwenden. Die *Referenzkarte* stellt das Karten-Element dar, das als Hauptkarte des Layouts verwendet wird. Es wird zugewiesen, wenn es ein Karten-Element im Layout gibt. Das Layout verwendet diese Karte in allen Eigenschaften und Variablen, die Einheiten oder den Maßstab berechnen. Dies gilt auch für den Export des Drucklayouts in georeferenzierte Formate.

Außerdem sind die Einstellungen für neue Layoutelemente wie Maßstabsleiste, Legende oder Nordpfeil (Ausrichtung, angezeigte Layer, Maßstab, …) standardmäßig an das Kartenelement gebunden, über das sie gezeichnet werden, und fallen auf die Referenzkarte zurück, wenn es keine überlappende Karte gibt.

Führungen und Gitter

Sie können auf Ihrem Blatt Papier einige Markierungen anbringen, die Ihnen helfen, einige Gegenstände genau zu platzieren. Diese Markierungen können sein:

- einfache horizontale oder vertikale Linien (**Führungen** genannt), die an der gewünschten Stelle platziert werden (siehe *Das Führungen-Bedienfeld* zur Erstellung von Führungen).
- oder ein regelmäßiges **Gitter**: ein Netz aus horizontalen und vertikalen Linien, das das Layout überlagert.

Einstellungen wie *Gitterabstand* oder *Gitterversatz* können in dieser Gruppe ebenso angepasst werden wie die *Einrasttoleranz*, die für Elemente verwendet wird. Die Toleranz ist der maximale Abstand, unter dem der Mauszeiger beim Verschieben, Ändern der Größe oder Erstellen eines Elements an einem Raster oder einer Hilfslinie gefangen wird.

Ob Gitter oder Führungen angezeigt werden sollen, wird im Menü *Ansicht* eingestellt. Dort können Sie auch festlegen, ob sie zum Fangen von Layoutelementen verwendet werden sollen. Wenn sich sowohl eine Gitter-Linie als auch eine Führung innerhalb der Toleranz eines Punktes befinden, haben die Führungen immer Vorrang - da sie manuell gesetzt wurden (daher die Annahme, dass sie explizit an sehr wünschenswerten Fangpositionen platziert wurden und vor dem allgemeinen Gitter ausgewählt werden sollten).

> **Bemerkung**
>
> Im Menü *Einstellungen* ► *Layoutoptionen* können Sie auch die oben genannten Raster- und Hilfslinienparameter einstellen. Diese Optionen gelten jedoch nur als Standardwerte für neue Drucklayouts.

Exporteinstellungen

Sie können in *Export-Auflösung* eine Auflösung festlegen, die für alle exportierten Maps verwendet werden soll. Diese Einstellung kann dann bei jedem Export einer Karte überschrieben werden.

Aufgrund einiger fortgeschrittener Rendering-Optionen (*Mischmodus*, *Effekte*…) kann es sein, dass ein Layoutelement gerastert werden muss, um korrekt exportiert zu werden. QGIS rastert es einzeln, ohne dass jedes andere Element ebenfalls gerastert werden muss. Dies erlaubt es, beim Drucken oder Speichern als PostScript oder PDF die Elemente so weit wie möglich als Vektoren zu belassen, z.B. wird ein Kartenelement mit Layer- Deckkraft nicht dazu führen, dass Beschriftungen, Maßstabsbalken usw. ebenfalls gerastert werden müssen. Sie können jedoch:

- erzwingen, dass alle Elemente gerastert werden, indem Sie das Kontrollkästchen *Als Raster drucken* aktivieren;
- oder verwenden Sie die gegenteilige Option, d.h. *Immer als Vektoren exportieren*, um den Export der Elemente als Vektoren zu zwingen. Beachten Sie, dass dies in einigen Fällen dazu führen kann, dass die Ausgabe anders aussieht als das Layout.

Wenn das Format es zulässt (z.B. `.TIF`, `.PDF`), führt der Export eines Drucklayouts standardmäßig zu einer georeferenzierten Datei (basierend auf dem Element *Referenzkarte* in der Gruppe *Allgemeine Einstellungen*). Bei anderen Formaten erfordert die georeferenzierte Ausgabe, dass Sie eine World-Datei erzeugen, indem Sie die Checkbox *World-Datei speichern* aktivieren. Die Weltdatei wird neben der/den exportierten Karte(n) erstellt, trägt den Namen der Seite, die mit der Referenzkarte ausgegeben wird, und enthält Informationen zur einfachen Georeferenzierung.

Layout auf Inhaltsgröße anpassen

Mit dem Werkzeug *Layout anpassen* in dieser Gruppe erstellen Sie eine eindeutige Seitenkomposition, deren Umfang den aktuellen Inhalt des Drucklayouts abdeckt (mit optionalem *Rand* um die beschnittenen Grenzen).

Beachten Sie, dass sich dieses Verhalten von der Option *Auf Inhalt zuschneiden* insofern unterscheidet, als dass alle Elemente auf einer echten und eindeutigen Seite platziert werden, die alle bestehenden Seiten ersetzt.

Variablen

Unter *Variablen* werden alle Variablen aufgelistet, die auf der Ebene des Layouts verfügbar sind (einschließlich aller globalen und Projektvariablen).

Es ermöglicht dem Benutzer auch die Verwaltung von Variablen auf Layoutebene. Klicken Sie auf die Schaltfläche , um eine neue benutzerdefinierte Variable auf Layoutebene hinzuzufügen. Ebenso können Sie eine benutzerdefinierte Variable auf Layoutebene aus der Liste auswählen und auf die Schaltfläche klicken, um sie zu entfernen.

Weitere Informationen zur Verwendung von Variablen finden Sie im Abschnitt *Allgemeine Werkzeuge*.

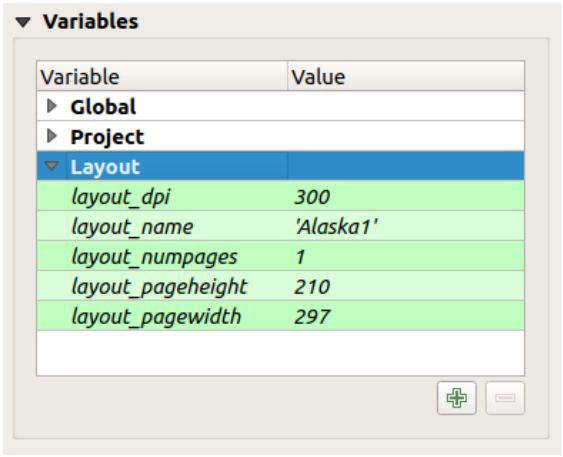


Abb. 18.4: Variablen-Editor im Drucklayout

Arbeiten mit den Seiteneigenschaften

Ein Layout kann aus mehreren Seiten zusammengesetzt sein. Eine erste Seite kann zum Beispiel eine Kartenansicht zeigen, eine zweite Seite die Attributtabelle eines Layers und eine dritte Seite einen HTML-Rahmen, der auf die Website Ihrer Organisation verweist. Oder Sie können auf jeder Seite mehrere Arten von Elementen hinzufügen.

Eine neue Seite hinzufügen

Darüber hinaus kann ein Layout mit unterschiedlichen Seitengrößen und/oder Ausrichtungen erstellt werden. Um eine Seite hinzuzufügen, wählen Sie das Werkzeug |neueSeite| *Seiten hinzufügen…* aus dem Menü *Layout* oder der *Layout-Werkzeugleiste*. Der Dialog *Seiten einfügen* öffnet sich und Sie werden aufgefordert, Angaben zu machen:

- die Anzahl der einzufügenden Seiten;
- die Position der Seite(n): vor oder nach einer bestimmten Seite oder am Ende des Drucklayouts;
- Die *Seitengröße*: Sie kann ein voreingestelltes Seitenformat (`A4`, `B0`, `Legal`, `Letter`, `ANSI A`, `Arch A` und deren Ableitungen sowie eine Auflösungsart, wie `1920x1080` oder `1024x768`) mit zugehöriger *Orientierung* (Hoch- oder Querformat) sein.

Die Seitengröße kann auch ein „benutzerdefiniertes“ Format haben; in diesem Fall müssen Sie *Breite* und *Höhe* (mit gesperrtem Größenverhältnis, falls erwünscht) eingeben und die zu verwendende Einheit (z.B. `mm`, `cm`, `px`, `pt`, `in`, ``ft`, …)` wählen. Die Umrechnung der eingegebenen Werte erfolgt automatisch beim Wechsel von einer Einheit zur anderen.

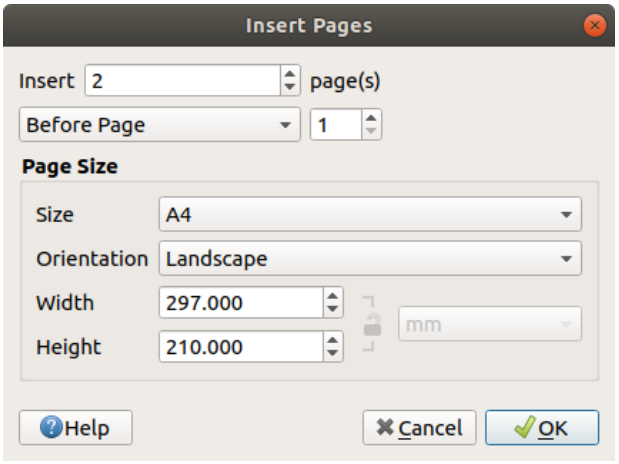


Abb. 18.5: Eine neue Seite im Drucklayout erstellen

Ändern der Seiteneigenschaften

Jede Seite kann später über das Fenster „Seite:guilabel:*Elementeigenschaften*“ angepasst werden. Um auf die Eigenschaften einer Seite zuzugreifen, gehen Sie wie folgt vor:

- Klicken Sie mit der linken Maustaste auf einen leeren Bereich der Seite.
- Klicken Sie mit der rechten Maustaste auf eine Seite und wählen Sie *Seiteneigenschaften…*.
- Oder gehen Sie über *Layout* ► *Seiteneigenschaften…* bzw *Layout* ► *Eigenschaften der Seite n…*

Im *Elementeeigenschaften* Bedienfeld finden Sie dann die folgenden Einstellungsmöglichkeiten:

- bei Layouts mit mehreren Seiten wird die Nummer der aktiven Seite und die Anzahl der verfügbaren Seiten angezeigt
- der oben beschriebene Rahmen *Seitengröße*. Sie können jede Eigenschaft mit den datenbezogenen Überschreibungsoptionen ändern (siehe auch *Die datendefinierte Übersteuerung in einem Atlas* für ein Beispiel);
- den *Hintergrund* der aktuellen Seite unter Verwendung der gewünschten *Farbe* oder *Symbols*.
- the *Apply to all Pages* button updates the current page properties to the other existing pages in the layout;
- mit *Seite aus Export ausschließen*, um zu steuern, ob die aktuelle Seite mit ihrem Inhalt in die *Layut Ausgabe* aufgenommen werden soll;

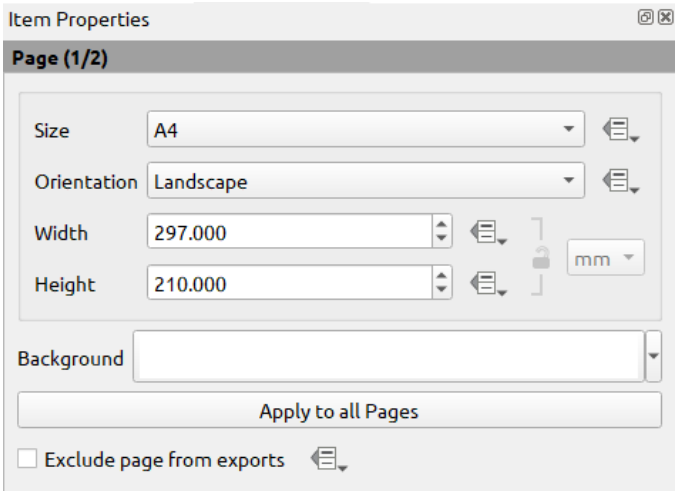


Abb. 18.6: Dialogfeld der Seiteneigenschaften

Das Führungen-Bedienfeld

Führungen sind vertikale oder horizontale Hilfslinien, die Sie auf einer Layoutseite einfügen können, um Sie bei der Platzierung von Elementen zu unterstützen, wenn Sie diese erstellen, verschieben oder ihre Größe ändern. Damit Führungen aktiv sind, müssen die Optionen *Ansicht* ► *Führung anzeigen* und *Ansicht* ► *An Führung ausrichten* aktiviert sein. Um eine Hilfslinie zu erstellen, gibt es zwei verschiedene Methoden:

- Wenn die Lineale sichtbar sind (*Ansicht* ► *Lineale anzeigen*), können Sie eine Führung erstellen, indem sie auf ein Lineal klicken, mit gedrückter Maustaste in den Seitenbereich ziehen und die Maustaste an der gewünschten Position loslassen.
- Für genauere Einstellungen aktivieren Sie das Bedienfeld *Führungen* über *Ansicht* ► *Führungen verwalten* oder durch Auswahl von *Führungen für Seite verwalten…* aus dem Kontextmenü der Seite.

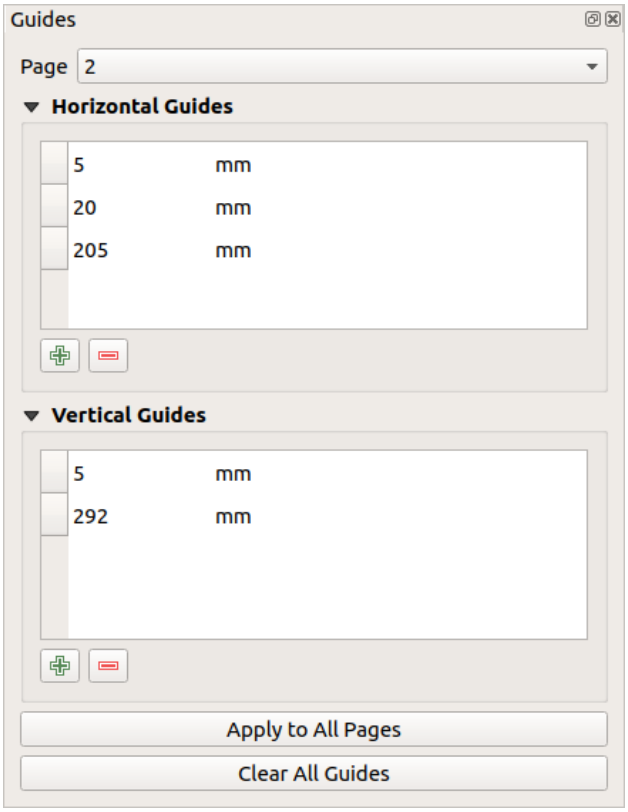


Abb. 18.7: Das Führungen-Bedienfeld

Das Bedienfeld *Führungen* ermöglicht die Erstellung von Führungen an exakter Stelle:

1. Wählen Sie die *Seite*, der Sie die Hilfslinien hinzufügen möchten
2. Klicken Sie auf die Schaltfläche Neue Führung hinzufügen und geben Sie die Koordinaten der horizontalen oder vertikalen Linie ein. Der Ursprung liegt in der linken oberen Ecke. Hierfür stehen verschiedene Einheiten zur Verfügung.

   Das Bedienfeld ermöglicht auch die Anpassung der Position vorhandener Hilfslinien an exakte Koordinaten: Klicken Sie auf den Wert und ersetzen Sie ihn.
3. Das Bedienfeld *Führungen* listet nur die Elemente für die aktuelle Seite auf. Es ermöglicht das Erstellen oder Entfernen von Hilfslinien nur für die aktuelle Seite. Sie können jedoch die Schaltfläche *Auf alle Seiten anwenden* verwenden, um die Hilfslinienkonfiguration der aktuellen Seite auf die anderen Seiten im Layout zu übertragen.
4. Um eine Führung zu löschen, wählen Sie sie aus und drücken Sie die Schaltfläche Ausgewählte Führung löschen. Verwenden Sie *Alle Führungen löschen*, um alle Führungen auf der aktuellen Seite zu entfernen.

Tipp

Fangen an vorhandenen Layoutelementen

Neben Hilfslinien und Rastern können Sie beim Verschieben, Ändern der Größe oder Erstellen neuer Elemente auch vorhandene Elemente als Fangreferenzen verwenden; diese werden **Schlaue Führungen** genannt und erfordern die Aktivierung der Option *Ansicht* ► *Schlaue Führungen*. Immer, wenn sich der Mauszeiger in der Nähe der Begrenzung eines Elements befindet, erscheint ein Einrastkreuz.

Das Elemente-Bedienfeld

Das Bedienfeld *Elemente* bietet einige Optionen zur Verwaltung der Auswahl und Sichtbarkeit von Elementen. Alle zur Kartenansicht hinzugefügten Elemente (einschließlich *Elementen-Gruppen*) werden in einer Liste angezeigt, und die Auswahl eines Elements bewirkt die Auswahl der entsprechenden Zeile in der Liste, ebenso wie die Auswahl einer Zeile die Auswahl des entsprechenden Elements in der Kartenansicht bewirkt. Dies ist also eine praktische Methode, um ein Element auszuwählen, das sich hinter einem anderen befindet. Beachten Sie, dass eine ausgewählte Zeile fett dargestellt wird. Die Auswahl mehrerer Einträge ist auch möglich, indem man `Umschalt` für zusammenhängende Einträge und `Strg` für nicht zusammenhängende Einträge drückt.

Für jedes verfügbare Element, können Sie:

- sichtbar machen oder nicht;
- |gesperrt| seine Position sperren oder entsperren;
- seine Z-Position ordnen. Sie können jedes Element in der Liste durch Klicken und Ziehen nach oben oder unten verschieben. Das oberste Element in der Liste wird in der Kartenansicht des Drucklayouts in den Vordergrund gebracht. Standardmäßig wird ein neu erstelltes Element in den Vordergrund gestellt.
- die Element-Namen durch Doppelklick auf den Text ändern;
- Klicken Sie mit der rechten Maustaste auf ein Element und wählen Sie, ob Sie es kopieren oder löschen oder seine *Elementeigenschaften* öffnen möchten.

Wenn Sie die richtige Position für ein Element gefunden haben, können Sie es sperren, indem Sie das Kästchen in der Spalte |gesperrt| anklicken. Gesperrte Elemente sind **nicht** in der Kartenansicht auswählbar. Gesperrte Elemente können entsperrt werden, indem Sie das Element im Bedienfeld *Elemente* auswählen und das Häkchen aus dem Kästchen entfernen oder die Symbole in der Symbolleiste verwenden.

Das Rücknahmeprotokoll-Bedienfeld: Aktionen rückgängig machen und wiederherstellen

IM Drucklayout ist es möglich, Änderungen rückgängig zu machen und wiederherzustellen. Dies kann mit dem Rückgängigmachen- und dem Wiederherstellen-Werkzeug geschehen, die im Menü *Bearbeiten*, in der Symbolleiste *Layout* oder im Kontextmenü verfügbar sind, wenn Sie mit der rechten Maustaste in den Drucklayoutbereich klicken:

- (…) rückgängig machen
- Wiederherstellen

Es ist aber auch möglich, Aktionen im Bedienfeld *Rücknahmeprotokoll* rückgängig zu machen (siehe Abb. 18.8). Das Bedienfeld listet die letzten Aktionen auf, die im Drucklayout durchgeführt wurden. Wenn Sie einen Eintrag mit der Maus markieren, wir auf der Karte der entsprechende Zustand angezeigt. Wenn Sie den Eintrag löschen wird die zugehörige Aktion, sowie **alle danach** folgenden, entfernt und somit rückgängig gemacht. **ACHTUNG**: die gelöschten Elemente können u.U. nicht wiederhergestellt werden. Es empfielt sich, das Projekt vorher zu speichern!

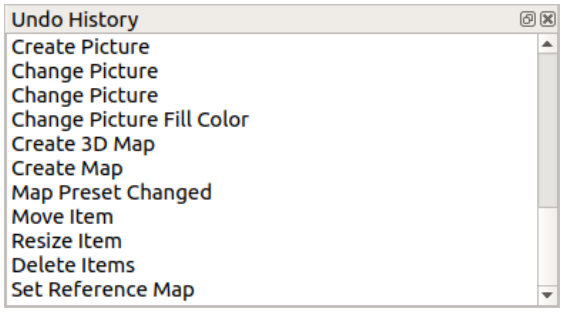


Abb. 18.8: Bedienfeld Rücknahmeprotokoll im Drucklayout

18.2 Layout-Elemente

18.2.1 Gemeinsame Optionen der Layout-Elemente

QGIS bietet eine große Anzahl von Elementen für das Layout einer Karte. Sie können aus Karte, Legende, Maßstabsbalken, Bild, Tabelle, Nordpfeil, Bildtyp usw. bestehen. Sie haben jedoch einige gemeinsame Optionen und Verhaltensweisen, die im Folgenden erläutert werden.

Erstellen eines Layout-Elements

Elemente können mit verschiedenen Werkzeugen erstellt werden, entweder ganz neu oder auf der Grundlage vorhandener Elemente.

So erstellen Sie ein Layout-Element ganz neu:

1. Wählen Sie das entsprechende Werkzeug entweder aus dem Menü *Element hinzufügen* oder der Leiste *Werkzeugkiste*.
2. Dann:
   - Klicken Sie auf die Seite und füllen Sie die Größen- und Platzierungsinformationen aus, die im Dialogfeld *Neue Elementeigenschaften* abgefragt werden (für Details siehe *Größe und Position*);

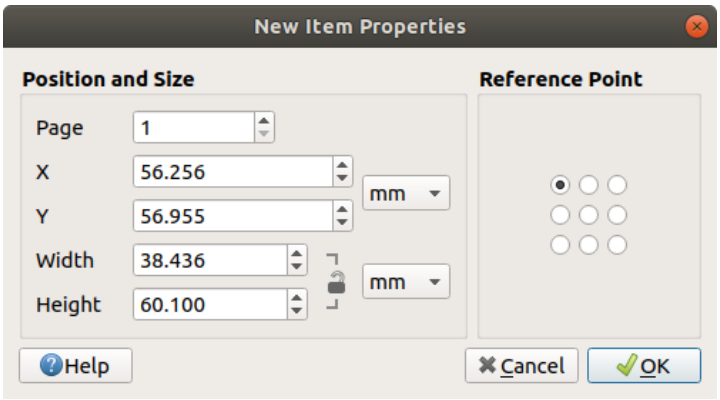


Abb. 18.9: Dialogfeld Neue Elementeigenschaften

   - Oder klicken und ziehen Sie, um die anfängliche Größe und Platzierung des Objekts zu bestimmen. Sie können *Gitter und Führung* verwenden, um eine genauere Position zu erhalten.

> **Bemerkung**
>
> Da sie spezielle Formen haben können, funktioniert das Zeichnen von Knoten- oder Pfeilelementen weder mit der Ein-Klick- noch mit der Klicken-und-Ziehen-Methode; Sie müssen jeden Knoten des Elements anklicken und platzieren. Siehe *Das Knoten-Element* für weitere Details.

Sie können auch:

1. Wählen Sie ein vorhandenes Element mit der Schaltfläche Element auswählen/verschieben in der *Werkzeugkiste*-Symbolleiste
2. Verwenden Sie das Kontextmenü oder die Werkzeuge des Menüs *Bearbeiten*, um das Element zu kopieren/auszuschneiden und es an der Mausposition als neues Element einzufügen.

Sie können auch den Befehl *Durch Einfügen ersetzen* (`Strg+Umschalt+V`) verwenden, um ein Element an denselben Koordinaten wie das Original zu platzieren.

Darüber hinaus können Sie mit dem Befehl *Layout* ► *Elemente zur Vorlage hinzufügen…* Elemente unter Verwendung einer Drucklayoutvorlage erstellen (für Details siehe *Layoutverwaltung*).

Tipp

Hinzufügen von Layoutelementen mit dem Dateibrowser

Ziehen Sie von Ihrem Dateibrowser oder über das Bedienfeld *Browser* eine Drucklayoutvorlage (`.qpt`-Datei) auf einen Drucklayoutdialog und QGIS fügt automatisch alle Elemente aus dieser Vorlage zum Layout hinzu.

Interaktion mit Layout-Elementen

Jedes Element innerhalb des Drucklayouts kann verschoben und in der Größe verändert werden, um ein perfektes Layout zu erstellen. Für beide Vorgänge müssen Sie zunächst das Werkzeug „Element auswählen/verschieben“ aktivieren und auf das Element klicken.

Sie können mehrere Elemente mit der Taste Element auswählen/verschieben auswählen: Klicken und ziehen Sie über die Elemente oder halten Sie die Taste `Umschalt` und klicken Sie auf jedes der gewünschten Elemente. Um die Auswahl eines Elements aufzuheben, klicken Sie es mit gedrückter `Umschalt`-Taste an.

Jedes Mal, wenn eine Auswahl getroffen wird, wird die Anzahl der ausgewählten Elemente in der Statusleiste angezeigt. Im Menü *Bearbeiten* finden Sie Aktionen zum Auswählen aller Elemente, zum Löschen der gesamten Auswahl, zum Umkehren der aktuellen Auswahl und mehr…

Elemente verschieben und in der Größe ändern

Sofern die Option *Ansicht* ► *Umgrenzungsrahmen anzeigen* nicht deaktiviert ist, zeigt ein ausgewähltes Element Quadrate an seinen Begrenzungen; das Bewegen eines dieser Quadrate mit der Maus verändert die Größe des Elements in die entsprechende Richtung. Wenn Sie während der Größenänderung `Shift` halten, bleibt das Seitenverhältnis erhalten. Wenn Sie `Alt` gedrückt halten, wird die Größe vom Mittelpunkt des Elements aus verändert.

Um ein Layout-Element zu verschieben, wählen Sie es mit der Maus aus und bewegen es, während Sie die linke Maustaste gedrückt halten. Wenn Sie die Bewegungen auf die horizontale oder vertikale Achse beschränken müssen, halten Sie die Taste `Umschalt` auf der Tastatur gedrückt, während Sie die Maus bewegen. Sie können ein ausgewähltes Element auch mit den `Pfeiltasten` auf der Tastatur bewegen; wenn die Bewegung zu langsam ist, können Sie sie durch Halten der `Umschalt`-Taste beschleunigen. Wenn Sie eine höhere Präzision benötigen, verwenden Sie die Eigenschaften *Position und Größe* oder das Fangen am Gitter oder den Führungen, wie oben für die Erstellung von Elementen beschrieben.

Das Ändern der Größe oder Verschieben mehrerer Elemente auf einmal erfolgt auf die gleiche Weise wie bei einem einzelnen Element. QGIS bietet jedoch einige fortgeschrittene Werkzeuge, um die Größe einer Auswahl von Elementen automatisch nach verschiedenen Regeln zu ändern:

- Die Höhe jedes Elements entspricht dem höchsten oder kürzesten ausgewählten Element;
- Die Breite jedes Elements entspricht dem breitesten oder dem schmalsten ausgewählten Element;
- Ändert die Größe von Einträgen in Quadrate : Jeder Eintrag wird so vergrößert, dass er ein Quadrat bildet.

Ebenso stehen *Werkzeuge* zur Verfügung, um mehrere Artikel durch äquidistantes Verteilen zu positionieren:

- Kanten (links, rechts, oben oder unten) von Elementen;
- horizontale oder vertikale Zentrierung von Elementen;
- Abstände zwischen den Elementen horizontal oder vertikal.

Elemente gruppieren

Die Gruppierung von Elementen ermöglicht es Ihnen, eine Gruppe von Elementen wie ein einzelnes zu bearbeiten: Sie können die Größe der Elemente als Ganzes leicht ändern, verschieben, löschen oder kopieren.

Um eine Gruppe von Elementen zu erstellen, markieren Sie mehr als ein Element und drücken Sie die Schaltfläche . *gruppieren* im Menü *Elemente* oder in der Werkzeugleiste *Aktionen* oder über das Kontextmenü. Eine Zeile mit dem Namen `Gruppe` wird dem Bedienfeld *Elemente* hinzugefügt und kann wie jedes andere Objekt des Elemente-Bedienfeldes ein- oder ausgeblendet werden <layout_items_panel>`. Gruppierte Elemente sind **nicht einzeln** auf der Kartenansicht auswählbar; verwenden Sie das Bedienfeld „Elemente“ für die direkte Auswahl und den Zugriff auf das Eigenschaftsfeld des Elements.

Fixieren von Elementen

Wenn Sie die richtige Position für ein Element gefunden haben, können Sie es mit der Schaltfläche *Gewählte Elemente fixieren* im Menü *Elemente* oder in der Werkzeugleiste *Aktionen* sperren oder das Kästchen neben dem Element im Bedienfeld *Elemente* anklicken. Gesperrte Elemente sind **nicht** auf der Kartenansicht auswählbar.

Fixierte Elemente können entsperrt werden, indem Sie das Element im Bedienfeld *Items* auswählen und das Häkchen entfernen oder die Symbole in der Symbolleiste verwenden.

Ausrichtung und Verteilung

Werkzeuge zum Anheben oder Absenken der Z-Position von Elementen im Layout befinden sich im Pulldown-Menü Hervorheben. Wählen Sie ein Element auf der Kartenansicht des Druck-Layouts und wählen Sie die entsprechende Funktion, um das ausgewählte Element über die anderen Elemente anzuheben oder abzusenken. Diese Reihenfolge wird im Bedienfeld *Elemente* angezeigt. Sie können Elemente im Bedienfeld *Items* auch anheben oder absenken, indem Sie die Beschriftung eines Objekts in dieser Liste anklicken und ziehen.

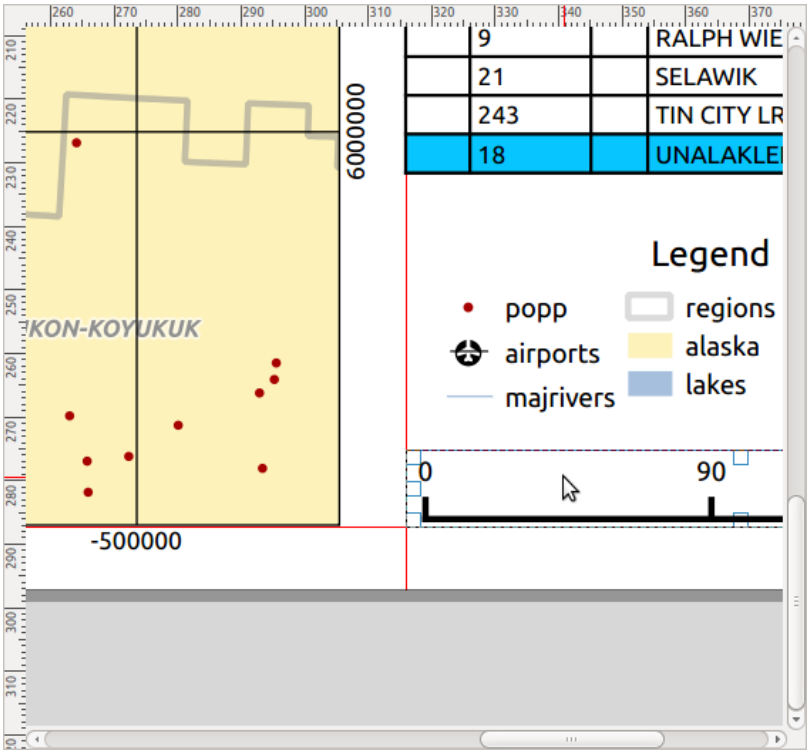


Abb. 18.10: Ausrichtungshilfslinien im Drucklayout

Es gibt mehrere Ausrichtungsoptionen im Pull-Down-Menü Ausgewählte Elemente ausrichten (siehe Abb. 18.10). Um eine Ausrichtungsfunktion zu verwenden, markieren Sie zunächst die Elemente und klicken dann auf eines der Ausrichtungssymbole:

- *Links ausrichten* oder *Rechts ausrichten*;
- *Oben ausrichten* oder *Unten ausrichten*;
- *Mittig ausrichten* horizontally or *Vertikal zentriert ausrichten*.

Alle ausgewählten Elemente werden dann an ihrem gemeinsamen Begrenzungsrahmen ausgerichtet. Beim Verschieben von Elementen auf der Kartenansicht erscheinen Ausrichtungshilfslinien, wenn Ränder, Mittelpunkte oder Ecken ausgerichtet werden.

Eine weitere Möglichkeit, die Platzierung von Layoutelementen zu verbessern, besteht darin, den Abstand zwischen ihnen auf der Layoutseite anzupassen. Wählen Sie dazu die Elemente aus und wählen Sie im Dropdown-Menü *Linken Rand verteilen*:

- *Linken Rand verteilen* oder *Rechte Kante verteilen* der Elemente äquidistant
- *Oberkanten verteilen* oder *Unterkanten verteilen* der Elemente äquidistant
- *Horizontale Zentren verteilen* oder *Vertikale Zentren verteilen* der Elemente äquidistant
- Gleichen Abstand zwischen den Elementen hinzufügen: *Horizontale Abstände gleichmäßig verteilen* oder *Vertikale Abstände gleichmäßig verteilen*

Gemeinsame Eigenschaften aller Elemente

Layout-Elemente haben eine Reihe gemeinsamer Eigenschaften, die Sie am unteren Rand des Bedienfelds *Elementeigenschaften* finden: Position und Größe, Drehung, Rahmen, Hintergrund, Elementkennung, Variablen und Darstellung (siehe Abb. 18.11).

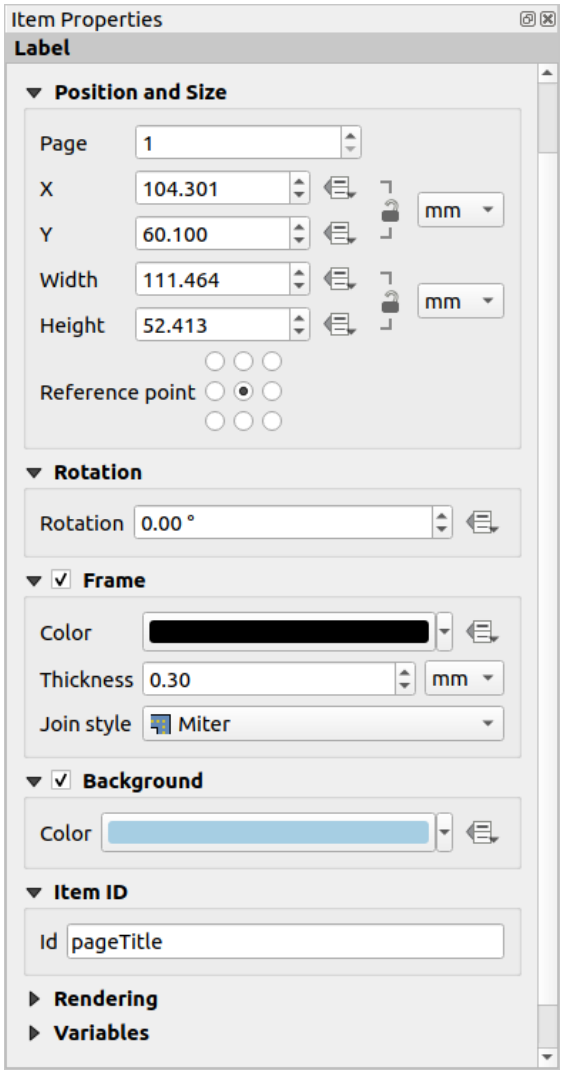


Abb. 18.11: Bereich der gemeinsamen Elementeigenschaften

Bemerkung

Das Symbol Datendefinierte Übersteuerung neben den meisten Optionen bedeutet, dass Sie diese Eigenschaft mit einem Layer, Attributen, einer Geometrie oder mit jeder anderen Eigenschaft eines Layout-Elements verknüpfen können, indem Sie *Ausdrücke* oder *Variablen* verwenden. Für weitere Informationen siehe *Einrichten von datendefinierter Übersteuerung*.

- Im Bereich *Position und Größe* können Sie die Größe und Position des Rahmens definieren, der das Element enthält (siehe *Größe und Position* für weitere Informationen).
- Die *Drehung* stellt die Drehung des Elements (in Grad) ein.
- Das Feld *Rahmen* zeigt oder verbirgt den Rahmen um das Element. Verwenden Sie die Widgets *Farbe*, *Dicke* und *Verbindungsstil*, um diese Eigenschaften anzupassen.
- In der Gruppe *Hintergrund* können Sie eine *Farbe* für den Hintergrundfarbe auswählen. Die Transparenz kann durch Ändern der Alpha-Feld-Einstellungen angepasst werden.
- Verwenden Sie die *Id* unter *Elementkennung*, um eine Beziehung zu anderen Drucklayoutelementen herzustellen. Dies wird mit QGIS Server und anderen möglichen Web-Clients verwendet. Sie können einem Element (z. B. einer Karte oder einer Beschriftung) eine Id zuweisen, und der Web-Client kann dann Daten

senden, um eine Eigenschaft (z. B. einen Beschriftungstext) für dieses spezifische Element festzulegen. Der Befehl GetProjectSettings listet die Elemente und IDs auf, die in einem Layout verfügbar sind.

- Mit *Darstellung* können Sie festlegen, ob und wie das Element angezeigt werden soll: Sie können z. B. *Mischmodi* anwenden, die Deckkraft des Elements anpassen oder *Element aus Exporten ausnehmen*.

Größe und Position

Die datendefinierte Steuerung der Größe und der Position ermöglicht es Ihnen, die Objekte genau zu platzieren.

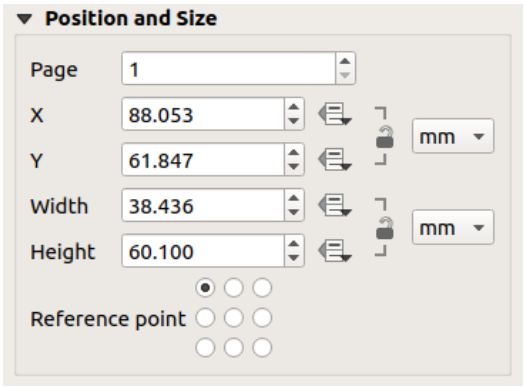


Abb. 18.12: Position und Größe

- die Nummer der Seite, auf der das Element platziert werden soll;
- den Referenzpunkt des Elements;
- die *X*- und *Y*-Koordinaten des *Referenzpunkts* des Elements auf der Seite. Das Verhältnis zwischen diesen Werten kann durch Klicken auf die Schaltfläche fixiert werden. Änderungen an einem Wert, die mit dem Widget oder dem Werkzeug Element auswählen/verschieben vorgenommen werden, werden in beiden wiedergegeben;
- die *Breite* und *Höhe* des Elementbegrenzungsrahmens. Wie bei den Koordinaten kann auch das Verhältnis zwischen Breite und Höhe festgelegt werden.

Darstellung

QGIS ermöglicht ein erweitertes Darstellen von Layout-Elementen, genau wie bei Vektor- und Raster-Layern.

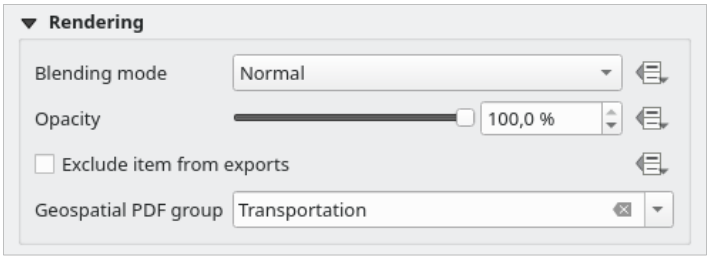


Abb. 18.13: Darstellung

- *Mischmodus*: Sie können spezielle Darstellungseffekte mit diesen Tools, die Sie vorher vielleicht nur von Grafikprogrammen kannten, erzielen. Die Pixel Ihrer oben liegenden und darunterliegenden Elemente werden durch den unten beschriebenen Modus gemischt (siehe *Mischmodi* für Beschreibungen jeden Effekts).
- *Deckkraft*: Mit diesem Werkzeug können Sie das darunter liegende Element im Layout sichtbar machen. Verwenden Sie den Schieberegler, um die Sichtbarkeit Ihres Elements an Ihre Bedürfnisse anzupassen. Sie können auch den Prozentsatz der Sichtbarkeit im Menü neben dem Schieberegler genau festlegen.

- *Element aus Exporten ausnehmen*: In Kombination mit datendefinierten Ausdrücken können Sie festlegen, dass ein Element bei bestimmten Exporten nicht angezeigt wird. Wenn Sie dieses Kontrollkästchen aktivieren, wird das Element beim Export in PDF, beim Drucken usw. nicht berücksichtigt.
- *Geospatial PDF group*: When set, a matching layer tree group will be created in the exported Geospatial PDF and the item will only be visible when this group is checked. This allows content to be selectively displayed as a group by viewers of the Geospatial PDF. E.g., it can allow extra layout content such as descriptive labels or legends to only be shown when layers from the group are visible, making the file export much more flexible.

Variablen

Unter *Variablen* werden alle Variablen aufgelistet, die auf der Ebene des Layoutelements verfügbar sind (einschließlich aller globalen, projekt- und layoutspezifischen Variablen). Layout-Kartenelemente haben einen zusätzlichen Abschnitt *Karteneinstellungen* für Variablen, die einfachen Zugriff auf Werte wie Maßstab, Ausdehnung usw. der Karte bieten.

In *Variablen* ist es auch möglich, Variablen auf der Ebene der Layoutelemente zu verwalten. Klicken Sie auf die Schaltfläche , um eine neue benutzerdefinierte Variable hinzuzufügen. Ebenso können Sie eine beliebige benutzerdefinierte Variable auf Elementebene aus der Liste auswählen und auf die Schaltfläche klicken, um sie zu entfernen.

Weitere Informationen zur Verwendung von Variablen finden Sie im Abschnitt *Speichern von Werten in Variablen*.

18.2.2 Das Karten-Element

Das Karten-Element ist der Hauptrahmen, der die von Ihnen hinzugefügten Daten in der Kartenansicht anzeigt. Verwenden Sie das Werkzeug *Karte hinzufügen* wie in der *Anleitung zum Erstellung von Elementen* beschrieben, um ein neues Karten-Element hinzuzufügen. Dies können Sie dann, wie in *Interaktion mit Layout-Elementen* dargestellt, bearbeiten.

Standardmäßig zeigt ein neues Karten-Element den aktuellen Status der Kartenansicht ` mit der entsprechenden Ausdehnung und den sichtbaren Layern. Sie können dies über das Bedienfeld :guilabel:`Elementeigenschaften des Karten-Elements :ref:` anpassen <label_mapview>`. Abgesehen von diesen *Elementeigenschaften* bietet dieses Element die folgenden Funktionalitäten:

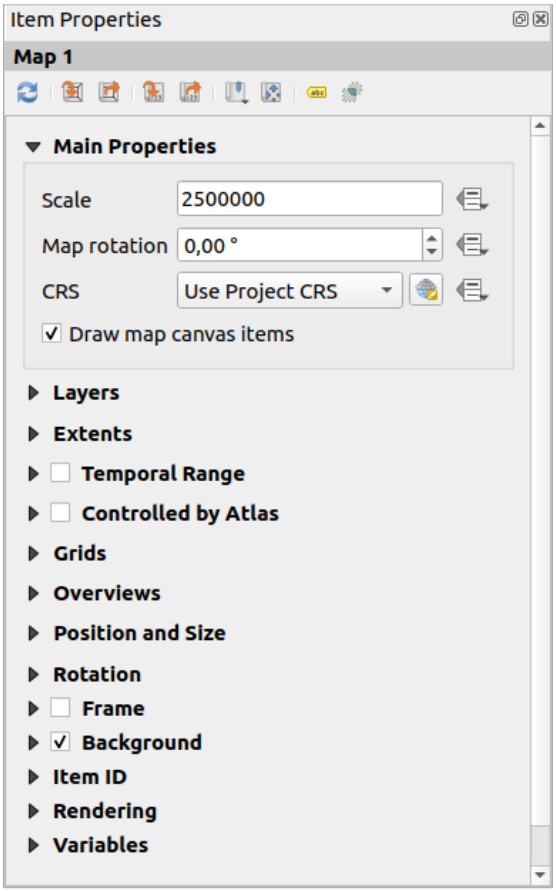


Abb. 18.14: Übersicht über das Elementeigenschaften-Bedienfeld eines Karten-Elements

Die Werkzeugleiste

Das Bedienfeld *Elementeigenschaften* eines Karten-Elements enthält eine Werkzeugleiste mit den folgenden Funktionen:

- Kartenvorschau aktualisieren
- Auf Kartenausdehnung der Hauptkarte setzen
- Aktuelle Kartenausdehnung in Hauptkarte anzeigen
- Kartenmaßstab auf den der Hauptkarte setzen
- Hauptkarte auf den aktuellen Kartenmaßstab setzen
- Lesezeichen: setzt die Ausdehnung des Kartenelements so, dass sie mit einem vorhandenen räumlichen Lesezeichen übereinstimmt.
- Kartenausdehnung interaktiv bearbeiten: Interaktives Schwenken und Zoomen innerhalb des Kartenelements
- Beschriftungseinstellungen: steuert das Verhalten von Objekt-Beschriftungen (Platzierung, Sichtbarkeit…) im Ausschnitt des Karten-Elements:
  - Setzen eines *Rand vom Kartenrand*, ein Abstand zu den Grenzen des Karten-Elements, innerhalb dessen keine Beschriftung angezeigt werden soll (auch möglich über datendefinierte Übersteuerung)
  - *Abgeschnittene Beschriftungen am Kartenrand erlauben*: steuert, ob Beschriftungen, die teilweise außerhalb des sichtbaren bzw. erlaubten Bereichs des Kartenelements liegen, gerendert werden sollen.

Wenn diese Option aktiviert ist, werden diese Beschriftungen angezeigt (wenn es keine Möglichkeit gibt, sie vollständig innerhalb des sichtbaren Bereichs zu platzieren). Wenn sie nicht aktiviert ist, werden teilweise sichtbare Beschriftungen übersprungen.

- *Beschriftungsblockierende Elemente*: ermöglicht es, andere Layoutelemente (wie Maßstabsleisten, Nordpfeile, eingefügte Karten usw.) als Blocker für die Kartenbeschriftungen im **aktiven** Kartenelement zu markieren. Dadurch wird verhindert, dass Kartenbeschriftungen unter diesen Elementen platziert werden - die Beschriftungslogik wird versuchen, eine alternative Platzierung für diese Beschriftungen zu versuchen, ist das nicht möglich, werden sie ganz verworfen.

  Wenn ein *Rand vom Kartenrand* gesetzt ist, werden die Beschriftungen nicht näher als der angegebene Abstand an die Layoutelementen heran platziert.

- *Nicht platzierte Beschriftungen anzeigen*: kann verwendet werden, um festzustellen, ob Beschriftungen auf der Layout-Karte fehlen (z.B. aufgrund von Konflikten mit anderen Kartenbeschriftungen oder aufgrund von unzureichendem Platz für die Platzierung der Beschriftung), indem sie in einer *vordefinierten Farbe* hervorgehoben werden.

- Zuschneidungseinstellungen: erlaubt es, das Kartenobjekt an das Atlas-Objekt und an Form- und Polygonelemente anzuhängen:

  - *Auf Atlas-Objekt zuschneiden*: Sie können festlegen, dass das Layout-Kartenelement automatisch auf das aktuelle *Atlas-Objekt* zugeschnitten wird.

    Es stehen verschiedene Zuschneidungsmodi zur Verfügung:

    * *Nur beim Zeichnen zuschneiden*: wendet einen zeichnungsbasierten Zuschnitt an, so dass Teile von Vektorobjekten, die außerhalb des Atlas-Features liegen, unsichtbar werden
    * *Objekte vor dem Zeichnen zuscheiden*: wendet den Clip vor dem Rendern von Objekten an, so dass die Ränder von Objekten, die teilweise außerhalb des Atlas-Features liegen, noch an der Grenze des Atlas-Features sichtbar sind
    * *Sich schneidende Objekte unverändert zeichnen*: rendert alle Objekte, die das aktuelle Atlas-Feature schneiden, aber ohne ihre Geometrie zu beschneiden.

    Sie können *Beschriftungen in Atlas-Objekt zwingen*. Wenn Sie die Option *Alle Layer zuschneiden* nicht auf das Atlas-Objekt anwenden wollen, können Sie die Option *Gewählte Layer zuschneiden* verwenden.

  - *Element zuschneiden*: es ist möglich, die Form des Kartenelements zu ändern, indem eine *Form* oder ein *Polygon* aus dem Druck-Layout verwendet wird. Wenn Sie diese Option aktivieren, wird die Karte automatisch an die in der Combobox ausgewählte Form angepasst. Auch hier stehen die oben genannten Zuschneidungsmodi zur Verfügung, und die Beschriftungen können so eingestellt werden, dass sie nur innerhalb der Beschneidungsform angezeigt werden.

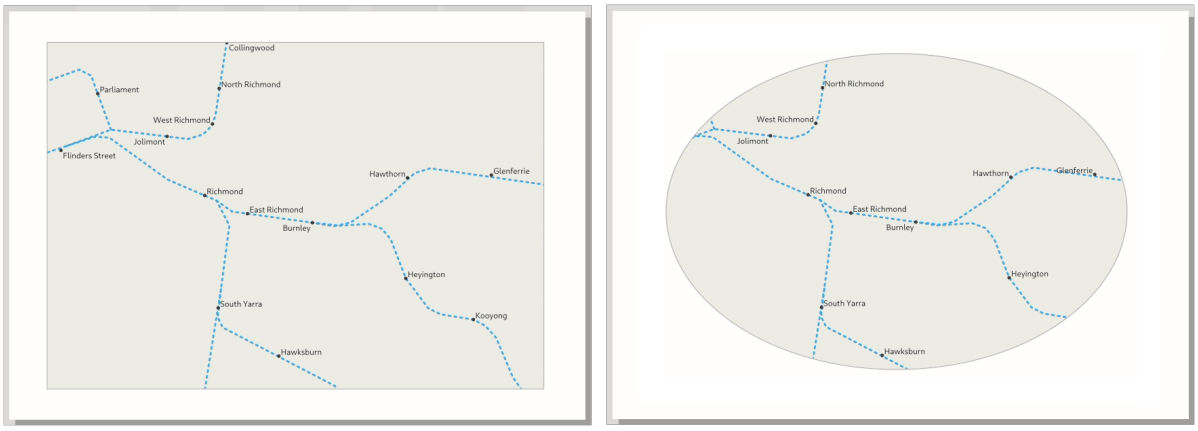


Abb. 18.15: Zuschneiden eines Karten-Elements auf eine Form

Haupteigenschaften

Im Bereich *Haupteigenschaften* (siehe Abb. 18.14) des Bedienfelds *Elementeigenschaften* sind folgende Optionen verfügbar:

- Die Schaltfläche *Kartenvorschau aktualisieren*, um die Darstellung des Kartenelements zu aktualisieren, wenn die Darstellung in der Kartenansicht geändert wurde. Beachten Sie, dass die Aktualisierung des Kartenelements in den meisten Fällen automatisch durch die Änderungen ausgelöst wird;
- Der *Maßstab* zur manuellen Einstellung des Maßstabs des Karten-Elements;
- Die *Kartendrehung* ermöglicht es Ihnen, den Inhalt des Kartenelements im Uhrzeigersinn zu drehen. Die Drehung der Kartenansicht kann hier nachgeahmt werden;
- Mit *KBS* können Sie den Inhalt des Kartenelements in einem beliebigen *KBS* anzeigen. Die Voreinstellung ist `Projekt-KBS benutzen`;
- Mit *Kartenelemente zeichnen* können Sie im Drucklayout *Anmerkungen* anzeigen, die auf der Kartenansicht platziert sind.

Layer

Standardmäßig wird die Darstellung von Karten-Elementen mit der Kartenansicht synchronisiert, d. h., wenn Sie die Sichtbarkeit von Layern umschalten oder ihren Stil im Bedienfeld *Layer* ändern, wird dies automatisch auf das Karten-Element angewendet. Da Sie, wie bei jedem anderen Element auch, mehrere Karten-Elemente zu einem Drucklayout hinzufügen können, kann es notwendig sein, diese Synchronisation zu unterbrechen, um verschiedene Ausschnitte, Layer-Kombinationen, unterschiedliche Maßstäbe o.ä. darzustellen. Im Bereich *Layer* (siehe Abb. 18.16) haben Sie dazu verschiedene Möglichkeiten.

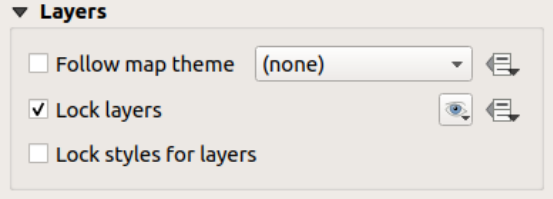


Abb. 18.16: Der Layer-Bereich

Wenn Sie das Karten-Element mit einem bestehenden *Kartenthema* konsistent halten wollen, markieren Sie die Checkbox *Kartenthema folgen* und wählen Sie das gewünschte Thema in der Dropdown-Liste aus. Jede Änderung des Themas im Hauptfenster von QGIS (mit der Funktion „Thema ersetzen“) wirkt sich automatisch auf den Karteneintrag aus. Wenn ein Kartenthema ausgewählt ist, ist die Option *Layerstile sperren* deaktiviert, da *Kartenthema folgen* auch den Stil (Symbolik, Beschriftungen, Diagramme) der Layer aktualisiert.

Um die in einem Karten-Element angezeigten Layer an die aktuelle Sichtbarkeit in der Kartenansicht zu fixieren, aktivieren Sie das Kontrollkästchen *Layer sperren*. Wenn diese Option aktiviert ist, wirken sich Änderungen an der Sichtbarkeit der Layer im QGIS-Hauptfenster nicht auf das Karten-Element des Layouts aus. Dennoch werden Stil und Beschriftung von gesperrten Layern entsprechend dem QGIS-Hauptfenster erneuert. Sie können dies verhindern, indem Sie *Layerstile sperren* verwenden.

Anstatt die aktuelle Kartenansicht zu verwenden, können Sie die Layer des Kartenelements auch auf die eines bestehenden Kartenthemas festlegen: Wählen Sie ein Kartenthema aus der Dropdown-Schaltfläche Layerliste aus Kartenthema übernehmen, wodurch auch das Kontrollkästchen *Layer sperren* aktiviert wird. Der Satz der sichtbaren Layer in diesem Kartenthema wird von nun an für das Kartenelement verwendet, bis Sie ein anderes Kartenthema auswählen oder die Option *Layer sperren* abwählen. Sie müssen dann eventuell die Ansicht aktualisieren, indem Sie die Schaltfläche Ansicht aktualisieren in der *Navigation*-Werkzeugleiste oder die Schaltfläche *Kartenvorschau aktualisieren* (siehe oben) verwenden.

Beachten Sie, dass im Gegensatz zur Option *Kartenthema folgen*, wenn die Option *Layer sperren* aktiviert und auf ein Kartenthema gesetzt ist, die Layer im Kartenelement nicht aktualisiert werden, selbst wenn das Kartenthema (mit der Funktion Thema ersetzen) im Hauptfenster von QGIS aktualisiert wird.

Gesperrte Layer im Karten-Element können auch über das Symbol neben der Option *datendefiniert* gesetzt werden. In diesem Fall wird die Auswahl in der Dropdown-Liste überschrieben. Sie müssen eine Liste von Layern, getrennt durch das Zeichen `|`, übergeben. Das folgende Beispiel sperrt das Karten-Element so, dass nur die Layer `layer 1` und `layer 2` verwendet werden:

```
concat ('layer 1', '|', 'layer 2')
```

Ausdehnung

Der Bereich *Ausdehnung* des Bedienfelds für die Kartenelemente bietet folgende Funktionen (siehe Abb. 18.17):

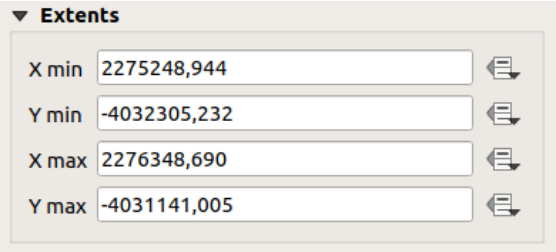


Abb. 18.17: Der Ausdehnung-Bereich

Der Bereich *Ausdehnung* zeigt die `X`- und `Y`-Koordinaten des im Karten-Element angezeigten Bereichs an. Jeder dieser Werte kann manuell ersetzt werden, um den angezeigten Bereich der Kartenansicht und/oder die Größe des Karten-Elements zu ändern. Die Ausdehnung kann auch mit den Werkzeugen oben im Bedienfeld des Karten-Elements geändert werden, wie z.B.:

- Auf Kartenausdehnung der Hauptkarte setzen
- Kartenmaßstab auf den der Hauptkarte setzen

Sie können die Ausdehnung eines Karten-Elements auch mit dem Werkzeug Elementinhalt verschieben ändern: Klicken und ziehen Sie innerhalb des Karten-Elements, um seine aktuelle Ansicht zu ändern, wobei derselbe Maßstab erhalten bleibt. Wenn das Werkzeug aktiviert ist, können Sie mit dem Mausrad hinein- oder herauszoomen und so den Maßstab der angezeigten Karte verändern. Kombinieren Sie die Bewegung mit gedrückter `Strg`-Taste, um einen kleineren Zoom zu erhalten.

Geländehöhenbereich

Mit dem Bereich *Geländehöhenbereich* in den Eigenschaften des Kartenelements können Sie den Inhalt bestimmter Layers anhand eines spezifischen Geländehöhenbereichs filtern. Während alle Layers sichtbar bleiben, werden die Daten derjenigen Layers gefiltert, die Geländehöhenfilterung unterstützen (derzeit Punktwolken und Raster-DEMs). Es werden nur die Teile dieser Layers angezeigt, die innerhalb des durch die *Untere* und *Obere* Grenzwerte festgelegten Geländehöhenbereiches liegen.

Der *Geländehöhenbereich* kann datendefiniert werden. Das bedeutet, dass Sie für verschiedene Objekte im *Atlas* oder *Bericht* unterschiedliche Geländehöhenbereiche festlegen können.

Zeitraum

Der Bereich *Zeitraum* des Bedienfelds *Elementeigenschaften ` für Karten-Elemente bietet Optionen zur Steuerung der Darstellung von Layern im Karten-Element auf der Grundlage eines zeitlichen Bereichs. Nur Layer, deren zeitliche Eigenschaften sich mit dem durch die Datum-/Zeit-Angaben im :guilabel:`Beginn* und *Ende*-Feld festgelegten Zeitbereich überschneiden, werden im Karten-Element angezeigt.

Die zugehörigen Felder für *datendefinierte Übersteuerung* helfen dabei, den Zeitbereich dynamisch zu gestalten, und ermöglichen die Ausgabe von zeitlichen Atlanten, d. h. automatisierten Karten mit fester räumlicher Ausdehnung, deren Inhalt sich zeitlich ändert. Wenn Sie z. B. als Coverage Layer eine csv-Datei mit einem Start- und einem Endpaar von Feldern und einer Reihe von Zeilen, die Datumsbereiche darstellen, verwenden, aktivieren Sie sowohl den zeitlichen Bereich als auch die Steuerung durch den Atlas in den Eigenschaften des Karten-Elements und wählen Sie den Atlassexport.

Gesteuert durch Atlas

Im Bereich *Gesteuert durch Atlas* ist nur verfügbar, wenn ein *Atlas* im Druck-Layout aktiv ist. Aktivieren Sie diese Option, wenn Sie möchten, dass das Karten-Element vom Atlas gesteuert wird; beim Durchlauf durch den Coverage-Layer wird die Ausdehnung des Karten-Elements auf das Atlas-Objekt wie folgt geschwenkt/gezoomt:

- *Rand um Objekte*: zoomt auf das Objekt im besten Maßstab und lässt um jedes Objekt einen Rand, der einen Prozentsatz der Breite oder Höhe des Kartenelements entspricht. Der Rand kann für alle Objekte gleich sein oder *datendefiniert übersteuert werden*, z.B. abhängig vom Kartenmaßstab;
- *Vordefinierter Maßstab (am passendsten)*: zoomt auf das Objekt in den *vordefinierten Maßstab*, in dem das Atlasobjekt am besten passt;
- *Fester Maßstab*: Atlasobjekte werden von einem zum anderen geschwenkt, wobei der Maßstab des Kartenobjekts beibehalten wird. Ideal, wenn Sie mit Objekten gleicher Größe arbeiten (z. B. einem Raster) oder Größenunterschiede zwischen Atlasobjekten hervorheben möchten.

Gitter

Mithilfe von Gittern können Sie Ihrer Karte Informationen in Bezug auf ihre Ausdehnung oder Koordinaten hinzufügen, entweder in der Projektion des Kartenelements oder in einer anderen. Der Bereich *Gitter* bietet die Möglichkeit, mehrere Gitter zu einem Kartenelement hinzuzufügen.

- Mit den Schaltflächen und können Sie ein ausgewähltes Gitter hinzufügen oder entfernen;
- Mit den Schaltflächen Pfeil nach oben und Pfeil nach unten können Sie ein Raster in der Liste nach oben oder unten verschieben, d. h. es über oder unter ein anderes Raster, über das Kartenelement verschieben.

Doppelklicken Sie auf das hinzugefügte Gitter, um es umzubenennen.

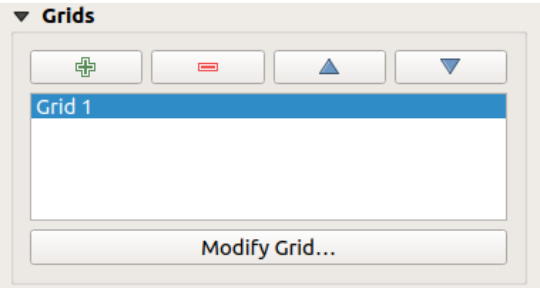


Abb. 18.18: Gitter-Bereich

Um ein Raster zu modifizieren, markieren Sie es und drücken Sie die Schaltfläche *Gitter ändern…*, um das Bedienfeld *Kartengittereigenschaften* zu öffnen und auf seine Konfigurationsoptionen zuzugreifen.

Aussehen des Gitters

Aktivieren Sie im Bedienfeld *Kartengittereigenschaften* das Kontrollkästchen *Aktiviere Gitter*, um das Gitter auf dem Kartenelement anzuzeigen.

Als Gittertyp können Sie auswählen:

- **Ausgefüllt**: zeigt eine Linie quer durch den Gitterrahmen. Der *Linienstil* kann in der *Farbe* und in der *Symboldarstellung* angepasst werden;
- **Kreuz**: zeigt ein Symbol am Schnittpunkt der Gitterlinien an, für das Sie den *Linienstil* und die *Kreuzbreite* einstellen können;
- **Markierungen**: Zeigt nur an den Schnittpunkten der Gitterlinien ein anpassbares Markierungssymbol an;
- oder **Nur Rahmen und Beschriftungen**.

Neben der Art des Gitters können Sie auch den Typ festlegen:

- das *KBS* des Rasters: standardmäßig folgt es dem KBS des Kartenelements. Drücken Sie die Taste KBS wählen, um es auf ein anderes KBS einzustellen.
- den *Intervall*, der für die Gitterreferenzen verwendet werden soll:
  - **Karteneinheiten**: Sie legen einen Abstand in der Karte (in der Einheit des KBS-Gitters) zwischen aufeinanderfolgenden Gitterpunkten in den Richtungen *X* und *Y* fest. Die Anzahl der Rasterpunkte hängt vom Maßstab der Karte ab.
  - Wenn Sie **Segmentbreite anpassen** wählen, wird das Rasterintervall auf der Grundlage der Kartenausdehnung dynamisch auf ein „schönes“ Intervall festgelegt. Dieses optimale Intervall wird innerhalb einer Reihe von Abständen berechnet, deren *Minimum* und *Maximum* Werte angepasst werden können.
  - Mit **Millimeter** oder **Zentimeter** legen Sie auf dem Papier einen Abstand zwischen aufeinanderfolgenden Rasterpunkten in den Richtungen *X* und *Y* fest. Die Anzahl der Rasterpunkte bleibt unabhängig vom Maßstab der Karte gleich.
- den *Versatz* von den Kanten des Kartenelements in *X* und/oder *Y*-Richtung
- und dem *Mischmodus* des Gitters (siehe *Mischmodi*), falls kompatibel.

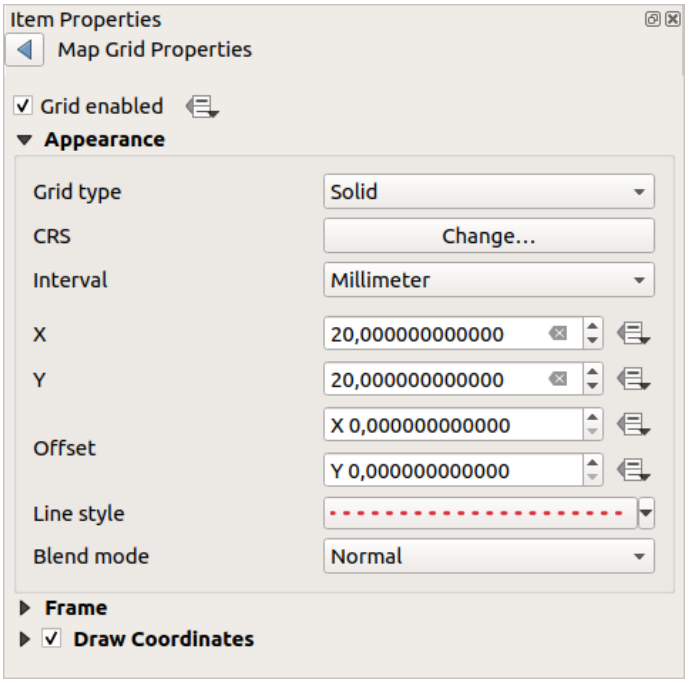


Abb. 18.19: Kartengittereigenschaften-Dialog

Rahmen des Gitters

Es gibt verschiedene Optionen für die Gestaltung des Rahmens, der die Karte enthält. Die folgenden Optionen sind verfügbar: `Kein Rahmen`, `Zebra`, `Zebra (nautisch)`, `Innere Markierungen`, `Äußere Markierungen`, `Innere und äußere Markierungen`, `Randlinie` und `Randlinie (nautisch)`.

Wenn kompatibel, ist es möglich, die *Rahmengröße*, einen *Rahmenrand*, die *Rahmenlinienstärke* mit zugehöriger Farbe und die *Rahmenfüllfarben* zu setzen.

Unter den Einstellungen für die :guilabel;`(…) Abgrenzung` können sie mit den Werten `Nur Breitengrad/Y` und `Nur Längengrad/X` verhindern, dass eine Mischung aus Breitengrad/Y und Längengrad/X Koordinaten an den Rändern angezeigt wird, wenn Sie mit gedrehten Karten oder reprojizierten Gittern arbeiten. Sie können auch festlegen, ob jede Seite des Gitterrahmens sichtbar sein soll oder nicht.

Wenn das Kartenelement gedreht wird (über die *Haupteigenschaften*) oder ein anderes KBS auf das Raster angewendet wird, sind die Rasterlinien möglicherweise nicht orthogonal zu den Seiten des Kartenelements. Dies kann dazu führen, dass das Raster nicht gut aussieht, wenn es mit inneren und/oder äußeren Häkchen dargestellt wird. Wenn Sie das Kontrollkästchen *Gitterdrehung folgen* aktivieren, werden die Markierungen an den Rasterlinien ausgerichtet. Außerdem können Sie einige weitere Eigenschaften einstellen:

- *Einteilungsausrichtung*: Die inneren und/oder äußeren Häkchen werden parallel zu den entsprechenden Gitterlinien ausgerichtet. Ihre Ausrichtung kann sein:
  - **Orthogonal**: Markierungen auf derselben Seite enden in einer Linie, parallel zur Seite. Dies kann z. B. dazu führen, dass einige Markierungen länger werden, wenn sie einen geringen Winkel zum Rahmen haben.
  - **Bestimmte Länge**: alle Häkchen haben die gleiche Länge, so dass sie möglicherweise nicht ausgerichtet sind
- *Winkelschwelle*: verhindert die Anzeige von Häkchen für Gitterlinien, die den Rahmenrand unterhalb eines bestimmten Schwellenwerts kreuzen
- *Rand von Kartenecke*: verhindert die Anzeige von Markierungen, die zu nahe an den Kartenecken liegen, da sie sich überschneiden und/oder außerhalb der Grenzen liegen könnten.

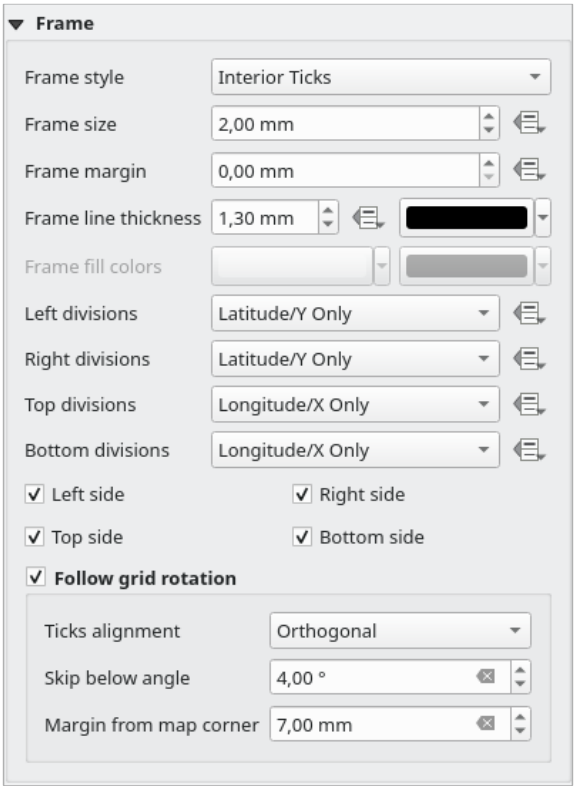


Abb. 18.20: Gitterrahmendialog

Koordinaten zeichnen

Das Kontrollkästchen *Koordinaten zeichnen* ermöglicht es Ihnen, dem Kartenrahmen Koordinaten hinzuzufügen. Die angezeigten Werte beziehen sich auf die gewählte Einheit unter *Intervall*. Die Optionen reichen von Dezimalzahlen bis zu Grad, Minuten und Sekunden, mit oder ohne Suffix, ausgerichtet oder nicht und einem benutzerdefinierten Format, das über den Ausdrucksdialog ausgewählt werden kann.

Für jede der Seiten *Links*, *Rechts*, *Oben* und *Unten* des Gitterrahmens können Sie angeben:

- wie die Koordinaten wiedergegeben werden sollen: **Zeige alles**, **Nur Breitengrad/Y anzeigen**, **Nur Längengrad/X anzeigen**, **Deaktiviert**. Die Anzeige von nur Breitengrad/Y- oder Längengrad/X-Werten in den Unterteilungen hilft zu verhindern, dass bei der Arbeit mit gedrehten Karten oder neu projizierten Gittern eine Mischung aus Breitengrad/Y- und Längengrad/X-Koordinaten auf jeder Seite angezeigt wird.
- die relative Position des Textes zum Gitterrahmen: **Außerhalb des Rahmens** oder **Im Rahmen**
- die Platzierung und Ausrichtung der Anmerkung:
  - **Horizontal**
  - **Vertikal aufsteigend**, **Vertikal absteigend**
  - **Grenzrichtung**
  - **Über dem Tick**, **Auf dem Tick**, **Unter dem Tick**, wenn ein Tick-basierter Rahmen verwendet wird

Sie können auch die *Schriftart Eigenschaften* (Schriftart, Größe, Farbe, Puffer,…), die *Abstand zum Kartenrahmen* und die *Koordinatengenauigkeit* (Anzahl der Nachkommastellen) für die gezeichneten Anmerkungen definieren.

Gitterdrehung folgen: verfügbar, wenn das Karten-Elemen gedreht oder das Gitter neu projiziert wird, hilft es Ihnen, die Platzierung der Anmerkungen anzupassen. Je nach ausgewähltem Platzierungsmodus werden die Beschriftungen auch gedreht:

- *Beschriftungsausrichtung*: sie kann **Orthogonal** oder von **Fester Länge** sein

- *Winkelschwelle*: verhindert die Anzeige von Anmerkungen für Gitterlinien, die den Rahmenrand unterhalb eines bestimmten Schwellenwerts kreuzen
- *Rand von Kartenecke*: verhindert, dass Anmerkungen zu nahe an den Kartenecken angezeigt werden, da sie sich überschneiden und/oder außerhalb der Grenzen liegen könnten.

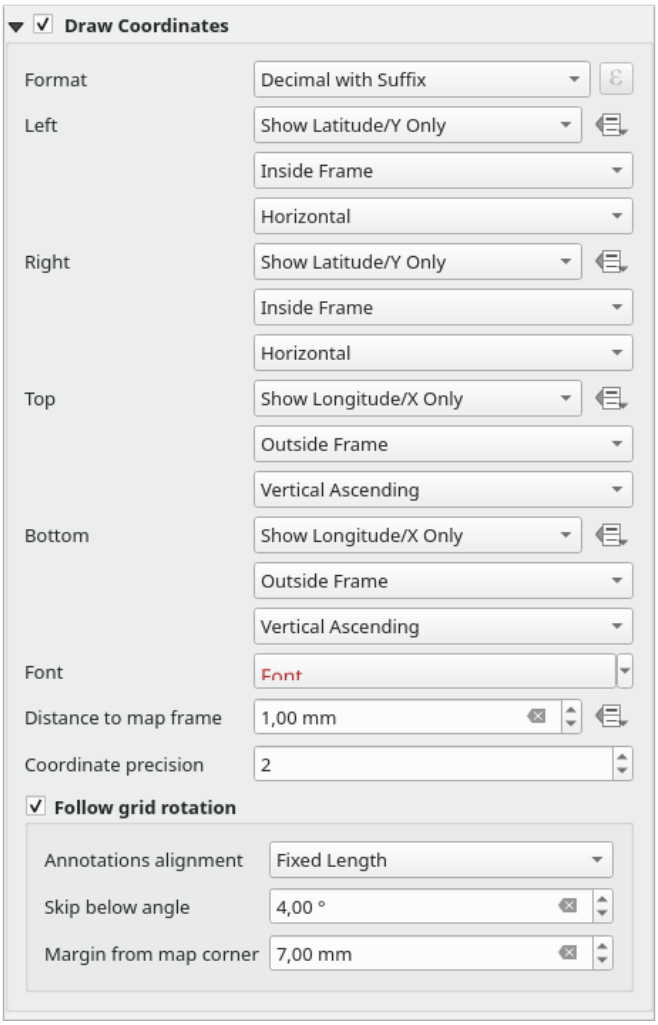


Abb. 18.21: Gitter-Koordinaten-Zeichnen-Dialog

Übersichten

Es kann vorkommen, dass Sie mehr als eine Karte im Drucklayout haben und das Untersuchungsgebiet eines Kartenelements auf einer anderen Karte lokalisieren möchten. Dies könnte zum Beispiel dazu dienen, den Kartenlesern zu helfen, das Gebiet in seinem größeren geografischen Kontext zu identifizieren, der in der zweiten Karte dargestellt ist.

Die Gruppe *Übersichten* des Bedienfeldes hilft Ihnen, die Verbindung zwischen zwei verschiedenen Kartenausschnitten herzustellen und bietet die folgenden Funktionen:

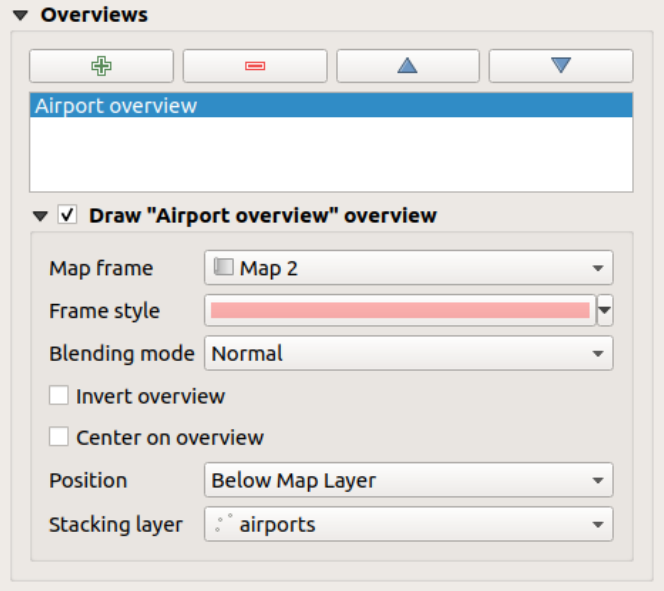


Abb. 18.22: Gruppe Übersichten

Um eine Übersicht zu erstellen, wählen Sie das Kartenelement aus, auf dem Sie die Ausdehnung des anderen Kartenelements anzeigen möchten, und erweitern Sie die Option *Übersicht* im Bedienfeld *Elementeigenschaften*. Drücken Sie dann die Schaltfläche , um eine Übersicht hinzuzufügen.

Bei der Erstellung wird die Übersicht zunächst „Übersicht 1“ (siehe Abb. 18.22) genannt. Sie können:

- sie mit einem Doppelklick umbenennen
- Mit den Schaltflächen und können Sie Übersichten hinzufügen oder entfernen
- Mit den Schaltflächen und können Sie eine Übersicht in der Liste nach oben oder unten verschieben und sie so über oder unter anderen Übersichten im Kartenelement platzieren (wenn sie sich an der gleichen *Stapelposition* befinden).

Wählen Sie dann die Übersicht in der Liste aus und aktivieren Sie das Kontrollkästchen *Übersicht „<Name der Übersicht> “zeichnen*, um das Zeichnen der Übersicht auf dem ausgewählten Kartenrahmen zu aktivieren. Sie können es mit anpassen:

- Der *Kartenrahmen* wählt das Kartenelement aus, dessen Ausmaße auf dem aktuellen Kartenelement angezeigt werden sollen.
- Der *Rahmenstil* verwendet die *Symboleigenschaften* zur Darstellung des Übersichtsrahmens.
- Der *Mischmodus* ermöglicht verschiedene Transparenzmischmodi einzustellen.
- Das Kontrollkästchen *Übersicht invertieren* erzeugt eine Maske um die Ausdehnungen, wenn es aktiviert ist: die referenzierten Kartenausdehnungen werden klar dargestellt, während der Rest des Kartenelements mit der Rahmenfüllfarbe überblendet wird (wenn eine Füllfarbe verwendet wird).
- Das Kontrollkästchen *Auf Übersicht zentrieren* verschiebt den Inhalt des Kartenelements so, dass der Übersichtsrahmen in der Mitte der Karte angezeigt wird. Sie können nur ein Übersichtselement zum Zentrieren verwenden, wenn Sie mehrere Übersichten haben.
- Mit der *Position* wird genau festgelegt, wo im Layer-Stapel des Kartenelements die Übersicht platziert wird. So kann z.B. eine Übersichtsausdehnung unter einigen Objekt-Layern wie Straßen gezeichnet werden, während sie über anderen Hintergrund-Layern angezeigt wird. Verfügbare Optionen sind:
  - *Unter Karte*
  - *Unter Kartenlayer* und *Über Kartenlayer*: platzieren den Übersichtsrahmen unterhalb bzw. oberhalb der Geometrien eines Layers. Der Layer wird mit der Option *Stapellayer* ausgewählt.

- *Unter Kartenbeschriftungen*: Da Beschriftungen immer über allen Objekten in einem Kartenelement dargestellt werden, wird der Übersichtsrahmen über allen Geometrien und unter jeder Beschriftung angeordnet.
- *Über Kartenbeschriftungen*: platziert den Übersichtsrahmen über allen Geometrien und Beschriftungen im Kartenelement.

18.2.3 Das 3D Kartenelement

Das Element 3D-Karte wird verwendet, um eine *3D-Kartenansicht* in einem Druck-Layout anzuzeigen. Verwenden Sie die Schaltfläche *3D-Karte hinzufügen* und folgen Sie den *Anleitungen zum Erstellen eines Layout-Elements*, um ein neues 3D-Karten-Element hinzuzufügen. Dieses können Sie dann auf die gleiche Weise bearbeiten, wie in *Interaktion mit Layout-Elementen* gezeigt.

Standardmäßig ist ein neues 3D-Kartenelement leer. Sie können die Eigenschaften der 3D-Ansicht im Bedienfeld *Elementeigenschaften* einstellen und anpassen. Zusätzlich zu den *gemeinsamen Eigenschaften* hat dieses Objekt die folgenden Funktionalitäten (Abb. 18.23):

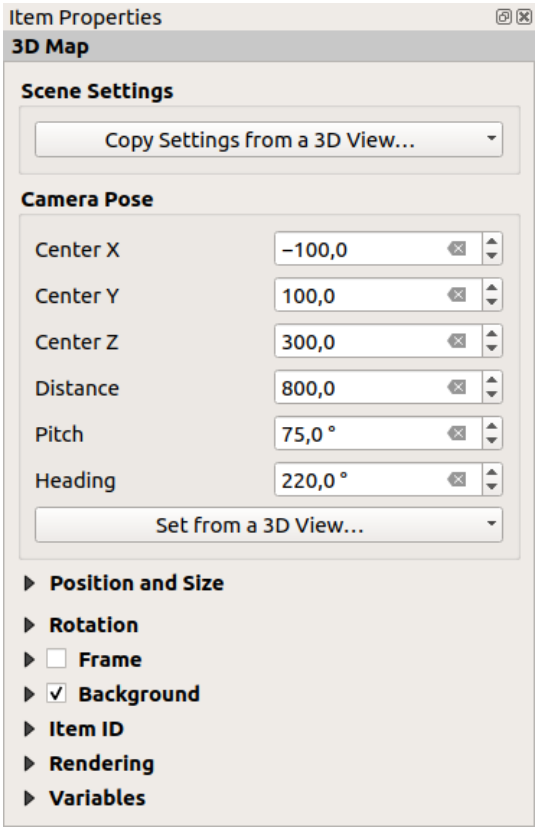


Abb. 18.23: Übersicht über die Elementeigenschaften eines 3D-Karten-Elements

Szeneneinstellungen

Drücken Sie *Einstellungen aus einer 3D-Ansicht kopieren…*, um die 3D-Kartenansicht auszuwählen, die angezeigt werden soll.

Die 3D-Kartenansicht wird mit ihrer aktuellen Konfiguration gerendert (Layer, Terrain, Lichter, Kameraposition und -winkel…).

Kamera-Pose

- *X-Zentrum* legt die X-Koordinate des Punktes fest, auf den die Kamera gerichtet ist
- *Y-Zentrum* legt die Y-Koordinate des Punktes fest, auf den die Kamera gerichtet ist
- *Z-Zentrum* legt die Z-Koordinate des Punktes fest, auf den die Kamera gerichtet ist
- *Entfernung* legt die Entfernung von der Kameramitte zu dem Punkt fest, auf den die Kamera gerichtet ist
- *Nick* bestimmt die Drehung der Kamera um die X-Achse (vertikale Drehung) in einem Wert von 0° bis 360°. 0°: Gelände gerade von oben gesehen; 90°: horizontal (von der Seite); 180°: gerade von unten; 270°: horizontal, auf dem Kopf stehend; 360°: gerade von oben.
- *Überschrift* legt die Drehung der Kamera um die Y-Achse fest (horizontale Drehung in einem Wert von 0° bis 360°. 0°/360°: Norden; 90°: Westen; 180°: Süden; 270°: Osten.

Mit dem Pulldown-Menü *Aus einer 3D-Ansicht setzen…* können Sie die Elemente mit den Parametern einer 3D-Ansicht füllen.

18.2.4 Das Beschriftungs-Element

Das Element *Beschriftung* ist ein Werkzeug, das Ihnen hilft, Ihre Karte mit Texten zu versehen, die zum Verständnis beitragen; das kann der Titel, der Autor, die Datenquellen oder jede andere Information sein. Sie können ein Label mit dem Werkzeug *Beschriftung hinzufügen* wie in der *Anleitung zur Erstellung von Elementen* erklärt, hinzufügen und es dann auf die gleiche Weise bearbeiten wie in *Interaktion mit Layout-Elementen* beschrieben.

Wenn Sie ein neues Beschriftungs-Element hinzufügen, enthält es einen Standardtext, den Sie über das Bedienfeld *Elementeigenschaften* ändern können. Abgesehen von den *gemeinsamen Eigenschaften*, hat dieses Elememt die folgenden Funktionen (siehe Abb. 18.24):

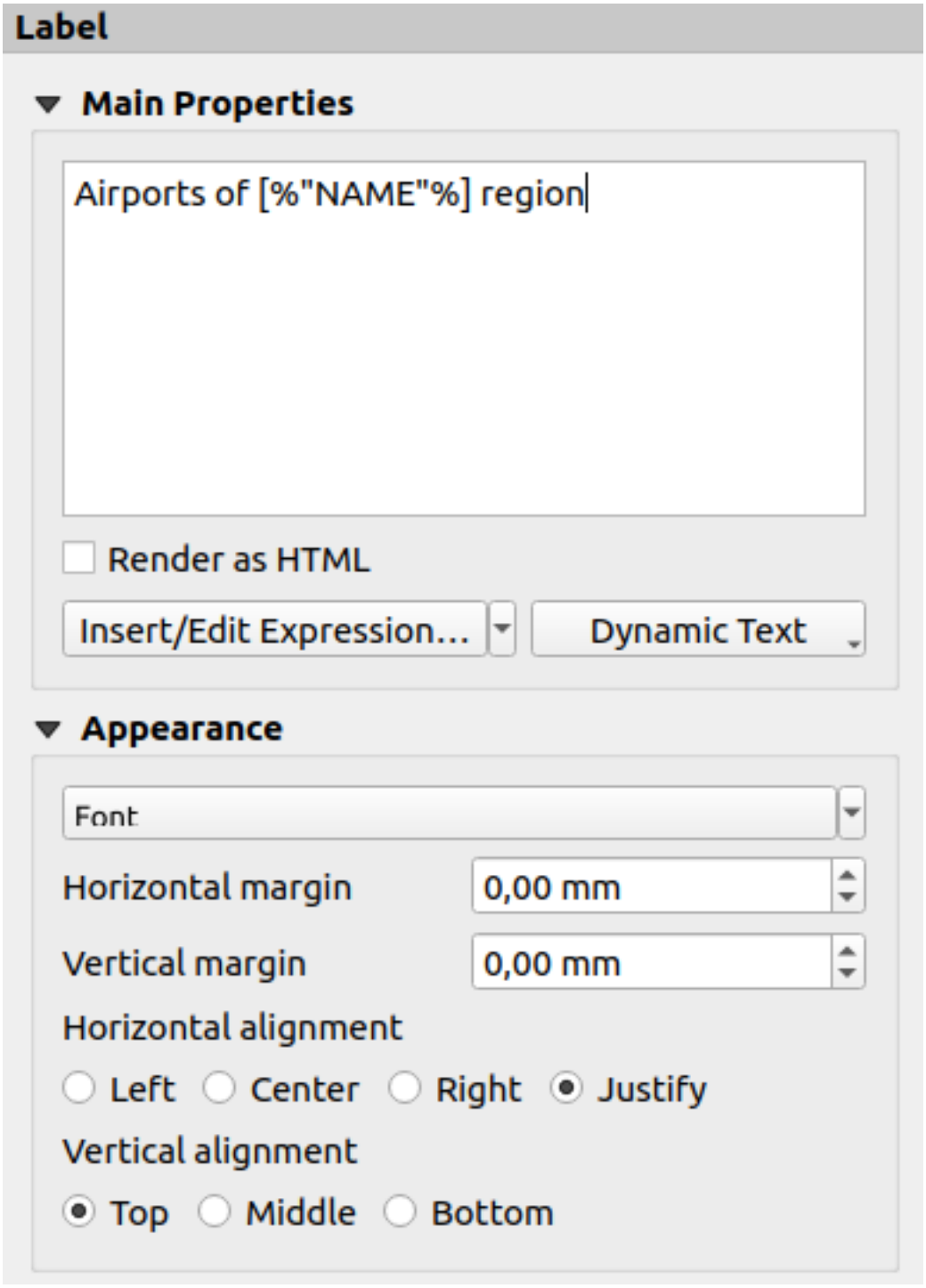


Abb. 18.24: Übersicht über die Elementeigenschaften eines Beschriftungs-Elements

Haupteigenschaften

Die Gruppe *Haupteigenschaften* ist der Ort, an dem der Text des Labels angegeben wird. Der Text kann statisch, dynamisch mit *Ausdrücken* und Variablen und/oder mit HTML formatiert sein. Dynamische Teile einer Beschriftung müssen von `[%` und `%]` umgeben sein, damit sie als solche interpretiert und ausgewertet werden.

- Um Ausdrücke in Beschriftungen zu verwenden, können Sie auf die Schaltfläche *Ausdruck einfügen/bearbeiten…* klicken, Ihre Formel im *Ausdrucksеditor* bearbeiten und mit *OK* bestätigen. QGIS fügt dann automatisch die umgebenden Zeichen zu Ihrer Eingabe hinzu.

  Hinweis

  When you click the *Insert/Edit Expression…* button with no selection made in the textbox, the new expression will be appended to the existing text. However, if you want to modify an existing expression, the behavior changes based on your selection: If some text is selected and the selection is within an expression (between ‚[%' and ‚%]'), or if no text is selected but the cursor is inside an expression, the whole expression will be selected. In all other cases, the selection will remain as is.

  Da Karten in der Regel mit einigen allgemeinen Textinformationen (Datum, Autor, Titel, Seitenzahl, …) gefüllt sind, bietet QGIS einen direkten Zugriff auf verschiedene Ausdrücke oder Variablen: Drücken Sie den *Dynamischer Text*-Button, um sie auszuwählen und in Ihr Beschriftungsfeld einzufügen.

  Tipp

  In der Menüleiste oben in einem Kartenlayout können Sie den Eintrag *Element hinzufügen* ► *Dynamischen Text hinzufügen* ► verwenden, um ein neues Beschriftungs-Element zu erstellen, das automatisch mit einem diesen vordefinierten Ausdruck gefüllt wird.

  Es ist möglich, ein dynamisches Label in ein statisches umzuwandeln: Klicken Sie auf den Dropdown-Pfeil neben der Schaltfläche *Ausdruck einfügen/bearbeiten…* und wählen Sie *Statischen Text konvertieren*. Alle dynamischen Teile des Inhalts des Etiketts werden ausgewertet und durch ihre aktuellen Werte ersetzt. Sie können dann den resultierenden Text bei Bedarf manuell anpassen.

- Labels können als HTML-Code interpretiert werden: check *Als HTML darstellen*. Sie können jetzt HTML-Tags oder -Stile, eine URL, ein anklickbares Bild, das auf eine Webseite verweist, oder andere HTML-basierte Formatierungen einfügen.

Der folgende Code kombiniert HTML-Rendering mit Ausdrücken, als Beispiel für eine komplexe Beschriftung. Das Ergebnis ist in Abb. 18.25 darstellt:

```
<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>
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
   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>
```

Abb. 18.25: Nutzung von HTML-Formatierungen bei der erweiterten Gestaltung einer Beschriftung

Erscheinungsbild

- Definieren Sie Schriftart und Stil des Textes, indem Sie auf die Schaltfläche *Schriftart* klicken. Im Menü *Beschriftungsschriftart* können Sie einige der Optionen der Formatierung für Textelemente verwenden.
- Sie können verschiedene horizontale und vertikale Ränder in mm angeben. Dies ist der Abstand vom Rand des Layoutelements. Die Beschriftung kann außerhalb der Grenzen der Beschriftung positioniert werden, z.B. um Beschriftungselemente mit anderen Elementen auszurichten. In diesem Fall müssen Sie negative Werte für den Rand verwenden.
- Die Text-Ausrichtung ist eine weitere Möglichkeit, Ihr Beschriftung zu positionieren. Das kann sein:
  - *Links*, *Zentriert*, *Rechts* oder *Blocksatz* für *Horizontale Ausrichtung*
  - and *Top*, *Vertical Center*, *Bottom* for *Vertical alignment*.

Tipp

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.

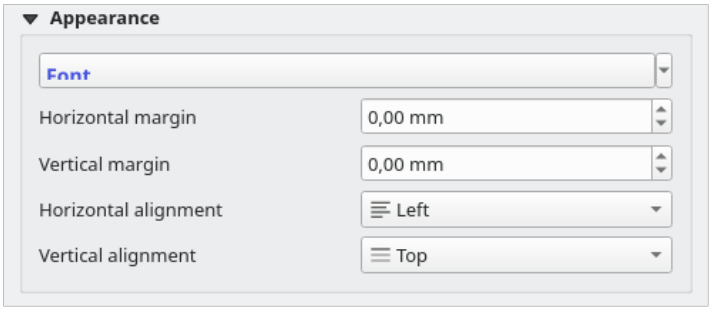


Abb. 18.26: Label Item Appearance Properties

Verwenden von Ausdrücken in Beschriftungs-Elementen

Nachfolgend finden Sie einige Beispiele für Ausdrücke, die Sie verwenden können, um die Beschriftung mit interessanten Informationen zu füllen. Beachten Sie, dass der Teil, der Ausdrücke enthält, im Texteingabefeld in den *Haupteigenschaften* von `[%` und `%]` umgeben sein muss.

- Anzeige eines Titels mit dem Wert des aktuellen Atlas-Objekts aus „field1 “:

```
'This is the map for ' || "field1"
```

  oder, wenn Sie es im Textfeld der *Haupteigenschaften* eingeben:

```
This is the map for [% "field1" %]
```

- Hinzufügen einer Seitennummerierung für verarbeitete Atlas-Objekte (z.B. `Seite 1/10`):

```
concat( 'Page ', @atlas_featurenumber, '/', @atlas_totalfeatures )
```

- Gibt den Namen aller Flughäfen in der Region des aktuellen Atlas-Objekts zurück, auf Grundlage des Attributes für die Regionszugehörigkeit (`fk_regionId`):

```
aggregate( layer := 'airports',
           aggregate := 'concatenate',
           expression := "NAME",
           filter := fk_regionId = attribute( @atlas_feature, 'ID' ),
           concatenator := ', '
         )
```

  Oder, wenn eine *Beziehung* über die Attribute gesetzt wurde:

```
relation_aggregate( relation := 'airports_in_region_relation',
                    aggregate := 'concatenate',
                    expression := "NAME",
                    concatenator := ', '
                  )
```

- Gibt den Namen aller Flughäfen in der Region des aktuellen Atlas-Objekts zurück, auf Grundlage ihrer räumlichen Beziehung:

```
aggregate( layer := 'airports',
           aggregate := 'concatenate',
           expression := "NAME",
           filter := contains( geometry( @parent ), $geometry ),
           concatenator := ', '
         )
```

  oder:

```
array_to_string( array:= overlay_contains( layer := 'airports',
                                           expression := "NAME" ),
                 delimiter:= ', '
               )
```

- Gibt die untere X-Koordinate der Ausdehnung des Elements `Map 1` zurück:

```
x_min( map_get( item_variables( 'Map 1' ), 'map_extent' ) )
```

- Ruft die Namen der Layer im aktuellen Layout-Element `Map 1` ab und formatiert die Namen zeilenweise:

```
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
)
```

- Zeigt eine Liste der Layer mit ihren Nutzungsrechten in einem Layoutelement `Map 1`. Diese Rechte müssen in den Eigenschaften der Layer auf dem Reiter *Metadaten* im Bereiche *Zugriff* eingetragen sein.

```
array_to_string( map_credits( 'Map 1', true ) )
```

18.2.5 Das Legenden-Element

Das Element *Legende* ist ein Kasten oder eine Tabelle, die die Bedeutung der in der Karte verwendeten Symbole erklärt. Eine Legende ist dann an ein Kartenelement gebunden. Sie können eine Legende mit dem Werkzeug *Legende hinzufügen* wie in der Anleitung zur Erstellung von Elementen erklärt, hinzufügen und sie dann auf die gleiche Weise bearbeiten wie in *Interaktion mit Layout-Elementen* beschrieben.

Standardmäßig zeigt die Legende alle verfügbaren Layer an und kann über das Bedienfeld *Elementeigenschaften* angepasst werden. Abgesehen von den *gemeinsamen Eigenschaften*, hat dieses Element die folgenden Funktionalitäten (siehe Abb. 18.27):

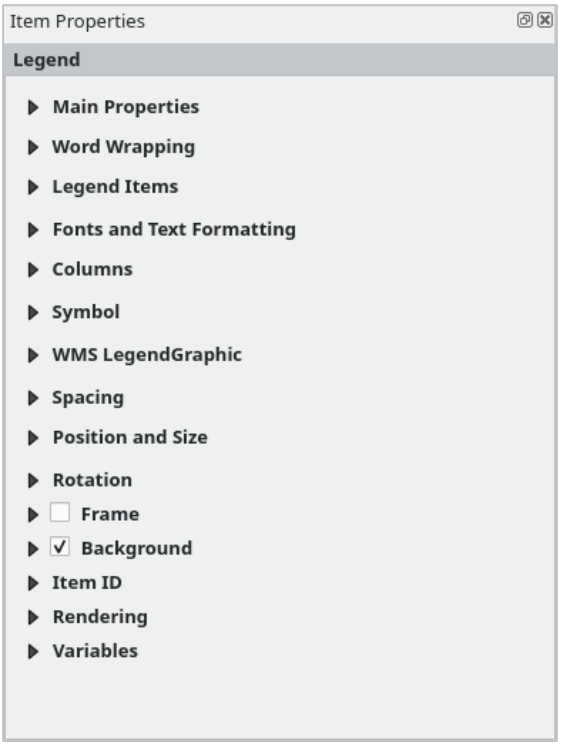


Abb. 18.27: Übersicht über die Elementeigenschaften eines Legenden-Elements

Haupteigenschaften

Der Bereich *Haupteigenschaften* im Bedienfeld *Elementeigenschaften* für Legenden-Elemente bietet die folgenden Funktionen (siehe Abb. 18.28):

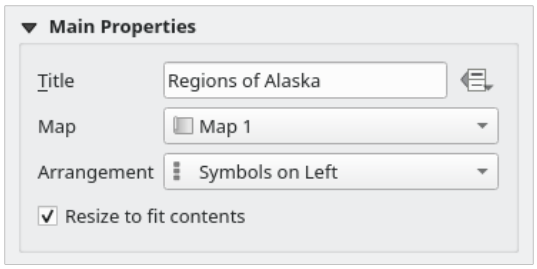


Abb. 18.28: Einstellung der Haupteigenschaften der Legende

In den Haupteigenschaften können Sie:

- den Legenden *Titel* ändern. Er kann durch die *Datendefinierte Übersteuerung* dynamisch gestaltet werden, was z.B. bei der Atlaserstellung hilfreich sein kann;
- Wählen Sie, auf welches *Karten*-Element sich die aktuelle Legende beziehen soll. Standardmäßig wird die Karte ausgewählt, über der die Legende gezeichnet wird. Wenn die Legende nicht über einer Karte gezeichnet wird, so wird auf die *Referenzkarte* zurückgegriffen.

> **Bemerkung**
>
> *Variablen* des verknüpften Karten-Elements (@map_id, @map_scale, @map_extent…) sind auch über das Menü der datendefinierten Übersteuerung der Legende zugänglich.

- Die Platzierung von Symbolen und Text in der Legende festlegen: *Arrangement* kann *Symbole auf der linken Seite* oder *Symbole auf der rechten Seite* sein. Der Standardwert hängt von dem verwendeten Gebietsschema ab (von rechts nach links oder nicht).
- Verwenden Sie *Größe an Inhalt anpassen*, um festzulegen, ob die Größe einer Legende automatisch an den Inhalt angepasst werden soll oder nicht. Wenn diese Option nicht aktiviert ist, wird die Legende nicht in der Größe angepasst, sondern bleibt bei der vom Benutzer eingestellten Größe. Alle Inhalte, die nicht in die Größe passen, werden abgeschnitten.

Word wrapping

The *Word wrapping* group of the legend *Item Properties* panel provides the following functionalities:

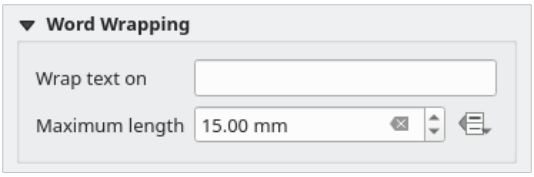


Abb. 18.29: Legend's word wrapping group

- Den Text der Legende an einem bestimmten Zeichen umbrechen: Jedes Mal, wenn das Zeichen erscheint, wird es durch einen Zeilenumbruch ersetzt;
- 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
  ```

Legendenelemente

Der Bereich *Legendenelemente* des Bedienfelds *Elementeigenschaften* bietet die folgenden Funktionen (siehe Abb. 18.30):

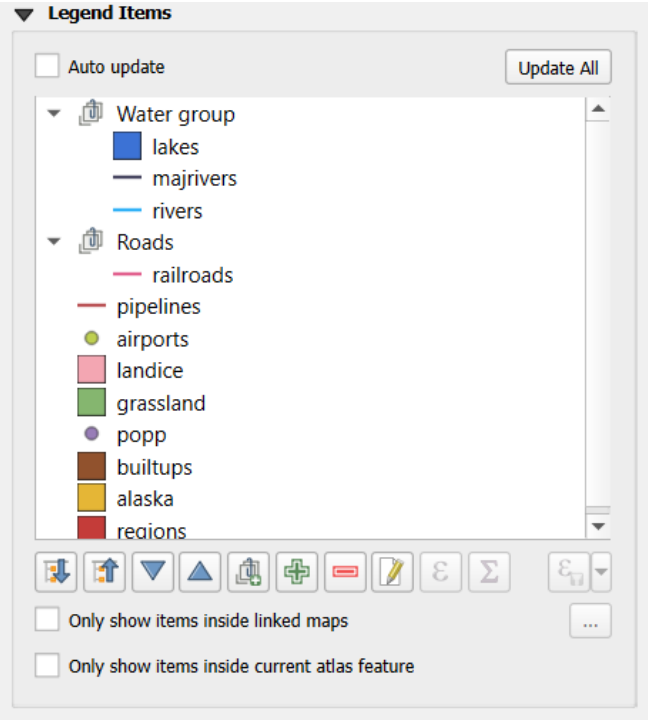


Abb. 18.30: Der Bereich Legendenelemente

- Die Legende wird automatisch aktualisiert, wenn das Kontrollkästchen *Automatisch aktualisieren* aktiviert ist. Wenn *Automatisch aktualisieren* nicht angehakt ist, haben Sie mehr Kontrolle über die Legendenelemente. Alle Icons unterhalb der Legendenelemente sind dann aktiviert.
- Im Fenster mit den Legendenelementen werden alle Legendenelemente aufgelistet und Sie können die Reihenfolge der Elemente ändern, Layer gruppieren, Elemente aus der Liste entfernen und wiederherstellen, Layernamen und Symbole bearbeiten und einen Filter hinzufügen.
  - Verwenden Sie Alle ausklappen und Alle einklappen Schaltflächen, um alle Gruppen und Untergruppen im Legendenbaum auszuklappen oder einzuklappen. Stellen Sie sicher, dass das Kontrollkästchen *Automatisch aktualisieren* deaktiviert ist, um diese Schaltflächen verwenden zu können.
  - Die Reihenfolge der Elemente kann mit den Schaltflächen und oder mit der „Drag-and-Drop“-Funktionalität geändert werden. Für WMS-Legendengrafiken kann die Reihenfolge nicht geändert werden.
  - Verwenden Sie die Schaltfläche , um eine Legendengruppe hinzuzufügen.
  - Verwenden Sie die Schaltfläche , um Layer hinzuzufügen, und die Schaltfläche , um Gruppen, Layer oder Symbolklassen zu entfernen.
  - Die Schaltfläche dient zur Bearbeitung des Layers, des Gruppennamens oder des Titels. Zuerst müssen Sie das Legendenelement auswählen. Ein Doppelklick auf das Element öffnet auch das Textfeld, um es umzubenennen.
  - Die Schaltfläche verwendet Ausdrücke, um jedes Symbol-Label des ausgewählten Layers anzupassen (siehe *Datendefinierte Beschriftungen in einer Legende*)
  - Die Schaltfläche addiert die Anzahl der Objekte für jede Klasse des Vektorlayers.
  - Die Funktion Filterausdruck hilft Ihnen zu filtern, welche der Legendenelemente eines Layers angezeigt werden, d.h. wenn Sie einen Layer verwenden, der verschiedene Legendenelemente hat (z.B. aus einer regelbasierten oder kategorisierten Symbologie), können Sie einen booleschen Ausdruck angeben, um

Stile aus dem Legendenbaum zu entfernen, die kein Objekt haben, das eine Bedingung erfüllt. Beachten Sie, dass die Objekte dennoch beibehalten und im Layout-Kartenelement angezeigt werden.

Während die Legende standardmäßig den Baum des Bedienfelds *Layer* nachahmt und die gleichen Gruppen, Layer und Symbolklassen anzeigt, können Sie mit einem Rechtsklick auf ein beliebiges Element den Namen des Layers ausblenden oder ihn als Gruppe oder Untergruppe anzeigen lassen. Falls Sie Änderungen an einem Layer vorgenommen haben, können Sie diese rückgängig machen, indem Sie *Reset to defaults* aus dem Kontextmenü des Legendeneintrags wählen.

Nachdem Sie die Symbologie im QGIS-Hauptfenster geändert haben, können Sie auf *Alle aktualisieren* klicken, um die Änderungen im Legendenelement des Drucklayouts anzupassen.

- Mit dem Kontrollkästchen *Nur Elemente in verknüpften Karten anzeigen* werden nur die in der verknüpften Karte sichtbaren Legendenelemente in der Legende aufgeführt. Wenn Sie mehr als eine Karte haben, können Sie auf … klicken und andere Karten in Ihrem Layout auswählen. Dieses Werkzeug bleibt verfügbar, wenn das Kontrollkästchen *Automatisch aktualisieren* aktiviert ist.
- Bei der Erstellung eines Atlas mit Polygon-Features können Sie Legendenelemente herausfiltern, die außerhalb des aktuellen Atlas-Features liegen. Aktivieren Sie dazu die Option *Nur Objekte im aktuellen Atlas-Feature anzeigen*.

Datendefinierte Beschriftungen in einer Legende

Die Schaltfläche ε ermöglicht es Ihnen, *Ausdrücke* zu jeder Symbol-Beschriftung eines bestimmten Layers hinzuzufügen. Variablen (`@symbol_label`, `@symbol_id` und `@symbol_count`) helfen Ihnen bei der Interaktion mit dem Legendeneintrag.

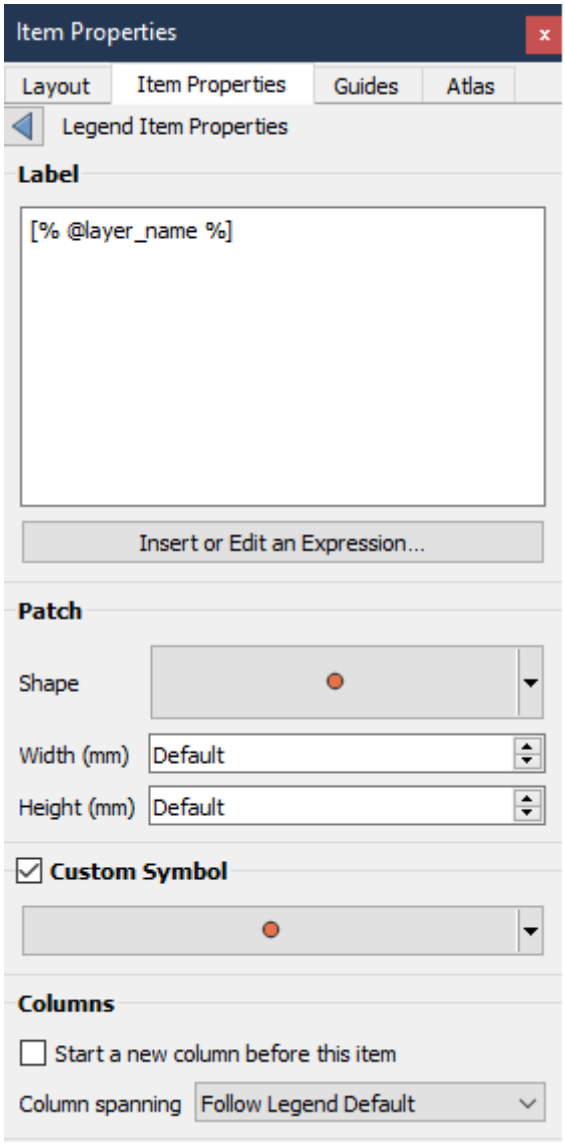
Bei einem Layer `regions`, der durch sein Feld `type` kategorisiert ist, können Sie beispielsweise an jede Klasse in der Legende die Anzahl der Objekte und die Gesamtfläche anhängen, z.B. `Borough (3) - 850ha`:

1. Wählen Sie den Layer-Eintrag im Legendenbaum
2. Drücken Sie die Schaltfläche ε, um den Dialog *Ausdruckserstellung* zu öffnen
3. Geben Sie den folgenden Ausdruck ein (*unter der Voraussetzung, dass die Symbolbeschriftungen nicht bearbeitet wurden*):

```
format( '%1 (%2) - %3ha',
        @symbol_label,
        @symbol_count,
        round( aggregate(@layer, 'sum', $area, filter:= "type"=@symbol_label)/
↪10000 )
      )
```

4. Klicken Sie auf *OK*

Legenden-Elemente anpassen



Legendenelemente können auch individuell in den *Legendenelementeigenschaften* angepasst werden. Diese Anpassungen können jedoch nur vorgenommen werden, wenn die Checkbox *Automatisch aktualisieren* deaktiviert ist.

Ein Doppelklick auf ein Element oder das Drücken von Gewählte Elementeigenschaften bearbeiten ermöglicht weitere Anpassungen:

Beschriftung

Bei allen Elementtypen kann der Beschriftungstext durch Eingabe oder durch Einfügen von Ausdrücken mithilfe des Befehls *Gewählte Elementeigenschaften bearbeiten* geändert werden. Ausdrücke können auch direkt an einer beliebigen Stelle in der Beschriftung des Eintrags eingefügt werden, indem die Notation [% Ausdruck %] verwendet wird.

Spalten

Mit der Eigenschaft Legendenelement können Sie auch das Verhalten der Spaltenaufteilung steuern, indem Sie erzwingen, dass die Spaltenaufteilung nach einem bestimmten Element oder nach allen Symbolen eines Layers erfolgt.

Die automatische Aufteilung eines Layers und seiner untergeordneten Elemente kann in diesem Widget auch auf Layer-Basis zugelassen oder blockiert werden.

Symbol

Für Elemente mit einem Symbol können Sie hier die maximale Höhe und Breite angeben, die ein Symbol einnehmen kann.

Für Vektorsymbole können Sie eine eigene Form für das Symbol angeben. Die Formen werden normalerweise durch einen Ausdruck definiert, um die Geometrie in einer einfachen Ebene darzustellen, aber diese Symbole können auch im Stilmanager gespeichert und später importiert werden. Das Standardsymbol für jeden Geometrietyp kann auch über den Stilmanager gesteuert werden.

Benutzersymbol

Für Vektorsymbole kann auch ein benutzerdefiniertes Symbol angegeben werden. Dies kann nützlich sein, um die Darstellung eines bestimmten Symbols zu optimieren, um es in der Legende zu verbessern oder um ein Symbol unabhängig von seiner eigentlichen Symbolvorschau zu haben. Dieses benutzerdefinierte Symbol hat Vorrang vor dem Symbol in der Legende, berücksichtigt aber die Angabe unter *Symbol*.

Schriftart und Textformatierung

Der Bereich *Schriftart und Textformatierung* des Bedienfeldes *Elementeigenschaften* bietet folgende Funktionen:

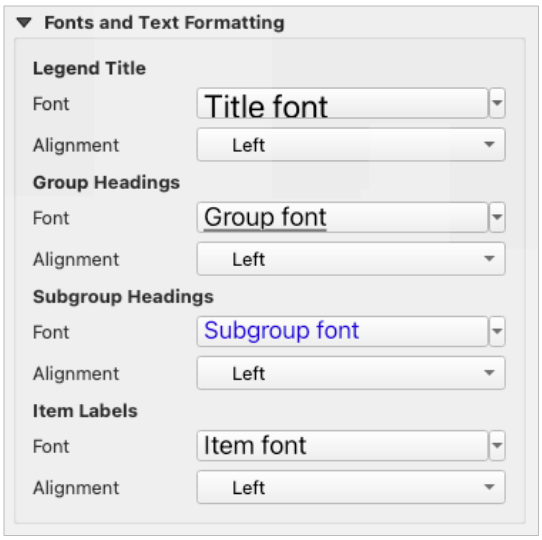


Abb. 18.31: Eigenschaften der Schriftarten

- Sie können die Schriftart des Legendentitels, der Gruppen und der Elemente (Objekte) in der Legende mit dem *Schriftauswahl*-Werkzeug ändern, das alle Möglichkeiten der *Textformatierung* bietet (Kontrolle des Schriftabstands, gemischte HTML-Formatierung, Farbgebung, Überblendung, Hintergrund, Textpuffer, Schatten, …)
- Für jede dieser Ebenen können Sie die *Ausrichtung* des Textes festlegen: Er kann *Links* (Standard für links-nach-rechts basierte Sprachumgebungen), *Zentriert* oder *Rechts* (Standard für rechts-nach-links basierte Sprachumgebungen) sein.

Spalten

Unter der Gruppe *Spalten* des Bedienfelds *Elementeigenschaften* können Legendenelemente über mehrere Spalten angeordnet werden:

- Legen Sie die Anzahl der Spalten im Feld *Anzahl* 1,00 fest. Dieser Wert kann datendefiniert gesteuert werden, z.B. nach Atlasobjekten, Legendeninhalten, der Rahmengröße…
- *Gleiche Spaltenbreite* stellt ein, wie Legendenspalten angepasst werden sollen.
- Die *Layer aufteilen* Option ermöglicht es eine kategorisierte oder abgestufte Layerlegende in Spalten aufzuteilen.

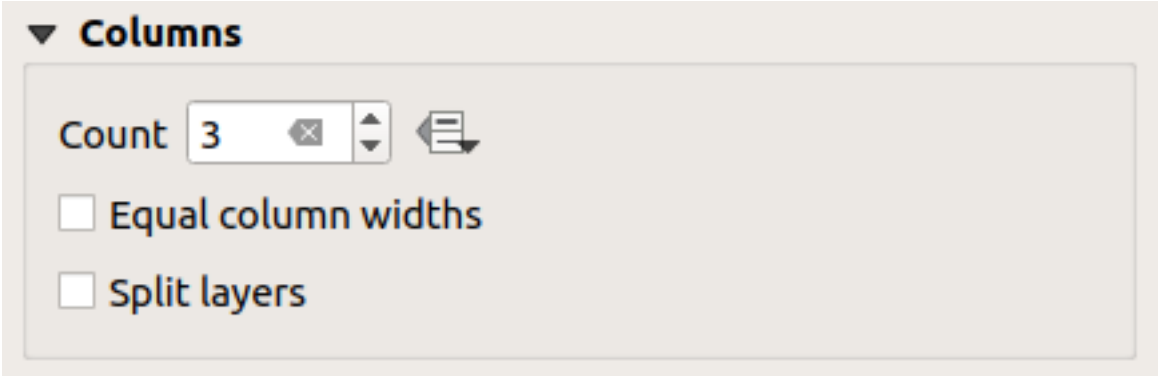


Abb. 18.32: Einstellungen für Legendenspalten

Symbol

Im Bereich *Symbol* des Bedienfelds *Elementeigenschaften* können Sie die Größe der Symbole, die neben den Legendenbeschriftungen angezeigt werden, konfigurieren. Sie können:

- Die *Symbolbreite* und *Symbolhöhe* setzen
- Setzen Sie die *Min. Symbolgröße* und *Max. Symbolgröße* der Marker: `0.00 mm` bedeutet, dass kein Wert gesetzt ist.
- *Randstrich um Rastersymbole zeichnen*: fügt dem Symbol einen Umriss hinzu, der die Bandfarbe des Raster-Layers darstellt; Sie können sowohl die *Strichfarbe* als auch die *Dicke* einstellen.

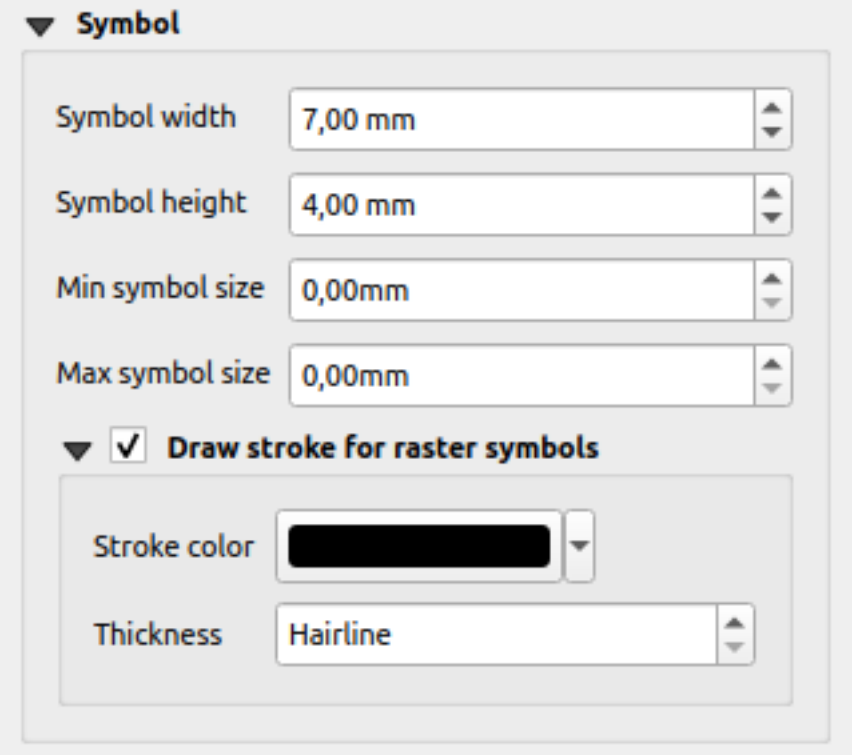


Abb. 18.33: Einstellungen für das Legenden-Symbol

WMS-LegendGrafik

Der Abschnitt *WMS-LegendGrafik* des Bedienfelds *Elementeigenschaften* bietet die folgenden Funktionalitäten (siehe Abb. 18.34):

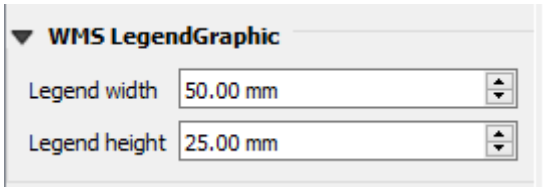
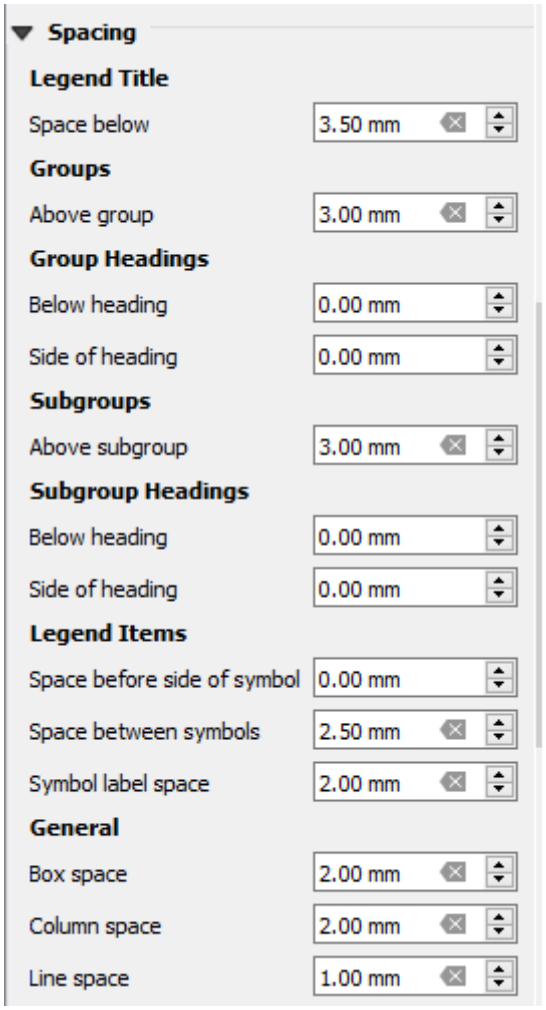


Abb. 18.34: WMS-LegendGrafik

Wenn Sie einen WMS Layer hinzugefügt haben und ein Legendenelement einfügen, wird eine Anfrage an den WMS Server gesendet, um eine WMS Legende bereitzustellen. Diese Legende wird nur angezeigt, wenn der WMS-Server die Funktion GetLegendGraphic bereitstellt. Der Inhalt der WMS-Legende wird als Rasterbild bereitgestellt.

WMS-LegendGrafik wird verwendet, um die *Legendenbreite* und die *Legendenhöhe* des WMS Lengendenrasterbildes anzupassen.

Zwischenräume



Mit dem Abschnitt *Zwischenräume* können Sie die Abstände innerhalb der Legende anpassen. Die Abstände können erheblich dazu beitragen, die Gruppierung der Elemente in der Legende und ihre Beziehung zueinander zu verdeutlichen.

Zwischenräume um und vor Titeln, Gruppen, Untergruppen, Symbolen, Beschriftungen, Kästen, Spalten und Zeilen können über diesen Dialog angepasst werden.

18.2.6 Das Maßstabs-Element

Maßstabsbalken dienen der visuellen Darstellung der Größe von Objekten und der Entfernung zwischen Objekten auf dem Karten-Element. Ein Maßstabsbalken-Element erfordert ein Karten-Element. Verwenden Sie das Werkzeug *Maßstab hinzufügen* wie in der *Anleitung zum Erstellung von Elementen* beschrieben, um ein neues Maßstabs-Element hinzuzufügen. Dies können Sie dann, wie in *Interaktion mit Layout-Elementen* dargestellt, bearbeiten.

Standardmäßig zeigt ein neuer Maßstabsbalken den Maßstab des Karten-Elements an, über dem er gezeichnet wird. Befindet sich kein Karten-Element darunter, wird die *Referenzkarte* verwendet. Sie können den Maßstab im Bedienfeld *Elementeigenschaften* anpassen. Abgesehen von den *gemeinsamen Eigenschaften* verfügt dieses Objekt über die folgenden Funktionen (siehe Abb. 18.35):

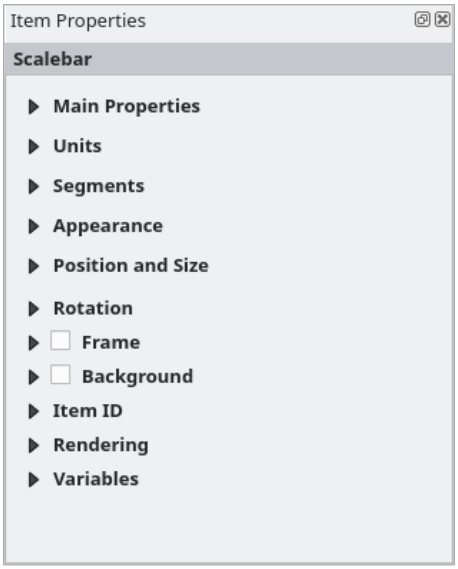


Abb. 18.35: Übersicht über die Elementeigenschaften eines Maßstabs-Elements

Haupteigenschaften

Die Gruppe *Haupteigenschaften* des Bedienfelds *Elementeigenschaften* bietet die folgenden Funktionen (siehe Abb. 18.36):

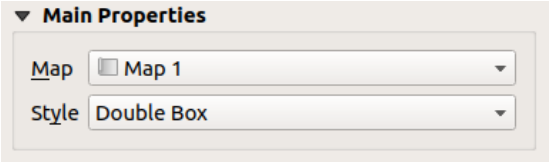


Abb. 18.36: Die Haupteigenschaften eines Maßstabs-Elements

1. Wählen Sie zunächst die Karte aus, die der Maßstabsbalken repräsentieren soll
2. Wählen Sie dann den Stil des Maßstabsbalkens. Verfügbare Stile sind:
   - **Einfacher Rahmen** oder **Doppelter Rahmen**: der Maßstab besteht aus einem oder zwei Kästchen in wechselnden Farben;
   - **Mittige**, **untere** oder **obere** Linienstriche;
   - **Gestufte Linie** zeichnet den Maßstab in einer gestuften Darstellung;
   - **Hohl** einfacher Rahmen, wobei aber jedes zweite Segment mit einer horizontalen Linie geteilt ist
   - **Numerisch**, wobei nur der Maßstab angezeigt wird (z. B. `1:50000`).
3. Eigenschaften nach Bedarf einstellen

Einheiten

Die Gruppe *Einheiten* des Bedienfelds *Elementeigenschaften* bietet die Möglichkeit, die Anzeigeeinheiten und einige Textformatierungen einzustellen (siehe Abb. 18.37):

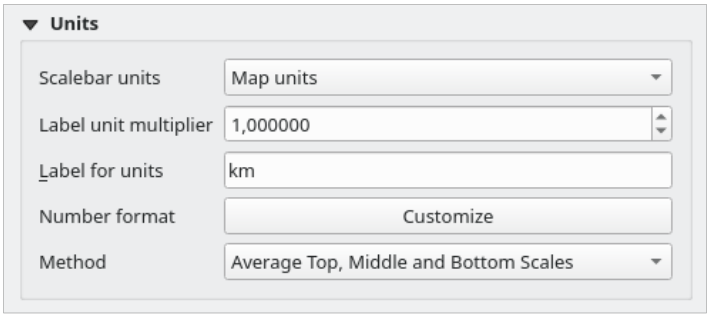


Abb. 18.37: Der Einheiten-Bereich eines Maßstabs-Elements

- Mit *Maßleisteneinheiten* wählen Sie die Einheiten, die Sie verwenden möchten. Es verschiedene zur Auswahl: **Karteneinheiten** (die Standardeinheit), **Meter**, **Fuß**, **Meilen** oder **Seemeilen** und einige Ableitungen.
- Der *Beschriftungseinheitsfaktor* gibt an, wie viele Skalenbalkeneinheiten pro beschriftete Einheit. Wenn z. B. die Maßstäbe auf „Meter“ eingestellt sind, ergibt ein Multiplikator von 1000 die Maßstäbe in „Kilometer“.
- Das Feld *Beschriftung für Einheiten* definiert den Text, der zur Beschreibung der Einheiten des Maßstabsbalkens verwendet wird, z.B. `m` oder `km`. Dieser sollte so angepasst werden, dass er den obigen Multiplikator widerspiegelt.
- Drücken Sie *Anpassen* neben *Zahlformat*, um alle Formatierungseigenschaften für die Zahlen im Maßstabsbalken zu kontrollieren, einschließlich Tausendertrennzeichen, Dezimalstellen, wissenschaftliche Notation usw. (siehe *Zahlen Formatierung* für weitere Details). Dies ist sehr nützlich, wenn Sie Karten für ein Publikum außerhalb des aktuellen QGIS Gebietsschemas erstellen, oder wenn Sie den Stil von den Gebietsschema-Vorgaben abweichen möchten (z.B. Tausendertrennzeichen hinzufügen, wenn die Gebietsschema-Vorgabe ist, diese auszublenden).
- 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.

Segmente

Der Bereich *Segmente* des Bedienfelds *Elementeigenschaften* bietet die Möglichkeit, die Anzahl und Größe der Segmente und Unterteilungen zu konfigurieren (siehe Abb. 18.38):

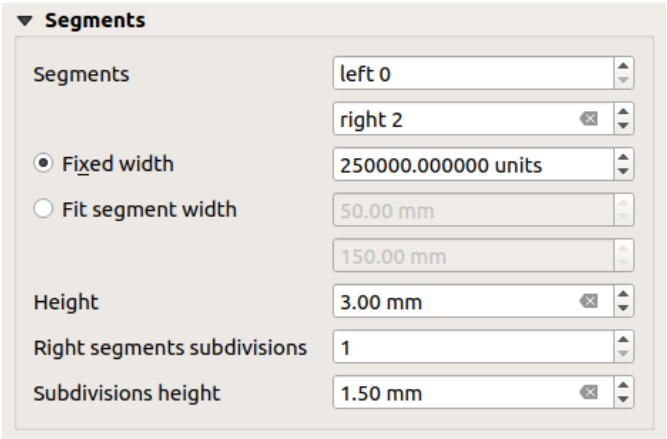


Abb. 18.38: Der Segmente-Bereich eines Maßstabs-Elements

- Sie können die Anzahl der *Segmente* festlegen, die links und rechts von der `0` des Maßstabsbalkens gezeichnet werden sollen:
  - Anzahl der Unterteilungen eines einzelnen Segments *Links* von der Null
  - Anzahl der Segmente *Rechts* von der Null
- Sie können die Breite eines Segments oder einen Bereich für die Gesamtlänge des Maßstabsbalkens festlegen:
  - Legen Sie fest, wie lang ein Segment in Ihren Maßstabseinheiten sein soll (*Feste Breite*)
  - Oder begrenzen Sie die Gesamtgröße des Maßstabsbalkens in `mm` mit der Option *An Segmentbreite anpassen*. Im letzteren Fall wird bei jeder Änderung des Kartenmaßstabs die Größe des Maßstabsbalkens angepasst (und seine Beschriftung aktualisiert), damit er zwischen den eingestellten oberen und unteren Bereich passt.
- *Höhe* wird benutzt, um die Höhe der Leiste einzustellen.
- *Rechte Segmentunterteilung* wird verwendet, um die Anzahl der Abschnitte zu definieren, die die rechten Segmente des Maßstabsbalkens haben können (für die Maßstabsbalkenstile *Linieneinteilung unten*, *Linieneinteilung mittig* und *Linieneinteilung oben*).
- *Unterteilungshöhe* wird verwendet, um die Höhe des Unterteilungssegments zu definieren.

Erscheinungsbild

Der Bereich *Erscheinungsbild* des Bedienfelds *Elementeigenschaften* bietet folgende Funktionen:

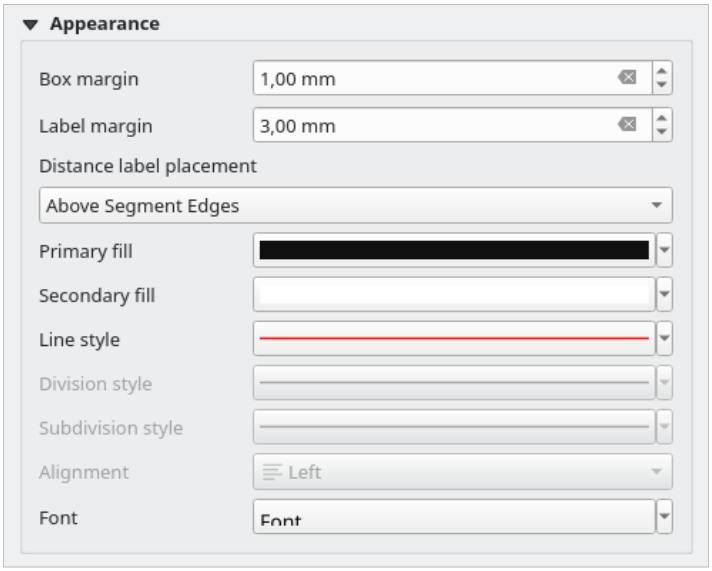


Abb. 18.39: Der Erscheinungsbild-Bereich einer Maßstabsleiste

Sie können festlegen wie der Maßstab in seinem Rahmen darsgestell wird.

- *Rahmenrand*: Zwischenraum zwischen Text- und Rahmengrenzen
- *Beschriftungsrand* : Abstand zwischen Text und Maßstabsbalken
- *Beschriftungsplatzierungsabstand* definiert die Position des Textes relativ zu den Segmenten der Maßstabsleiste: *Über Segmentkanten*, *Über Segmentzentren*, *Unter Segmentkanten* oder *Unter Segmentzentren*
- *Hauptfüllung* und *Nebenfüllung* der Skalenbalkenzeichnung unter Verwendung von *Füllsymboleigenschaften* (Farbe, Deckkraft, Muster, Effekte…) — für die Stile *Einfacher Rahmen*, *Doppelter Rahmen* und *Hohl*
- *Linienstil* des Maßstabsbalkens unter Verwendung von *Liniensymboleigenschaften* (Farbe, Strich, Verbindung, Kappenstil, Muster, Effekte…) — für alle außer *Numerisch* Stil

- *Teilungsstil* und *Unterteilungsstil* jeweils für Teilungs- und Unterteilungssegmente in den Skalenbalkenstilen *Linieneinteilung oben*, *Linieneinteilung mittig* und *Linieneinteilung unten* unter Verwendung von *Liniensymboleigenschaften* (Farbe, Strich, Verbindung, Kappenstil, Muster, Effekte…)
- *Ausrichtung* setzt den Text auf die linke, mittlere oder rechte Seite des Rahmens (nur für *Numerische* Maßstabsleisten)
- *Schriftart*, um die *Eigenschaften* (Größe, Schriftart, Farbe, Buchstabenabstand, Schatten, Hintergrund…) der Beschriftung des Maßstabsbalkens zu setzen.

Da die meisten Anzeigeeigenschaften des Maßstabsbalkens auf Symbolen beruhen, deren Eigenschaften datendefiniert werden können, ist es möglich, datendefinierte Maßstabsbalken darzustellen.

Beispiel: Der folgende Code, der auf die Eigenschaft „Fett“ der Skalenbeschriftungen angewendet wird, zeigt Zahlen in Fettschrift an, wenn sie ein Vielfaches von 500 sind:

```
-- returns True (or 1) if the value displayed on the bar
-- is a multiple of 500

@scale_value % 500 = 0
```

18.2.7 Das Tabellen-Element

Sie können Tabellen-Elemente verwenden, um Ihre Karte zu dekorieren und zu erklären:

- *Attributtabelle*: extrahiert automatisch eine Teilmenge der Attribute eines Layers basierend auf vordefinierten Regeln
- *Feste Tabelle*: ermöglicht die Erstellung von Tabellen mit manuell eingegebenen Inhalten (d. h. Tabellenkalkulationsstil), sodass Sie vollständig benutzerdefinierte Tabellen erstellen können.

Das Attributtabellen-Element

Für jeden Layer im Projekt können die Attribute im Drucklayout angezeigt werden. Verwenden Sie das Werkzeug *Attribut-Tabelle hinzufügen* wie in der *Anleitung zur Erstellung von Elementen* beschrieben, um ein neues Tabellen-Element hinzuzufügen. Dies können Sie dann, wie in *Interaktion mit Layout-Elementen* dargestellt, bearbeiten.

Standardmäßig lädt ein neues Attributtabellen-Element die ersten Zeilen des ersten (alphabetisch sortierten) Layers mit allen Feldern. Sie können die Tabelle jedoch dank des Bedienfelds *Elementeigenschaften* anpassen. Abgesehen von den *gemeinsamen Eigenschaften* verfügt dieses Element über die folgenden Funktionalitäten (siehe Abb. 18.40):

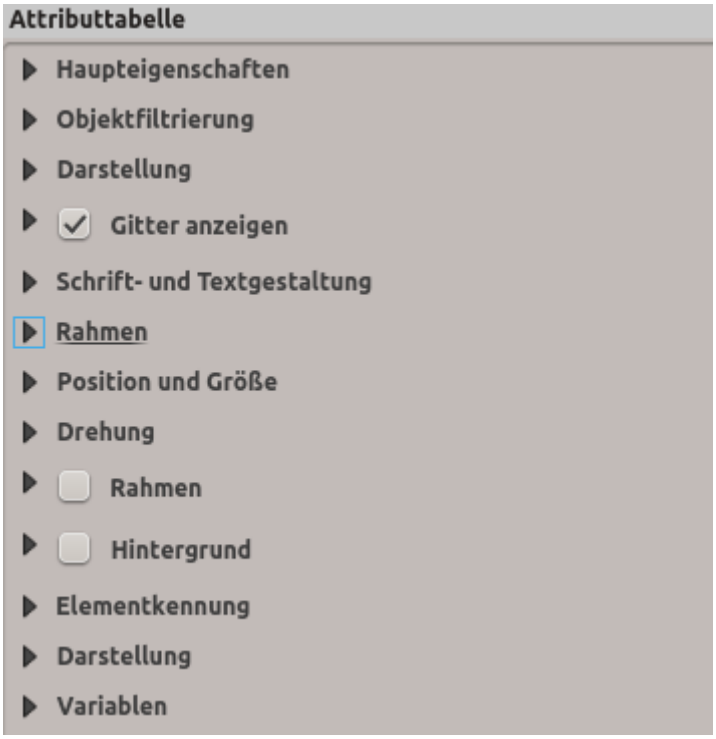


Abb. 18.40: Übersicht über die Elementeigenschaften eines Attributtabellen-Elements

Haupteigenschaften

Der Bereich *Haupteigenschaften* der Attributtabelle bietet die folgenden Funktionalitäten (siehe Abb. 18.41):

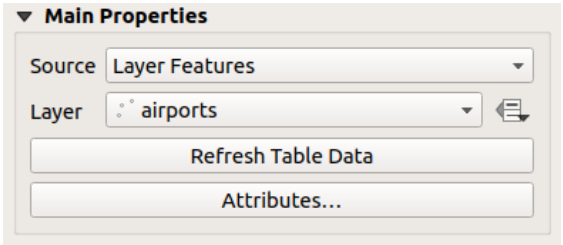


Abb. 18.41: Die Haupteigenschaften eines Attributtabellen-Elements

- Für *Quelle* können Sie standardmäßig nur **Layer-Objekte** auswählen, sodass Sie einen *Layer* aus den im Projekt geladenen Vektorlayern auswählen können.

  Die Schaltfläche Datendefinierte Übersteuerung neben der Layer-Liste ermöglicht es Ihnen, den Layer, der zur Befüllung der Tabelle verwendet wird, dynamisch zu ändern, z.B. könnten Sie die Attributtabelle mit verschiedenen Layer-Attributen pro Atlas-Seite füllen. Beachten Sie, dass die verwendeten Tabellenfelder (Abb. 18.44) diejenigen des Layers sind, der in der Dropdown-Liste *Layer* ausgewählt ist, und dass diese Struktur nicht verändert wird. Das bedeutet, dass eine datendefinierte Einstellung auf die Tabelle eines anderen Layers, mit anderen Feldern, zu leeren Spalten in der Tabelle führt.

  Falls Sie im Bedienfeld *Atlas* die Option *Einen Atlas erzeugen* aktiviert haben (siehe *Einen Atlas erzeugen*), stehen zwei weitere Optionen unter *Quelle* zur Verfügung:

  - **Aktuelles Atlas- Objekt** (siehe Abb. 18.42): es wird keine Option zur Auswahl des Layers angezeigt und das Tabellen-Element wird nur eine Zeile mit den Attributen des aktuellen des Atlas-Objekts anzeigen.
  - **Beziehungskinder** (siehe Abb. 18.43): Drop-down-Menü mit den Beziehungsnamen wird angezeigt. Diese Funktion kann nur verwendet werden, wenn Sie eine *Beziehung* mit Ihrem Atlas Coverage Layer als

übergeordnetes Objekt definiert haben. Die Tabelle zeigt dann die untergeordneten Zeilen des aktuellen Objekts des Atlas Coverage Layers.

- Die Schaltfläche *Tabellendaten aktualisieren* kann verwendet werden, um die Tabelle zu aktualisieren, wenn sich der aktuelle Inhalt der Tabelle geändert hat.

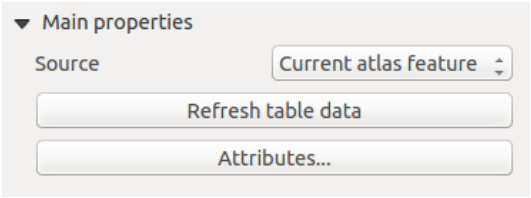


Abb. 18.42: Die Haupteigenschaften eines Attributtabellen-Elements bei der Auswahl ‚Aktuelles Atlas-Objekt'

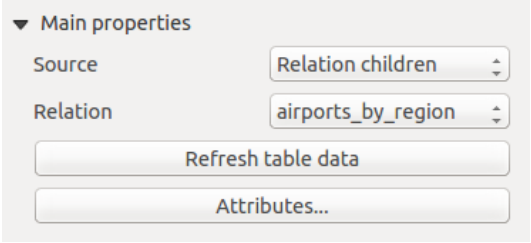


Abb. 18.43: Die Haupteigenschaften eines Attributtabellen-Elements bei der Auswahl „Beziehungskinder"

- Die Schaltfläche *Attribute…* startet den Dialog *Attribute wählen* (siehe Abb. 18.44), mit dem man den sichtbaren Inhalt der Tabelle ändern kann. Der obere Teil des Fensters zeigt die Liste der anzuzeigenden Attribute und der untere Teil hilft beim Sortieren der Daten.

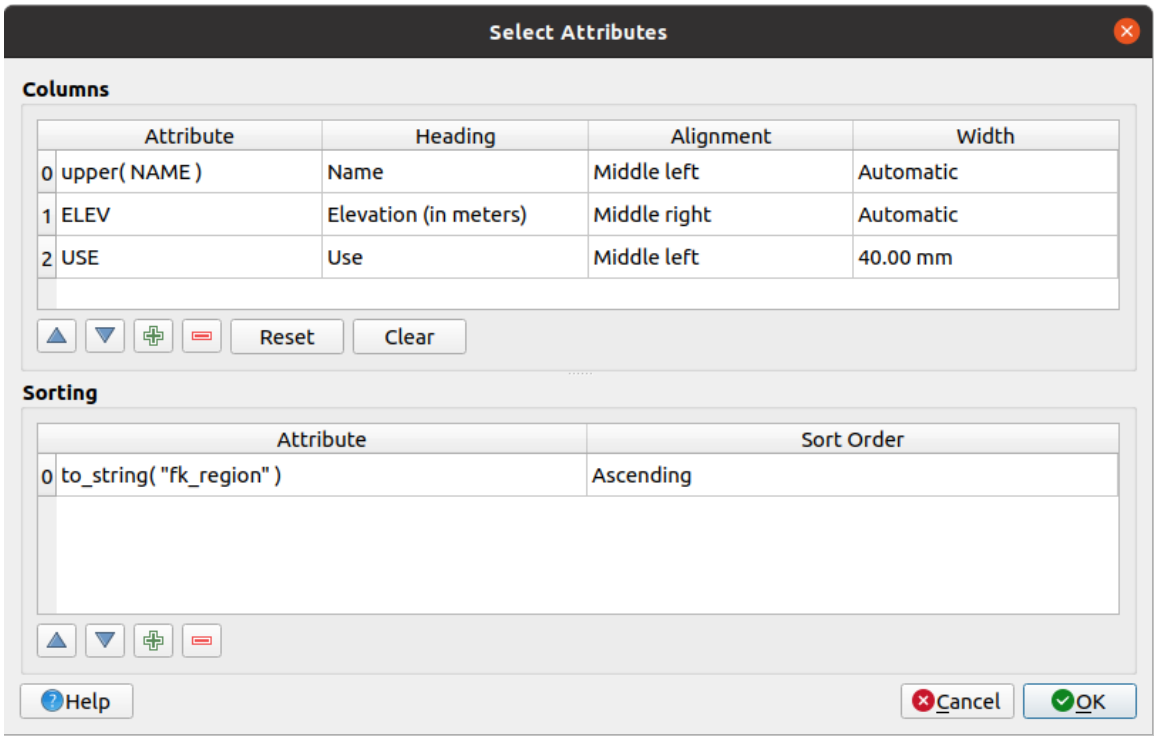


Abb. 18.44: Der Dialog Attribute wählen

Im oberen Abschnitt *Spalten* können Sie:

- Verschieben Sie Attribute in der Liste nach oben oder unten, indem Sie die Zeilen markieren und dann mit den Schaltflächen und die Zeilen verschieben. Es können mehrere Zeilen gleichzeitig ausgewählt und verschoben werden.
- Fügen Sie ein Attribut mit der Schaltfläche hinzu. Dadurch wird am unteren Ende der Tabelle eine leere Zeile eingefügt, in der Sie ein Feld als Attributwert auswählen oder ein Attribut über einen regulären Ausdruck erstellen können.
- Entfernen Sie ein Attribut mit der Schaltfläche . Es können mehrere Zeilen auf einmal ausgewählt und entfernt werden.
- Setzen Sie die Attributtabelle mit der Schaltfläche *Rücksetzen* auf ihren Standardzustand zurück.
- Löschen Sie die Tabelle mit der Schaltfläche *Leeren*. Dies ist nützlich, wenn Sie eine große Tabelle haben, aber nur eine kleine Anzahl von Attributen anzeigen möchten. Anstatt jede Zeile manuell zu entfernen, kann es schneller sein, die Tabelle zu löschen und die benötigten Zeilen hinzuzufügen.
- Zellüberschriften können durch Änderung des Textes in der Spalte *Überschrift* angepasst werden.
- Die Textausrichtung der Zellen kann mit der Spalte *Ausrichtung* verwaltet werden, die die Position des Textes innerhalb der Tabellenzelle festlegt.
- Die Zellenbreite kann manuell verwaltet werden, indem benutzerdefinierte Werte in die Spalte *Breite* eingetragen werden.

Im unteren Abschnitt *Sortierung* können Sie:

- Fügen Sie ein Attribut hinzu, nach dem die Tabelle sortiert werden soll: Drücken Sie die Schaltfläche und eine neue leere Zeile wird hinzugefügt. Fügen Sie ein Feld oder einen Ausdruck in die Spalte *Attribut* ein und setzen Sie die *Sortierreihenfolge* auf **Aufsteigend** oder **Absteigend**.
- Wählen Sie eine Zeile in der Liste aus und verwenden Sie die Schaltflächen und , um die Sortierpriorität auf Attributsebene zu ändern. Wenn Sie eine Zelle in der Spalte *Sortierung* auswählen, können Sie die Sortierreihenfolge des Attributfeldes ändern.
- Verwenden Sie die Schaltfläche , um ein Attribut aus der Sortierliste zu entfernen.

Objektfiltrierung

Der Bereich *Objektfiltrierung* der Attributtabelle bietet die folgenden Funktionalitäten (siehe Abb. 18.45):

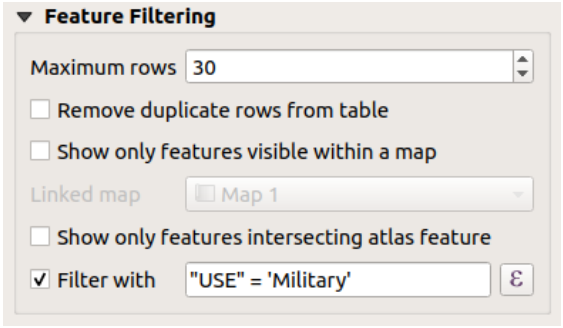


Abb. 18.45: Der Bereich Objektfiltrierung eines Attributtabellen-Elements

Sie können:

- Die *Maximalen Zeilen*, die dargestellt werden sollen, festlegen.
- *Doppelte Zeilen aus der Tabelle entfernen* aktivieren, um nur eindeutige Datensätze zu zeigen.

- Aktivieren Sie das Kontrollkästchen *Nur auf der Karte sichtbare Objekte anzeigen* und wählen Sie die entsprechende *Verknüpfte Karte*, deren Attribute für sichtbare Objekte angezeigt werden sollen.
- Die Option *Nur das Atlas-Objekt schneidende Objekte anzeigen* ist nur verfügbar, wenn *Einen Atlas erzeugen* aktiviert ist. Wenn die Option aktiviert ist, werden in der Tabelle nur Objekten angezeigt, die das aktuelle Atlas-Objekt schneiden.
- Aktivieren Sie :guilabel:`Filtern nach‘ und geben Sie einen Filter direkt in die Eingabezeile ein verwenden Sie die Schaltfläche um mit dem Ausdruckseditor einen Filter zu erzeugen. Ein paar Beispiele für Filteranweisungen, die Sie verwenden können, wenn Sie den Layer „airports“ aus dem Alaska-Datensatz geladen haben:
  - `ELEV > 500`
  - `NAME = 'ANIAK'`
  - `NAME NOT LIKE 'AN%'`
  - `regexp_match( attribute( $currentfeature, 'USE' ) , '[i]')`

  Der letzte reguläre Ausdruck wird nur die Flughäfen einfügen die einen Buchstaben ‚i‘ in der Attributspalte ‚USE‘ haben.

Erscheinungsbild

Der Bereich *Erscheinungsbild* der Attributtabelle bietet die folgenden Funktionalitäten (siehe Abb. 18.46):

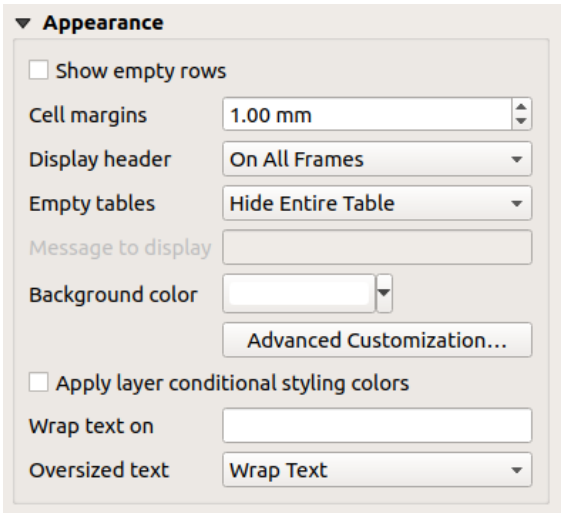


Abb. 18.46: Der Bereich Erscheinungsbild eines Attributtabellen-Elements

- Klicken Sie *Zeige leere Zeilen*, um die Attributtabelle mit leeren Zellen zu füllen, diese Option kann auch verwendet werden, um zusätzliche leere Zellen zu schaffen, wenn Sie ein Ergebnis anzeigen wollen.
- Mit *Zellenränder* können Sie die Ränder um den Text in jeder Zelle der Tabelle definieren.
- Mit *Kopf anzeigen* können Sie von einer Liste eine der voreingestellten Optionen ‚Im ersten Rahmen‘, ‚Auf allen Rahmen‘ oder ‚Kein Kopf‘ auswählen.
- Die Option *Leere Tabellen* kontrolliert, was dargestellt wird, wenn die Ergebnisauswahl leer ist:
  - **Nur Kopf zeichnen** zeichnet nur den Kopf, außer Sie haben ‚Kein Kopf‘ bei *Kopf anzeigen* ausgewählt.
  - **Ganze Tabelle ausblenden** zeichnet nur den Hintergrund der Tabelle. Sie können im Bereich *Rahmen* *Hintergrund nicht anzeigen, wenn Rahmen leer ist* aktivieren, um die Tabelle komplett auszublenden.

  - **Eingestellte Meldung anzeigen** zeichnet den Kopf und fügt eine Zelle ein, die sich über alle Spalten erstreckt und eine Nachricht wie ‚Kein Ergebnis' kann in der Option *Anzuzeigende Melkdung* bereitgestellt werden.

- Die Option *Anzuzeigende Meldung* ist nur aktiviert, wenn Sie **Eingestellte Meldung anzeigen** bei *Leere Tabelle* gewählt haben. Die angegebene Nachricht wird in der Tabelle in der ersten Zeile gezeigt, wenn das Ergebnis eine leere Tabelle ist.
- Mit *Hintergrundfarbe* können Sie die Hintergrundfarbe der Tabelle mit Hilfe des *FArbauswahl*-Werkzeugs festlegen. Mit der Option *Fortgeschrittene Anpassung* können Sie verschiedene Hintergrundfarben für jede Zelle definieren (siehe Abb. 18.47)

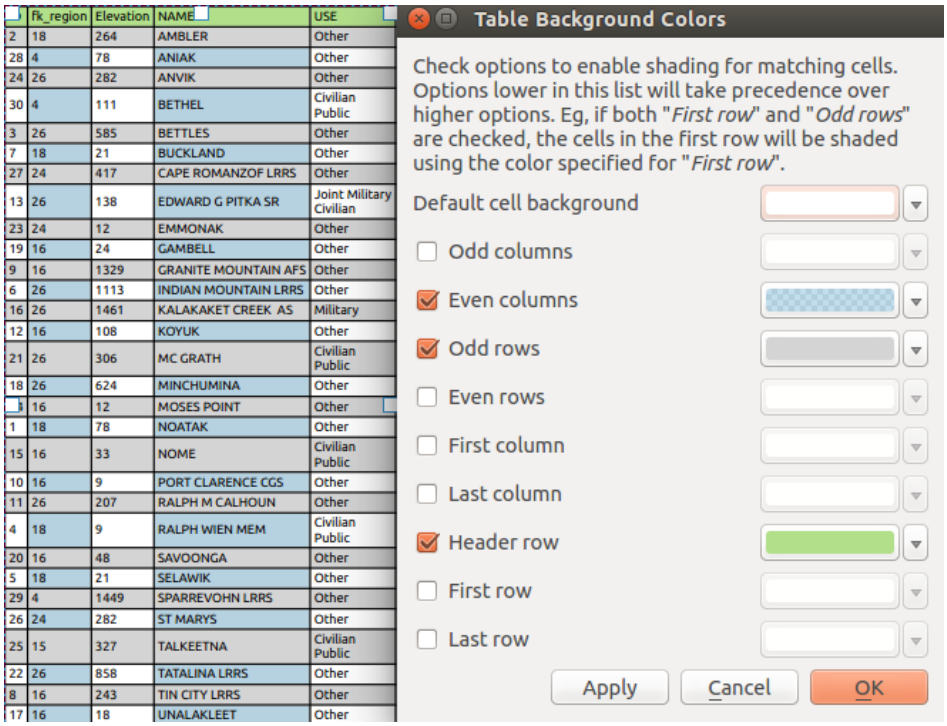


Abb. 18.47: Der Dialog Tabellenhintergrundfarben

- *Bedingte Layergestaltungsregeln verwenden*: die im Layer vorhandene *bedingte Tabellenformatierung* wird innerhalb der Layout-Attribut-Tabelle angewendet (Hintergrundfarbe, Schriftfamilie und Eigenschaften wie fett, kursiv, durchgestrichen, unterstrichen, Farbe, …). Bedingte Formatierungsregeln haben Vorrang vor anderen Formatierungseinstellungen in der Layouttabelle, z. B. haben sie Vorrang vor anderen Einstellungen für die Hintergrundfarbe von Zellen wie alternierenden Zeilenfarben.
- Mit der **:gui-Label:`Textumbruch bei`** Option können Sie ein Zeichen definieren, bei dem der Zellinhalt umgebrochen wird
- Mit *Übergroßer Text* definieren Sie das Verhalten, wenn die für eine Spalte eingestellte Breite kleiner ist als die Länge ihres Inhalts. Es kann **Text umbrechen** oder **Text abschneiden** sein.

Bemerkung

Weitere Eigenschaften des Attributtabellen-Elements werden im Anschluss im Abschnitt *Gemeinsame Funktionalitäten der Tabellen* beschrieben.

Das Feste Tabellenelement

Zusätzliche Informationen über die Karte können manuell in eine Tabelle eingefügt werden, indem man *Feste Tabelle hinzufügen* wählt und der Anleitung zur Erstellung von Layout-Elementen folgt, um ein neues Tabellenelement hinzuzufügen. Dieses können Sie dann, wie in *Interaktion mit Layout-Elementen* dargestellt, bearbeiten.

Standardmäßig wird im Kartenlayout eine leere Tabelle mit zwei minimierten Spalten und Zeilen angezeigt. Sie müssen die Tabelle im Bedienfeld *Elementeigenschaften* anpassen. Abgesehen von den *gemeinsamen Elementeigenschaften*, hat dieses Element die folgenden Funktionalitäten:

Haupteigenschaften

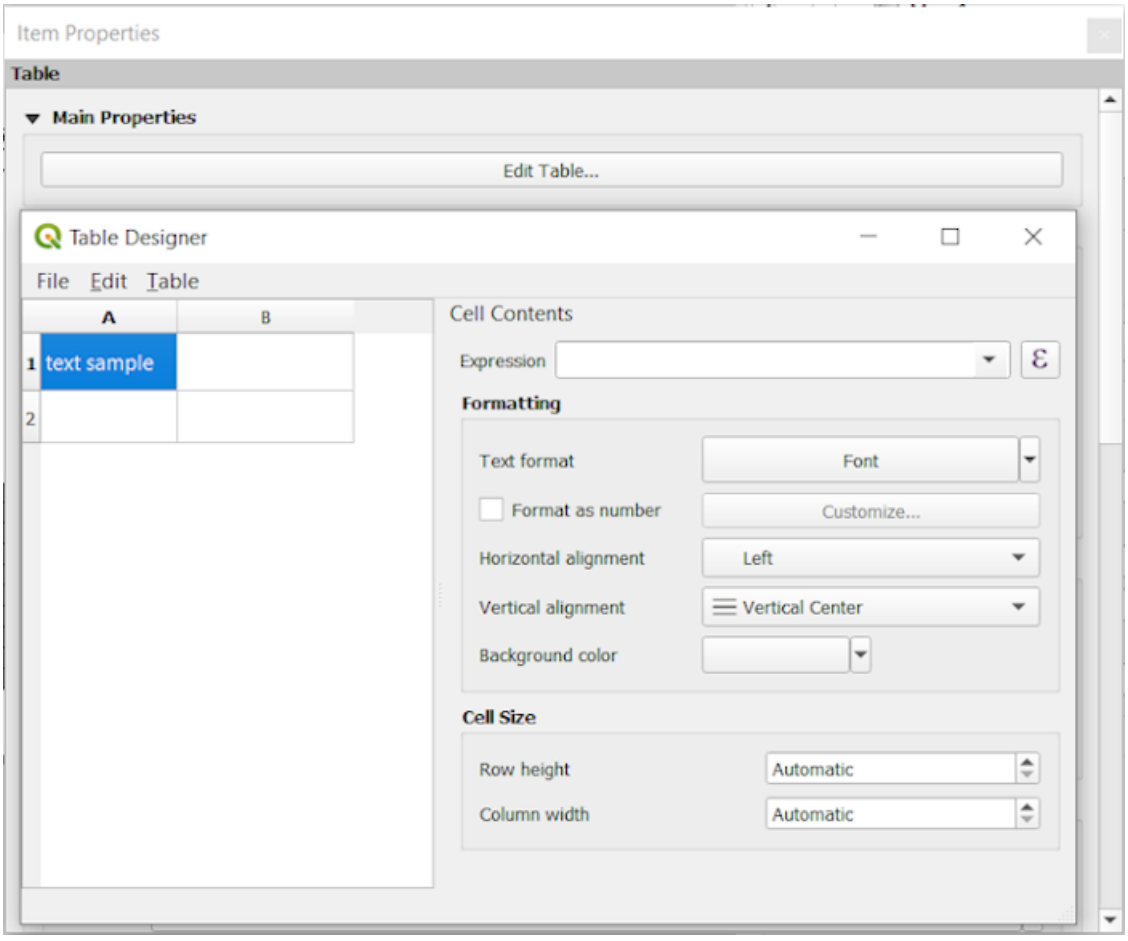


Abb. 18.48: Bearbeitungsfeld für ein festes Tabellen-Elemente

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):

- On the left hand of the dialog, a table frame displays the active table item with its content and settings. Click or double-click on a cell and write a text to modify its contents. Press `Ctrl+Enter` to add linebreak (for a multiline text). The logic for selecting multiple cells in the table is the same as in the vector layer *attribute table*. Grab and move a row or column separator to resize the row above or the column before. A double-click on the separator will resize the row above or the column before to fit their contents. A right-click over column or row headers shows a contextual menu to insert columns or rows around the selection or to delete them.
- Über die Menüs oben im Dialogfeld können Sie:
  - *Datei* ► *Inhalt aus Zwischenablage importieren*: den Inhalt der Tabelle überschreiben.

  - *Datei* ► *Editor schließen*: das Bearbeitungsfeld *Tabellenentwurf* schließen.
  - Über das Menü *Bearbeiten* mit Auswahlfunktionen für Zeilen und Spalten arbeiten:
    * *Alle wählen*: alle Zellen in der Tabelle wählen
    * *Zeilen wählen* oder *Spalten wählen*: wenn Zellen in der Tabelle ausgewählt sind, Sie können die Auswahl jeweils zu ihrer (ihren) Spalte(n) oder Zeile(n) ausweiten
    * *Zelle(n) leeren*: den Inhalt der ausgewählten Zellen löschen
  - Im Menü *Tabelle* die Struktur der Tabelle festlegen, und zwar:
    * *Insert rows* ► above or below the selection
    * *Insert columns* ► before or after the selection
    * *Zeilen löschen* oder *Spalten löschen*: Zeilen oder Spalten ausgewählter Zellen löschen
    * *Gewählte Zellen verbinden*: mehrere Zellen in einer rechteckigen Form auswählen und diese zusammenführen, wobei deren Inhalte in der Zelle oben links zusammengefasst werden (deren Formatierung auch auf die zusammengeführte Zelle angewendet wird, sofern sich keine Zelle der letzten Zeile/Spalte in der Auswahl befindet)
    * *Split selected cells*: select a merged cell and you can split it back to its individual cells. The current text is kept in the top left cell, and the other cells are filled with their contents before they get merged. They are also reapplied their original styling.
    * *Kopfzeile einschließen*: die Formatierung kann nur über das Widget :ref:`Schrift- und Textgestaltung ` gesteuert werden

- Select the cells and, on the right of the table frame, you can provide the *Cell Contents*:
  - Either manually enter the contents of each cell, or use an *Expression* to automatically populate it.
  - Under the *Formatting* group, set specific options that apply to the selection and take precedence over the *global table content styling*:
    * durch Auswahl des *Textformats*
    * durch *Als Zahl formatieren* (mehrere Formate sind verfügbar)
    * durch die Definition von *Horizontaler Ausrichtung* und *Vertikaler Ausrichtung*
    * durch Auswahl einer *Hintergrundfarbe*
  - Definieren Sie die *Zellengröße* mit *Zeilenhöhe* und *Spaltenbreite*.

Erscheinungsbild

Der Bereich *Erscheinungsbild* der festen Tabelle bietet die folgenden Funktionalitäten:

- Klicken Sie auf *Zeige leere Zeilen*, um die Attributtabelle mit leeren Zellen zu füllen.
- Mit *Zellenränder* können Sie die Ränder um den Text in jeder Zelle der Tabelle definieren.
- Mit *Kopf anzeigen* können Sie von einer Liste eine der voreingestellten Optionen ‚Im ersten Rahmen', ‚Auf allen Rahmen' oder ‚Kein Kopf' auswählen.
- Mit *Hintergrundfarbe* können Sie die Hintergrundfarbe der Tabelle mit dem *Farbauswahl* Werkzeug einstellen. Mit der Option *Fortgeschrittene Anpassungen…* können Sie verschiedene Hintergrundfarben für jede Zelle definieren.
- Mit *Übergroßer Text* definieren Sie das Verhalten, wenn die für eine Spalte eingestellte Breite kleiner ist als die Länge ihres Inhalts. Es kann **Text umbrechen** oder **Text abschneiden** sein.

Bemerkung

Weitere Eigenschaften des festen Tabellenelements sind im Abschnitt *Gemeinsame Funktionalitäten der Tabellen* beschrieben.

Gemeinsame Funktionalitäten der Tabellen

Gitter anzeigen

Die Gruppe *Gitter anzeigen* der Tabellenelemente bietet die folgenden Funktionen (siehe Abb. 18.49):

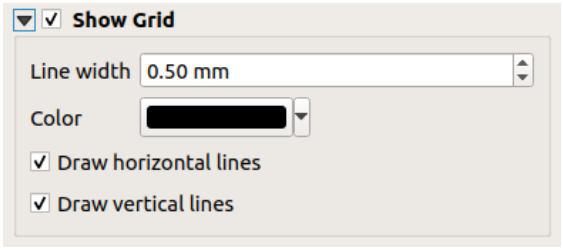


Abb. 18.49: Der Bereich Gitter anzeigen von Tabellen-Elementen

- Aktivieren Sie das Kontrollkästchen *Gitter anzeigen*, wenn Sie das Gitter, also die Umrisse der Tabellenzellen, anzeigen wollen. Sie können *Horizontale Linien anzeigen* oder *Vertikale Linien anzeigen* einzeln oder zusammen aktivieren.
- Mit *Linienstärke* können Sie die Dicke der Linien, die im Gitter verwendet werden, festlegen.
- Die :guilabel:`Farbe‘ des Gitters kann mit Hilfe des Farbauswahl-Werkzeugs eingestellt werden.

Schrift- und Textgestaltung

Der Bereich *Schrift- und Textgestaltung* eines Tabellen-Elements bietet die folgenden Funktionen (siehe Abb. 18.50):

Abb. 18.50: Der Bereich Schrift- und Textgestaltung von Tabellen-Elementen

- Sie können *Schriftart*-Eigenschaften für *Tabellenkopf* und *Tabelleninhalt* definieren, unter Verwendung des erweiterten *Texteinstellungen*-Werkzeug (mit Puffer, Schatten, Farbeffekten, Transparenz, Hintergrund, Farbgebung, …). Beachten Sie, dass diese Änderungen keine Auswirkungen auf die Zellen haben, denen eine benutzerdefinierte Schriftart zugewiesen wurde, entweder aus dem Abschnitt *Erscheinungsbild* oder aus dem Dialog *Tabellenentwurf*. Es werden nur Zellen mit der Standarddarstellung überschrieben.
- For *Table heading* you can additionally set the *Alignment* to `Follow column alignment` or override this setting by choosing `Left`, `Center` or `Right`. The column alignment is set using the *Select Attributes* dialog (see Abb. 18.44).

Rahmen

Der Bereich *Rahmen* der Tabellenelementeigenschaften bietet die folgenden Funktionen (siehe Abb. 18.51):

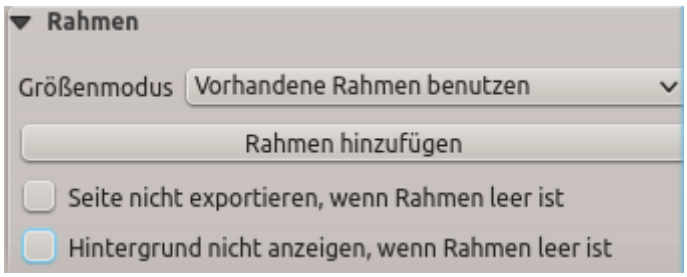


Abb. 18.51: Der Bereich Rahmen von Tabellen-Elementen

- Mit dem *Größenmodus* können Sie auswählen, wie der Inhalt der Attributtabelle gerendert wird:
  - `Vorhandene Rahmen verwenden` zeigt das Ergebnis nur im ersten Rahmen und den hinzugefügten Rahmen an.
  - `Auf nächste Seite erweitern` erstellt so viele Rahmen (und entsprechende Seiten), wie nötig sind, um die gesamte Auswahl der Attributtabelle anzuzeigen. Jeder Rahmen kann im Layout verschoben werden. Wenn Sie die Größe eines Rahmens ändern, wird die resultierende Tabelle auf die anderen Rahmen aufgeteilt. Der letzte Rahmen wird so zugeschnitten, dass er in die Tabelle passt.
  - `Bis zum Ende wiederholen` erzeugt ebenso viele Rahmen wie die Option „Auf nächste Seite erweitern“, allerdings haben alle Rahmen die gleiche Größe.
- Verwenden Sie die Schaltfläche *Rahmen hinzufügen*, um einen weiteren Rahmen mit der gleichen Größe wie der ausgewählte Rahmen hinzuzufügen. Das Ergebnis der Tabelle, das nicht in den ersten Rahmen passt, wird im nächsten Rahmen fortgesetzt, wenn Sie den Modus „Größe ändern“ „Vorhandene Rahmen verwenden“ verwenden.
- Aktivieren Sie *Seite nicht exportieren, wenn der Rahmen leer ist* verhindert, dass die Seite exportiert wird, wenn der Tabellenrahmen keinen Inhalt hat. Das bedeutet, dass alle anderen Layoutelemente, Karten, Skalenleisten, Legenden usw. im Ergebnis nicht sichtbar sind.
- Das Aktivieren von *Hintergrund nicht anzeigen, wenn Rahmen leer ist* verhindert, dass der Hintergrund gezeichnet wird, wenn der Rahmen keinen Inhalt hat.

18.2.8 Die Elemente Markierung, Bild und Nordpfeil

- *Das Bild-Element*
  - *Haupteigenschaften*
  - *Größe und Platzierung*
  - *Bilddrehung*
- *Das Nordpfeil-Element*
- *Das Markierungs-Element*

Neben den Karten- oder Legenden-Elementen im Drucklayout können Sie Ihre Umsetzung mit Bildern oder Anmerkungen dekorieren. QGIS bietet verschiedene Werkzeuge, um dies zu erreichen:

- das *Bild-Element*: fügt dem Layout ein Bildraster oder einer SVG-Datei (z.B. Logos, Bilder, …) hinzu

- das *Nordpfeil-Element* ist ein Bild-Element, das mit einem Nordpfeil vordefiniert ist
- das *Markierungs-Element* fügt dem Layout ein QGIS *Markierungs-Symbol* hinzu. Es kann z.B. verwendet werden, um Markierungen über einem Kartenelement zu platzieren oder um erweiterte benutzerdefinierte Legenden zu erstellen.

Das Bild-Element

Sie können ein Bild hinzufügen, indem Sie es aus Ihrem Dateimanager auf die Kartenansicht ziehen, es mit `Strg+V` oder *Bearbeiten* ► *Einfügen* direkt in das Layout einfügen und die Funktion Bild hinzufügen verwenden, wie in der *Anleitung zum Erstellung von Elementen*. Das Element kann dann bearbeitet werden, wie in *Interaktion mit Layout-Elementen* erklärt.

Wenn Sie Bild hinzufügen verwenden, wird das Bildelement ein leerer Rahmen sein, den Sie über das Bedienfeld *Elementeigenschaften* anpassen können. Neben den *gemeinsamen Elementeigenschaften* bietet dieses Element die folgenden Funktionalitäten:

Haupteigenschaften

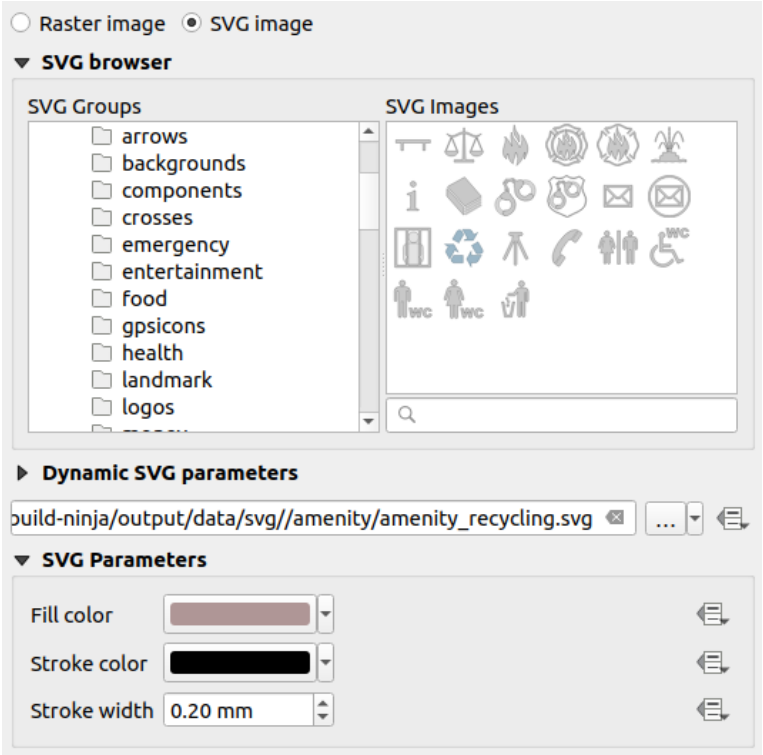


Abb. 18.52: Elementeigenschaften für Bilder

Das Element Bild unterstützt zwei Arten von Bildern:

- *Rasterbild*: Ein Dateiauswahl-Widget kann verwendet werden, um die Daten zu laden. Verwenden Sie die Schaltfläche … Durchsuchen, um eine Datei auf Ihrem Computer auszuwählen oder geben Sie den Pfad direkt in das Textfeld ein. Sie können auch eine URL angeben, die auf ein Bild verweist. Das zugehörige Bild kann auch als *einbettete Datei* in das Layout eingefügt werden.

  Benutzen Sie die Schaltfläche Datendefinierte Übersteuerung, um die Bildquelle aus einem Objektattribut zu lesen oder über einen Ausdruck einzustellen.

- *SVG Bild*: verwendet standardmäßig die SVG-Bibliotheken, die unter *Einstellungen* ► *Optionen* ► *System* ► *SVG Pfade* zur Verfügung stehen. Sie können jedoch jede andere Datei verwenden, und die Dateiauswahl folgt den gleichen Regeln wie für das Rasterbild. Die SVG-Parameter können auch dynamisch gesetzt werden.

  Die von QGIS bereitgestellten (Standard-) `.SVG`-Dateien sind anpassbar, d.h. Sie können ganz einfach andere *Füllfarbe*, *Strichfarbe* (einschließlich Deckkraft) und *Strichbreite* als das Original anwenden, indem Sie das entsprechende Objekt in der Gruppe *SVG-Parameter* verwenden. Diese Eigenschaften können auch *data-defined* sein.

  Wenn Sie eine :file:`.SVG“-Datei hinzufügen, in der diese Eigenschaften nicht aktiviert sind, müssen Sie möglicherweise die folgenden Tags in die Datei einfügen, um z. B. Transparenz zu unterstützen:

  – *fill-opacity=“param(fill-opacity)“*

  – *stroke-opacity=“param(outline-opacity)“*

  Weitere Details unter *Parametrisierbare SVG*.

Bemerkung

Durch Ziehen und Ablegen einer Bilddatei (Raster oder SVG) auf die Layoutseite wird ein Layoutbildelement mit den entsprechenden Einstellungen erstellt.

Größe und Platzierung

Abb. 18.53: Layoutbilder Größe und Platzierungseigenschaften

Mit der Einstellung *Größenmodus* können Sie festlegen, wie das Bild dargestellt wird, wenn sich die Größe des Rahmens ändert:

- Zoom“: Vergrößert/verkleinert das Bild auf den Rahmen unter Beibehaltung des Seitenverhältnisses des Bildes
- `Strecken`: ändert die Seitenverhältnisse des Bildes, damit es in den Rahmen passt
- `Zuschneiden`: Verwenden Sie diesen Modus nur für Rasterbilder. Er setzt die Größe des Bildes auf die ursprüngliche Bildgröße, ohne Skalierung, und der Rahmen wird zum Beschneiden des Bildes verwendet. So wird nur der Teil des Bildes sichtbar, der sich innerhalb des Rahmens befindet.
- `Rahmen zoomen und Größe anpassen`: vergrößert das Bild, damit es in den Rahmen passt, und passt dann die Größe des Rahmens an die resultierenden Bildabmessungen an
- `Rahmen auf Bildgröße setzen`: setzt die Größe des Rahmens auf die Originalgröße des Bildes (keine Skalierung)

Je nach ausgewähltem *Größenmodus* können die Optionen *Platzierung* und *Bilddrehung* deaktiviert sein. Mit *Platzierung* können Sie die Position des Bildes innerhalb seines Rahmens auswählen (oben/mittig/unten und links/mittig/rechts).

Bilddrehung

Bilder können mit dem Feld *Bilddrehung* gedreht werden. Durch Aktivieren des Kontrollkästchens *Mit Karte abgleichen* wird die Drehung des Bildes mit der Drehung des ausgewählten Kartenelements synchronisiert. Dies ist ein praktisches Objekt, um ein beliebiges Bild als Nordpfeil zu verwenden. Die *Nordausrichtung* kann sein:

- **Gitternorden**: die Richtung einer Gitterlinie, die parallel zum Zentralmeridian des nationalen/lokalen Gitters verläuft
- **Geographisch Norden**: Richtung eines Längenmeridians.

Mit *Versatz* können Sie der Bilddrehung auch eine Abweichung hinzufügen.

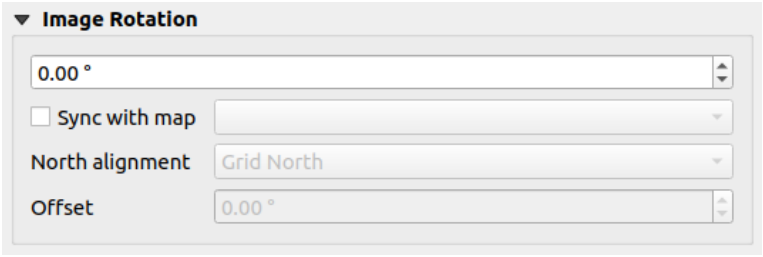


Abb. 18.54: Bildrotationseigenschaften eines Bild-Elements

Das Nordpfeil-Element

Sie können einen Nordpfeil mit der Schaltfläche Nordpfeil hinzufügen hinzufügen, indem Sie der *Anleitung zur Erstellung von Elementen* folgen und ihn dann, wie in *Interaktion mit Layout-Elementen* beschrieben, bearbeiten.

Da Nordpfeile Bilder sind, hat das Element *Nordpfeil* grundsätzlich die gleichen Eigenschaften wie das Element *Bild*. Die wichtigsten Unterschiede sind:

- Beim Hinzufügen des Elements wird ein Standard-Nordpfeil anstelle eines leeren Rahmens verwendet.
- Das Element Nordpfeil wird standardmäßig mit einem Karten-Element verbunden: Die Eigenschaft *Mit Karte abgleichen* ist die Karte, über der das Element Nordpfeil gezeichnet wurde. Wenn es nicht über einem Karten-Element gezeichnet wurde, wird statdessen die *Referenzkarte* verwendet.

Bemerkung

Viele der Nordpfeile haben kein „N“ im Nordpfeil hinzugefügt. Dies geschieht absichtlich, da es Sprachen gibt, die kein „N“ für Norden verwenden.

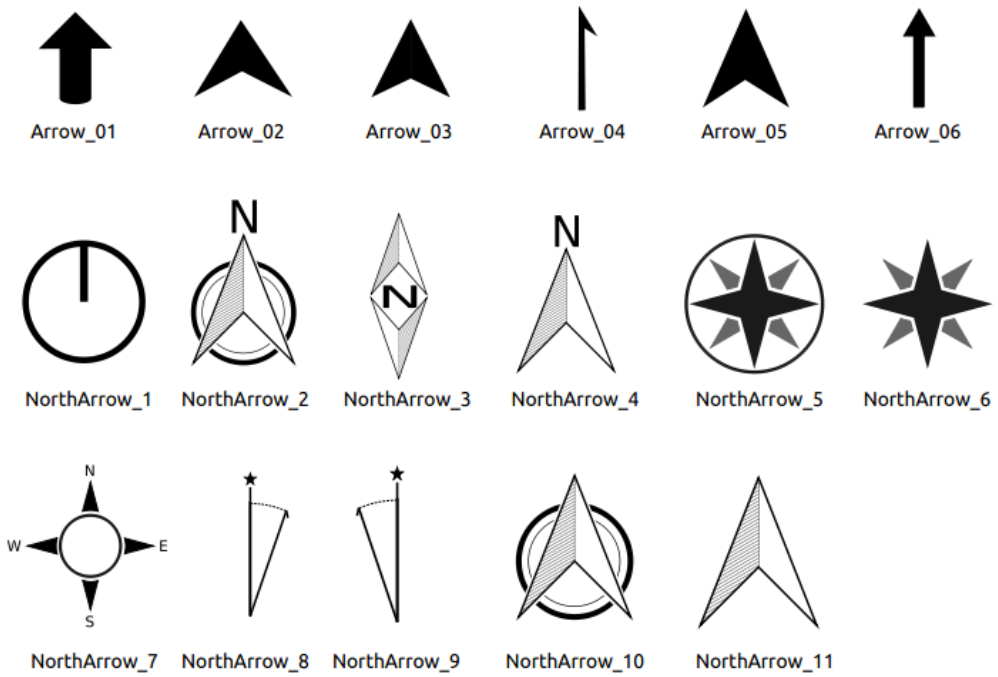


Abb. 18.55: Nordpfeile, die von der SVG-Bibliothek zur Verfügung gestellt werden.

Das Markierungs-Element

Um ein Markierungs-Element hinzuzufügen, wählen Sie die Schaltfläche Markierung hinzufügen und klicken Sie auf die Seite. Ein Standard-Punktmarkierungssymbol wird hinzugefügt. Dann können Sie es verändern, wie in *Interaktion mit Layout-Elementen* erklärt. Beachten Sie jedoch, dass Sie im Gegensatz zu den meisten anderen Elementen die Größe des Elements ändern können, da seine Größe durch die Eigenschaften der eingebetteten Symbole gesteuert wird.

Das Markierungselement kann über das Bedienfeld *Elementeigenschaften* angepasst werden. Neben den *gemeinsamen Elementeigenschaften*, können Sie auch:

- das *Symbol* ändern, unter der Verwendung der *Symbolauswahl*
- die Drehung des Markierungselements mit der Karte synchronisieren (siehe *Bilddrehung*) um z.B. als Nordpfeil zu fungieren. Die Drehung der Karte wird zu jeder bestehenden Drehung der Markierungssymbolebene hinzugefügt (wenn Sie also z.B. die Dreiecksmarkierung um 90° drehen müssen, damit sie gerade nach oben zeigt, funktioniert sie immer noch gut im Nordpfeil-Modus).

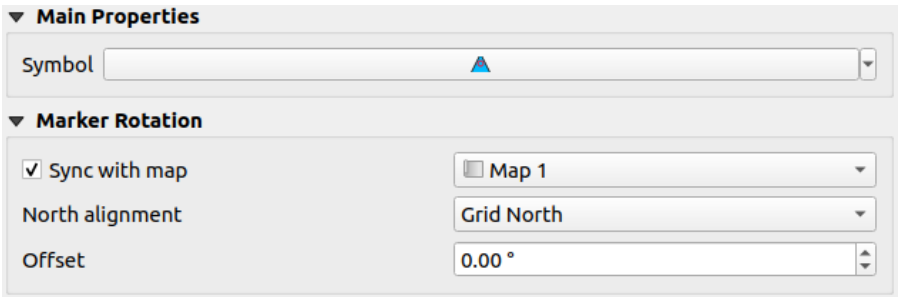


Abb. 18.56: Die speziellen Eigenschaften eines Markierungs-Elements

18.2.9 Das Geländehöhenprofil-Element

Das Element Geländehöhenprofil wird verwendet, um eine *Höhenprofilansicht* in einem Layout anzuzeigen. Verwenden Sie die Schaltfläche *Geländehöhenprofil hinzufügen* und folgen Sie der Anleitung zum Erstellen von LAyout-Elementen, um ein neues Höhenprofil-Element hinzuzufügen. Dies können Sie dann, wie in *Interaktion mit Layout-Elementen* gezeigt, anpassen.

Ein neues Geländehöhenprofil hat Standardeinstellungen, die ein leeres Diagramm erzeugen. Sie können die Eigenschaften im Bedienfeld *Elementeigenschaften* anpassen. Zusätzlich zu den *gemeinsamen Elementeigenschaften* verfügt dieses Element über die folgenden Funktionalitäten:

Das Bedienfeld *Elementeigenschaften* enthält eine obere Symbolleiste mit den folgenden Funktionen:

- Geländehöhenprofil aktualisieren um die Darstellung des Elements zu aktualisieren
- Aus Profil kopieren: ein Dropdown-Menü, aus dem Sie eine Höhenprofilansicht auswählen können. Die Ansichtseinstellungen werden auf das Layout-Höhenprofil-Element angewendet und können später geändert werden.

Layer

Markieren Sie unter der Gruppe *Layer* in der Strukturansicht die Layer, die Sie im Profilelement darstellen möchten. Denken Sie daran, die Eigenschaften *Höhe* der ausgewählten Layer richtig zu konfigurieren.

Profilkurve

- *Gesteuert durch Atlas*: die *Profilkurve* wird aus dem aktuellen Atlas-Objekt übernommen und die Ansicht des Höhenprofils aktualisiert, während Sie durch die Atlas-Objekte gehen. Dies wird derzeit für einen aktiven Layout-Atlas oder -Bericht unterstützt, der einen Coverage-Layer vom Typ Liniengeometrie verwendet.
- Der *Toleranz*-Abstand, der datendefiniert übersteuert werden kann, hilft Ihnen zu kontrollieren, wie weit ein Objekt der sichtbaren Layer von der Profilkurve entfernt sein soll, um in der Layout-Höhenansicht angezeigt zu werden. Derzeit werden nur punktförmige Objekte zurückgegeben.

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.

Diagrammbereich

Ein Layout-Höhenprofil-Element zeigt nicht notwendigerweise das gesamte Ausmaß der Höhenprofilansicht an, auf der es basiert. Sie können den zu rendernden Bereich einschränken, wenn Sie dies wünschen:

- auf der X-Achse den *Minimalabstand* und *Maximalabstand* vom Startpunkt der Profilkurve
- Auf der Y-Achse die Werte *Minimale Geländehöhe* und *Maximale Geländehöhe*

Distanzachse und Geländehöhenachse

Die Gruppen *Distanzachse* und *Geländehöhenachse* bieten Optionen, um das Raster über dem Höhenprofil auf der X- bzw. Y-Achse zu optimieren:

- die Entfernungsanzeige *Einheit*, mit der die Einheit der Kartenansicht überschrieben werden kann
- die Teilung auf der Achse mit einem *Hauptintervall* und *Unterintervall*
- die Liniensymbole, die auf die entsprechenden *Hauptgitterlinien* und *Untergitterlinien* anzuwenden sind
- wie regelmäßig die Teilungselemente beschriftet werden sollen (*Beschriftungsintervall*) sowie deren *Beschriftungsformat* und *Beschriftungsschrift*
- die *Abstandsbeschriftungen*: legt fest, ob das Symbol für die Abstandseinheit neben *Jeder Wert*, *Erster Wert*, *Letzter Wert*, *Erster und letzter Wert* oder gar nicht (*Keine*) platziert werden soll

Diagrammbereich

Unter *Diagrammbereich* können Sie die Darstellung des Bereichs konfigurieren, in dem das Geländehöhenprofil angezeigt wird:

- ein *Hintergrund* Füllsymbol
- ein *Rand* Liniensymbol
- die Ränder vom Rand des Geländehöhenprofils

18.2.10 Das HTML-Rahmen Element

Es ist möglich, einen Rahmen hinzuzufügen, der den Inhalt einer Website anzeigt, oder sogar eine eigene HTML-Seite zu erstellen und zu gestalten und sie anzuzeigen. Sie können ein HTML-Element mit dem Befehl *HTML hinzufügen*, wie in der Anleitung zum Erstellen von Elementen <create_layout_item>`erklärt, hinzufügen und es dann, wie in :ref:`interact_layout_item erklärt, bearbeiten. Beachten Sie, dass die HTML-Skalierung durch die Auflösung des Layout-Exports zum Zeitpunkt der Erstellung des HTML-Rahmens gesteuert wird.

Das HTML-Element kann über das Bedienfeld *Elementeigenschaften* angepasst werden. Abgesehen von den *gemeinsamen Eigenschaften*, hat dieses Element die folgenden Funktionen (siehe Abb. 18.57):

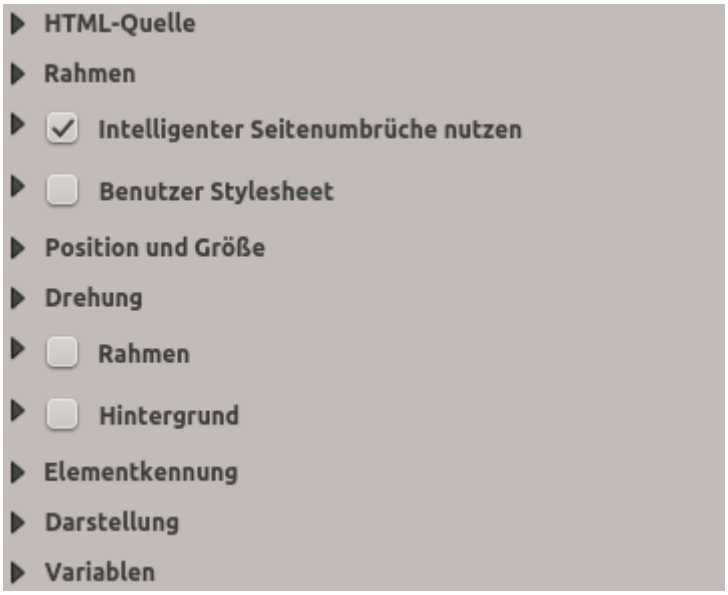


Abb. 18.57: Übersicht über die Elementeigenschaften eines HTML-Elements

HTML-Quelle

Die Gruppe *HTML-Quelle* des Bedienfeldes *Elementeigenschaften* eines HTML-Rahmens bietet die folgenden Funktionen (siehe Abb. 18.58):

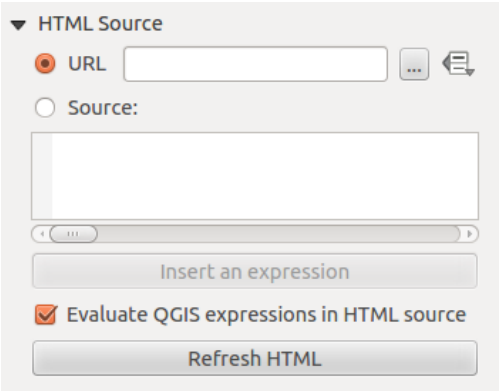


Abb. 18.58: HTML-Rahmen, die Eigenschaften der HTML-Quelle

- In *URL* können Sie die URL einer Webseite eingeben (z.B. indem Sie sie von Ihrem Internet-Browser kopieren) oder eine HTML-Datei mit der Schaltfläche … Durchsuchen auswählen. Es besteht auch die Möglichkeit, die Schaltfläche Datendefinierte Übersteuerung zu verwenden, um eine URL aus dem Inhalt eines Attributfeldes einer Tabelle oder mithilfe eines regulären Ausdrucks anzugeben.
- Unter *Quelle* können Sie einen Text mit HTML-Formatierungen direkt ins Textfenster eingeben oder eine ganze HTML-Seite zur Verfügung zu stellen.
- Die Schaltfläche *Einen Ausdruck einfügen oder bearbeiten…* kann verwendet werden, um einen Ausdruck wie `[%Year($now)%]` in das Textfeld Quelle einzufügen, um das aktuelle Jahr anzuzeigen. Diese Schaltfläche wird nur aktiviert, wenn der Auswahlknopf *Quelle* ausgewählt ist. Klicken Sie nach dem Einfügen des Ausdrucks irgendwo in das Textfeld, bevor Sie den HTML-Rahmen aktualisieren, sonst geht der Ausdruck verloren.
- Aktivieren Sie *QGIS-Ausdrücke in HTML-Quelle auswerten*, um das Ergebnis des Ausdrucks, den Sie eingefügt haben, zu sehen, andernfalls sehen Sie den Ausdruck.

- Verwenden Sie die Schaltfläche *HTML aktualisieren*, um den/die HTML-Frame(s) zu aktualisieren und das Ergebnis der Änderungen zu sehen.

Rahmen

Die Gruppe *Rahmen* im Bedienfeld HTML-Rahmen *Elementeigenschaften* bietet folgende Funktionen (siehe Abb. 18.59):

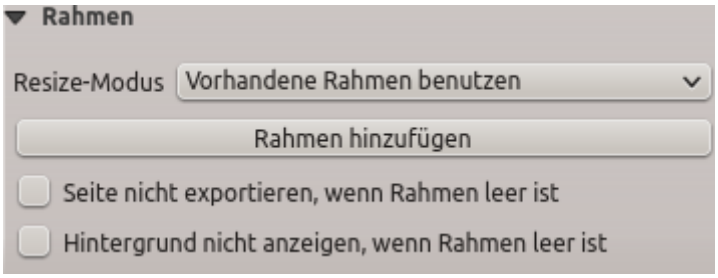


Abb. 18.59: HTML-Rahmen, die Eigenschaften des Rahmen

- Mit dem *Größenmodus* können Sie auswählen, wie der HTML Inhalt gerendert wird:
  - `Vorhandene Rahmen verwenden` zeigt das Ergebnis nur im ersten Rahmen und den hinzugefügten Rahmen an.
  - `Auf die nächste Seite erweitern` erzeugt so viele Rahmen (und entsprechende Seiten), wie nötig sind, um die Höhe der Webseite darzustellen. Jeder Rahmen kann auf dem Layout verschoben werden. Wenn Sie die Größe eines Rahmens ändern, wird die Webseite auf die anderen Rahmen aufgeteilt. Der letzte Rahmen wird so zugeschnitten, dass die Webseite passt.
  - `Auf jeder Seite wiederholen` wiederholt die obere linke Seite der Webseite auf jeder Seite in Rahmen der gleichen Größe.
  - Die Option `Bis zum Ende wiederholen` erzeugt ebenso viele Rahmen wie die Option `Bis zur nächsten Seite erweitern`, nur daß alle Rahmen die gleiche Größe haben.
- Verwenden Sie die Schaltfläche *Rahmen hinzufügen*, um einen weiteren Rahmen mit der gleichen Größe wie der ausgewählte Rahmen hinzuzufügen. Wenn die HTML-Seite nicht in den ersten Rahmen passt, wird sie im nächsten Rahmen fortgesetzt, wenn Sie *Größenmodus* oder *Vorhandene Rahmen verwenden* auswählen.
- Aktivieren Sie ☑ *Seite nicht exportieren, wenn der Rahmen leer ist* verhindert, dass die Seite exportiert wird, wenn der Rahmen keinen HTML-Inhalt hat. Das bedeutet, dass alle anderen Layoutelemente, Karten, Maßstabsleisten, Legenden usw. im Ergebnis nicht sichtbar sind.
- Das Aktivieren von ☑ *Hintergrund nicht anzeigen, wenn Rahmen leer ist* verhindert, dass der HTML-Rahmen gezeichnet wird, wenn der Rahmen leer ist.

Intelligente Seitenumbrüche und Benutzer-Stylesheet nutzen

Die Bereiche *Intelligente Seitenumbrüche nutzen* und *Benutzer Stylesheet* eines HTML-Elements bieten die folgenden Funktionalitäten (siehe Abb. 18.60):

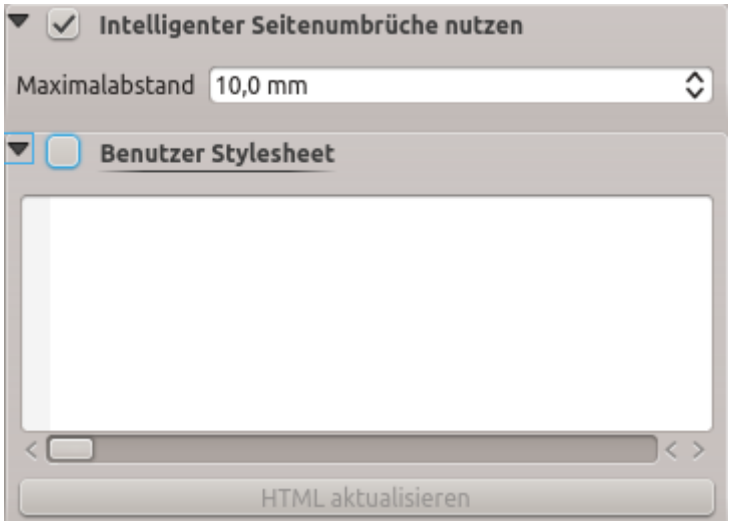


Abb. 18.60: HTML-Rahmen, Verwendung von intelligenten Seitenumbrüchen und Benutzer-Stylesheet-Eigenschaften

- Aktivieren Sie *Intelligente Seitenumbrüche nutzen*, um zu verhindern, dass der HTML-Rahmen inmitten einer Textzeile umbricht, sodass er weiter schön und glatt im nächsten Rahmen weitergeht.
- Setzen Sie den erlaubten *Maximalabstand* wenn ausgerechnet werden soll wo die Seitenumbrüche im HTML platziert werden sollen. Dieser Abstand ist der maximale Größe des leeren Raums am Fuß eines Rahmens nachdem der optimale Platz für den Seitenumbruch berechnet wurde. Das Setzen eines größeren Wertes resultiert in einer besseren Auswahl des Platzes für den Seitenumbruch, es resultiert aber auch in mehr nicht genutztem Platz am Fuß des Rahmens. Dies wird nur genutzt wenn *Intelligenter Seitenumbrüche nutzen* aktiviert ist.
- Aktivieren Sie *Benutzer Stylesheet*, um HTML-Stile anzuwenden, die oft in kaskadierenden Stilvorlagen bereitgestellt werden. Nachfolgend ein Beispiel für Style Code, um die Farbe des `<h1>` Header-Tags auf grün zu setzen und die Schriftart und -größe des Textes in den Absatz-Tags `<p>` einzustellen.

```
h1 {color: #00ff00;
}
p {font-family: "Times New Roman", Times, serif;
   font-size: 20px;
}
```

- Verwenden Sie die Schaltfläche *HTML aktualisieren*, um das Ergebnis der Stylesheet-Einstellungen zu sehen.

18.2.11 Das Form-Element

QGIS bietet eine Reihe von Werkzeugen, um regelmäßige oder komplexere Formen über das Drucklayout zu zeichnen.

Bemerkung

Im Gegensatz zu anderen Drucklayoutelementen können Sie weder den Rahmen noch die Hintergrundfarbe des Begrenzungsrahmens einer Form gestalten (der standardmäßig auf transparent eingestellt ist).

Das Rechteck-Element

Das *Form* Element ist ein Werkzeug, das Ihnen hilft, Ihre Karte mit regelmäßigen Formen wie Dreieck, Rechteck, Ellipse… zu dekorieren. Sie können eine regelmäßige Form mit dem Werkzeug Form hinzufügen hinzufügen, das Zugriff auf die Optionen Rechteck hinzufügen, Ellipse hinzufügen und Dreieck hinzufügen bietet. Sobald Sie das entsprechende Werkzeug ausgewählt haben, können Sie das Element gemäß der *Anleitung zum Erstellen von Layput-Elementen* zeichnen. Wie andere Layoutelemente kann eine regelmäßige Form auf die gleiche Weise bearbeitet werden, wie in *Interaktion mit Layout-Elementen* beschrieben.

Bemerkung

Wenn Sie die Taste `Umschalt` gedrückt halten, während Sie die Grundform mit der Klick- und Ziehmethode zeichnen, können Sie ein perfektes Quadrat, einen Kreis oder ein Dreieck erstellen.

Das Standardformelement kann über das Bedienfeld *Elementeigenschaften* angepasst werden. Abgesehen von den *gemeinsamen Elementeigenschaften*, hat dieses Objekt die folgenden Funktionen (siehe Abb. 18.61):

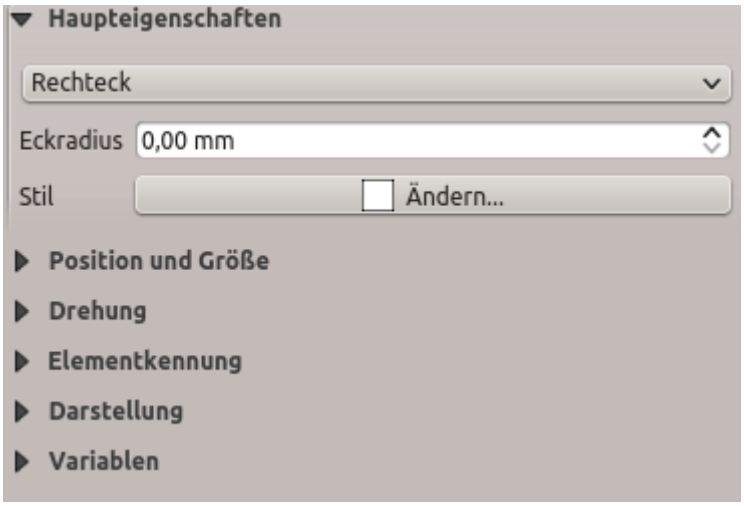


Abb. 18.61: Bedienfeld für die Eigenschaften von Formelementen

Die Gruppe *Haupteigenschaften* zeigt den Typ des Form-Elements (**Ellipse**, **Rechteck** oder **Dreieck**) innerhalb des gegebenen Rahmens an und erlaubt es, diesen zu wechseln.

Sie können den Stil der Form mit den erweiterten Werkzeugen für die *Symbolauswahl* und die *Farbe* festlegen.

Für die Rechteckform können Sie den Wert des *Eckradius* in verschiedenen Einheiten einstellen, um die Ecken abzurunden.

Das Knoten-Element

Während das Werkzeug *Form hinzufügen* die Möglichkeit bietet, einfache und vordefinierte geometrische Elemente zu erstellen, hilft Ihnen das Werkzeug *Knotenelement hinzufügen* bei der Erstellung eines benutzerdefinierten und erweiterten geometrischen Elements. Für Polylinien oder Polygone können Sie so viele Linien oder Seiten zeichnen, wie Sie möchten, und die Eckpunkte der Elemente können unabhängig und direkt mit dem Werkzeug *Knotenelement bearbeiten* manipuliert werden. Das Element selbst kann wie in *Interaktion mit Layout-Elementen* dargestellt bearbeitet werden.

So fügen Sie ein Knoten-Element hinzu:

1. Klicken Sie auf das Symbol Knotenelement hinzufügen.
2. Wählen Sie entweder das Werkzeug Polygon hinzufügen oder Polyline hinzufügen.
3. Führen Sie aufeinanderfolgende Linksklicks aus, um Knoten Ihres Elements hinzuzufügen. Wenn Sie beim Zeichnen eines Segments die `Umschalt`-Taste gedrückt halten, wird das Segment auf ein Vielfaches von 45° ausgerichtet.
4. Wenn Sie fertig sind, klicken Sie mit der rechten Maustaste, um die Form zu beenden.

Sie können das Aussehen der Form im Bedienfeld *Elementeigenschaften* anpassen.

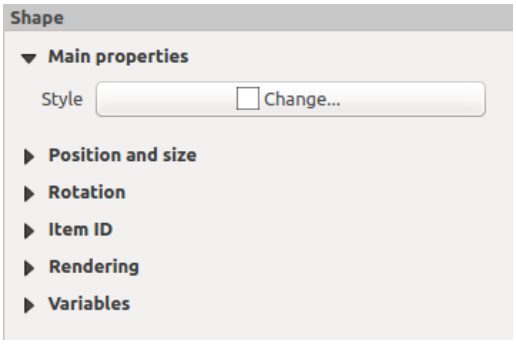


Abb. 18.62: Bedienfeld für Polygonknotenform-Eigenschaften

In den *Haupteigenschaften* können Sie den Stil der Form mit den erweiterten Werkzeugen für die *Symbolauswahl* und die *Farbe* festlegen.

Für Polylinien-Knotenpunkte können Sie auch das *Line markers* parametrisieren, d.h. hinzufügen:

- Start- und/oder Endmarkierungen mit Optionen:
  - *Keine*: zeichnet eine einfache Polylinie.
  - *Pfeil*: fügt eine reguläre dreieckige Pfeilspitze hinzu, die Sie individuell anpassen können.
  - *SVG* Markierung : verwendet eine `SVG`-Datei als Pfeilspitze des Elements.
- passen Sie die Pfeilspitze an:
  - *Pfeilstrichfarbe*: legt die Strichfarbe der Pfeilspitzenumrandung fest.
  - *Pfeilfüllfarbe*: legt die Füllfarbe der Pfeilspitze fest.
  - *Pfeilstrichstärke*: legt die Strichstärke der Pfeilspitze fest.
  - *Pfeilspitzenbreite*: legt die Größe der Pfeilspitze fest.

SVG-Bilder werden automatisch mit der Linie gedreht. Die Strich- und Füllfarben von QGIS vordefinierten SVG-Bildern können mit den entsprechenden Optionen geändert werden. Benutzerdefiniertes SVG kann einige Tags erfordern, die dieser *Anweisung* folgen.

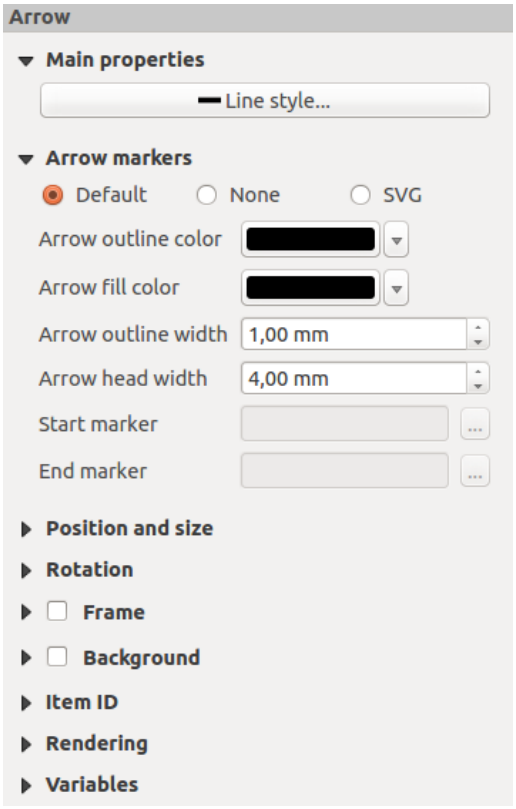


Abb. 18.63: Bedienfeld für die Eigenschaften von Polylinienknotenformen

Das Pfeilelement

Das Werkzeug Pfeil hinzufügen ist eine Abkürzung, um eine mit Pfeilen versehene Polylinie standardmäßig zu erstellen und hat daher die gleichen Eigenschaften und das gleiche Verhalten wie ein *Polylinienknotenpunkt*.

Das Pfeilelement kann verwendet werden, um einen einfachen Pfeil hinzuzufügen, z. B. um die Beziehung zwischen zwei verschiedenen Drucklayoutelementen zu zeigen. Um einen Nordpfeil zu erstellen, sollte jedoch zuerst das Element *Bild* in Betracht gezogen werden, da es Zugang zu einer Reihe von Nordpfeilen im Format `.SVG` bietet, die Sie mit einem Kartenelement synchronisieren können, so dass sie sich automatisch mit diesem drehen.

Bearbeiten der Geometrie eines Knoten-Elements

Mit Knotenelement bearbeiten wird ein spezielles Werkzeug zur Bearbeitung von knotenbasierten Formen bereitgestellt. In diesem Modus können Sie einen Knoten auswählen, indem Sie auf ihn klicken (eine Markierung wird auf dem ausgewählten Knoten angezeigt). Ein ausgewählter Knoten kann entweder durch Ziehen oder mit den Pfeiltasten verschoben werden. Außerdem können Sie in diesem Modus Knoten zu einer bestehenden Form hinzufügen: Doppelklicken Sie auf ein Segment, und ein Knoten wird an der Stelle hinzugefügt, auf die Sie klicken. Schließlich können Sie den aktuell ausgewählten Knoten durch Drücken der Taste `Entf` entfernen.

18.3 Eine Ausgabe erzeugen

Abb. 18.64 zeigt ein Beispiel für ein Drucklayout, das alle in den vorigen Abschnitten beschriebenen Arten von Layout-Elementen enthält.

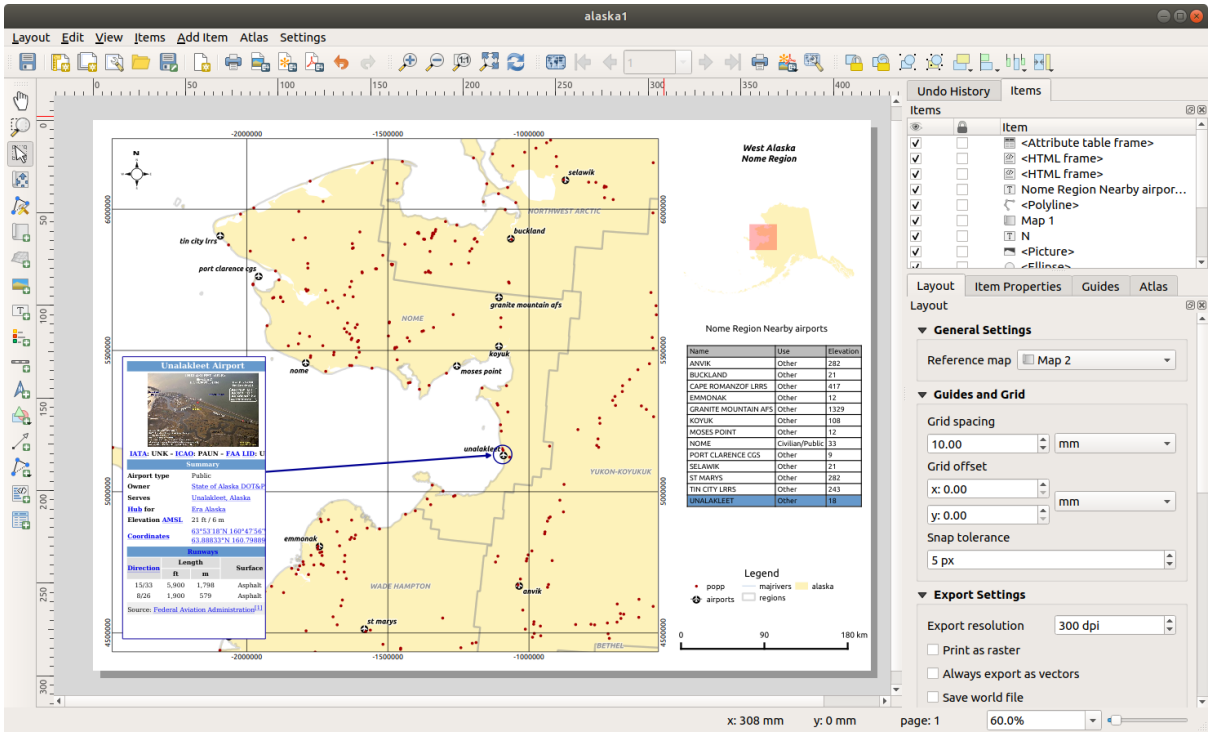


Abb. 18.64: Ein Drucklayout mit verschiedenen Layout-Elementen

Über das Menü *Layout* oder die Layout-Werkzeugleiste können Sie das Drucklayout in verschiedenen Dateiformaten ausgeben, wobei es möglich ist, verschiedene Einstellungen vorzunehmen (s.u.):

- Mit dem Symbol Drucklayout können Sie das Layout auf einem angeschlossenen Drucker oder in einer PostScript-Datei ausgeben, je nach den auf Ihrem System installierten Druckertreibern.
- Das Symbol Als Bild exportieren exportiert in Drucklayout-Bildformate wie `PNG`, `BMP`, `TIF`, `JPG`, und viele andere…
- Das Symbol Als SVG exportieren speichert das Drucklayout als `SVG` (Scalable Vector Graphic).
- Das Symbol Als PDF exportieren speichert das Drucklayout direkt als `PDF` (Portable Document Format) Datei.

18.3.1 Exporteinstellungen

Wenn Sie ein Drucklayout exportieren, können Sie eine Reihe von Exporteinstellungen vornehmen, um die gewünschte Ausgabe zu erhalten. Diese Konfigurationen sind:

- Die *Exporteinstellungen* des *Layout* Bedienfeldes, wie *Export-Auflösung*, *Als Raster drucken*, *Immer als Vektoren exportieren* oder *World-Datei speichern*
- *Seite aus Export ausschließen* im Bedienfeld *Elementeigenschaften der Seite*
- *Element aus Export ausnehmen* in den *Elementeigenschaften der verschiedenen Elemente*

Darüber hinaus wird eine Reihe von vordefinierten Prüfungen automatisch auf das Layout angewendet. Derzeit wird u. a. geprüft, ob Maßstabsleisten korrekt mit Kartenelementen verknüpft sind und ob Kartenübersichtselemente ebenfalls

korrekt mit einer Karte verknüpft sind. Wenn die Prüfungen fehlschlagen, erhalten Sie eine Warnung, die Sie auf das Problem hinweist.

18.3.2 Exportieren als Rasterbild

So exportieren Sie ein Layout als Bild:

1. Klicken Sie auf das Symbol Als Bild exportieren.
2. Wählen Sie das zu verwendende Bildformat, den Ordner und den Dateinamen (z. B. `myill.png`). Enthält das Layout mehr als eine Seite, wird jede Seite in eine Datei mit dem angegebenen Dateinamen und der angehängten Seitenzahl exportiert (z. B. `myill_2.png`).
3. Im nächsten Dialog (*Bildexportoptionen*):
   - Sie können die Voreinstellungen für die *Export-Auflösung* und die exportierten Seitenabmessungen (aus dem Bedienfeld *Layout*) hier noch ändern.
   - For JPEG and JPG image formats, you can set the *JPEG Quality* of the output image.
   - Die Bildberechnung kann auch mit der Option *Anti-Aliasing aktivieren* verbessert werden.
   - Wenn Sie Ihr Layout als **georeferenziertes Bild** exportieren möchten (z.B. um es in einem anderen QGIS-Projekt zu verwenden), aktivieren Sie die Option *World-Datei erzeugen*, und eine *ESRI-World-Datei* mit dem gleichen Namen wie das exportierte Bild, aber einer anderen Erweiterung (`.tfw` für TIFF, `.pnw` für PNG, `jgw` für JPEG, …) wird beim Export erzeugt. Diese Option kann auch im *Layout Bedienfeld* standardmäßig aktiviert werden.

     > **Bemerkung**
     >
     > Bei mehrseitiger Ausgabe erhält nur die Seite, die die *Referenzkarte* enthält, eine World-Datei, wenn die Option *World-Datei erzeugen* aktiviert ist.

   - Wenn Sie die Option *Auf Inhalt zuschneiden* aktivieren, wird das vom Layout ausgegebene Bild den minimalen Bereich umfassen, der alle Elemente (Karte, Legende, Maßstabsleiste, Formen, Beschriftung, Bild…) jeder Seite des Drucklayouts umschließt:
     - Wenn das Drucklayout nur eine einzige Seite umfasst, wird die Größe der Ausgabe so angepasst, dass sie ALLE Elemente des Drucklayouts enthält. Die Seite kann dann je nach Position (auf, über, unter, links oder rechts von der Seite) auf alle Elemente reduziert oder erweitert werden.
     - Bei einem mehrseitigen Layout wird die Größe jeder Seite so angepasst, dass die Elemente in ihrem Bereich enthalten sind (linke und rechte Seite für alle Seiten, plus oben für die erste Seite und unten für die letzte Seite). Jede Seite mit geänderter Größe wird in eine separate Datei exportiert.

     Mit dem Dialogfeld *Auf Inhalt zuschneiden* können Sie auch Ränder um die zugeschnittenen Bereiche hinzufügen.
   - Wenn Sie das Kontrollkästchen *Datei nach dem Export öffnen* aktivieren, wird die exportierte Datei automatisch im Standard-Bildbetrachter geöffnet.

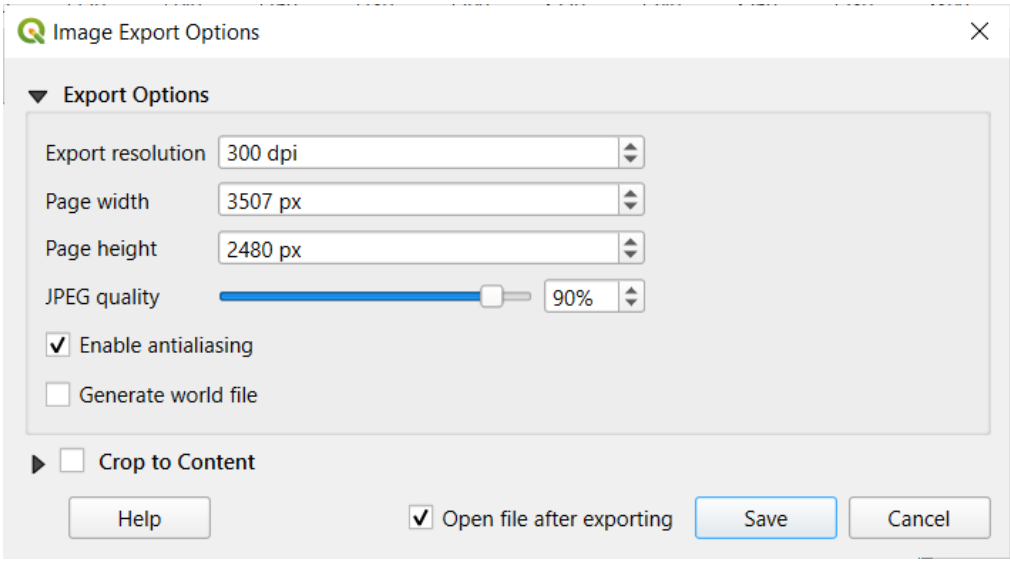


Abb. 18.65: Bildexportoptionen

Tipp

Verwenden Sie Bildformate, die Transparenz unterstützen, wenn Elemente über den Papierbereich hinausgehen

Layoutelemente können außerhalb des Papierbereichs platziert werden. Beim Export mit der Option *Beschneiden auf Inhalt* kann das resultierende Bild daher über die Papierausdehnung hinausreichen. Da der Hintergrund außerhalb der Papierausdehnung transparent ist, wird bei Bildformaten, die keine Transparenz unterstützen (z.B. `BMP` und `JPG`), der transparente Hintergrund als Schwarz wiedergegeben, was das Bild „verfälscht“. Verwenden Sie in solchen Fällen transparenzkompatible Formate (z.B. `TIFF` und `PNG`).

Bemerkung

Wenn das Format (z.B. `PNG`) und die zugrunde liegende Qt-Bibliothek dies unterstützen, kann das exportierte Bild *Projekt-Metadaten* (Autor, Titel, Datum, Beschreibung…) enthalten.

18.3.3 Als SVG exportieren

So exportieren Sie ein Layout als SVG:

1. Klicken Sie auf das Symbol Als SVG exportieren.
2. Geben Sie den Pfad und den Dateinamen ein (der als Basisname für alle Dateien im Falle einer mehrseitigen Zusammenstellung verwendet wird, wie beim Bildexport)
3. Im nächsten Dialogfeld *SVG-Exportoptionen* können Sie die Voreinstellungen aus den *Exporteinstellungen* des Layouts überschreiben oder neue konfigurieren:
   - *Kartenlayer als SVG-Gruppen exportieren*: exportierte Elemente werden in Layern gruppiert, deren Namen mit den Layernamen aus QGIS übereinstimmen, was es viel einfacher macht, den Inhalt des Dokuments zu verstehen.
   - *Immer als Vektor exportieren*: einige Rendering-Optionen verlangen, dass Objekte gerastert werden, um eine bessere Darstellung zu erhalten. Aktivieren Sie diese Option, um die Objekte als Vektoren

beizubehalten, mit dem Risiko, dass das Aussehen der Ausgabedatei nicht mit der Druckvorschau des Layouts übereinstimmt (für weitere Details, siehe *Exporteinstellungen*).

- *RDF-Metadaten exportieren* wie z. B. Titel, Autor, Datum, werden mit exportiert
- *Geometrien vereinfachen, um die Größe der Ausgabedatei zu reduzieren*: Dies verhindert, dass ALLE Geometrie-Knotenpunkte exportiert werden, da dies zu einer sehr komplexen und großen Exportdatei führen kann, die u.U. in anderen Anwendungen nicht geladen werden kann. Geometrien werden während des Exports des Layouts vereinfacht, um alle überflüssigen Eckpunkte zu entfernen, die sich bei der Exportauflösung nicht merklich unterscheiden (z.B. wenn die Exportauflösung `300 dpi` beträgt, werden Eckpunkte, die weniger als `1/600 Zoll` voneinander entfernt sind, entfernt).
- Stellen Sie mit der Option *Textexport* ein, ob Textbeschriftungen immer oder vorzugsweise als *Text oder Pfade* exportiert werden sollen.
- *Auf Inhalt zuschneiden* siehe *Einstellungen für Raster-Export*
- *Gekachelte Rasterlayerexporte abschalten*: Beim Exportieren von Dateien verwendet QGIS ein eingebautes Raster Layer Tiled Rendering, das Speicher spart. Manchmal kann dies zu sichtbaren „Nähten“ in den erzeugten Rasterdateien führen. Das Aktivieren dieser Option würde das Problem beheben, allerdings auf Kosten eines höheren Speicherverbrauchs während des Exports.
- Wenn Sie das Kontrollkästchen *Datei nach dem Export öffnen* aktivieren, wird die exportierte Datei automatisch mit dem Standard-SVG-Viewer geöffnet.

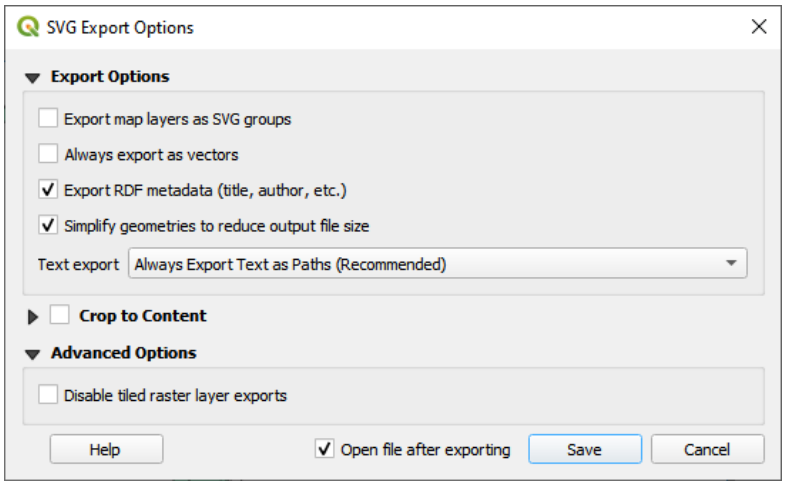


Abb. 18.66: SVG Exportoptionen

Bemerkung

Zur Zeit ist die SVG Ausgabe sehr einfach gehalten. Dies ist kein QGIS Probelm sondern ein Problem mit der darunterliegenden Qt Bibliothek. Dies wird hoffentlich in zukünftigen Versionen behoben.

18.3.4 Als PDF exportieren

So exportieren Sie ein Layout im PDF-Format:

1. Klicken Sie auf das Symbol Als PDF exportieren.
2. Geben Sie den Pfad und den Dateinamen ein: Anders als beim Bild- und SVG-Export werden alle Seiten des Layouts in eine einzige PDF-Datei exportiert.
3. Im folgenden Dialogfeld *PDF-Exportoptionen* können Sie die Voreinstellungen aus den *Exporteinstellungen* des Layouts überschreiben oder neue Einstellungen vornehmen:

   - ☐ *Immer als Vektor exportieren*: einige Rendering-Optionen verlangen, dass Objekte gerastert werden, um eine bessere Darstellung zu erhalten. Aktivieren Sie diese Option, um die Objekte als Vektoren beizubehalten, mit dem Risiko, dass das Aussehen der Ausgabedatei nicht mit der Druckvorschau des Layouts übereinstimmt (für weitere Details, siehe *Exporteinstellungen*).

     > **Bemerkung**
     >
     > **Einstellungen für die Deckkraft von Vektor-Layern**
     >
     > Wenn Sie die Deckkraft eines Layers (oder die Deckkraft für eine Vektorsymbolik) auf weniger als 100 % einstellen, werden diese Objekte beim PDF-Export zwangsweise gerastert dargestellt. Je nach Größe des Layers kann dies zu einer erheblichen Vergrößerung der PDF-Datei führen.

   - ☑ *Georeferenzinformationen anhängen*: nur verfügbar, wenn die *Referenzkarte*, aus der die Informationen entnommen werden, auf der ersten Seite steht.
   - ☑ *RDF-Metadaten exportieren* wie z. B. Titel, Autor, Datum, werden mit exportiert
   - Stellen Sie mit der Option *Textexport* ein, ob Textbeschriftungen immer oder vorzugsweise als *Text oder Pfade* exportiert werden sollen.
   - Steuern Sie die PDF *Bildkompression* mit:
     - *Mit Verlust (JPEG)*, was der Standard-Komprimierungsmodus ist
     - or *Verlustfrei*, die in den meisten Fällen größere Dateien erzeugt, aber für Druckausgaben oder für die Nachbearbeitung in externen Anwendungen viel besser geeignet ist (erfordert Qt 5.13 oder höher).
   - ☐ *Geospatial-PDF erzeugen*: Erzeugt eine georeferenzierte PDF-Datei.
   - ☐ *Gekachelte Rasterlayerexporte abschalten*: Beim Exportieren von Dateien verwendet QGIS ein eingebautes Raster Layer Tiled Rendering, das Speicher spart. Manchmal kann dies zu sichtbaren „Nähten“ in den erzeugten Rasterdateien führen. Das Aktivieren dieser Option würde das Problem beheben, allerdings auf Kosten eines höheren Speicherverbrauchs während des Exports.
   - ☑ *Geometrien für kleinere Dateigrößen vereinfachen*: Geometrien werden während des Exports der Karte vereinfacht, indem Knotenpunkte entfernt werden, die sich bei der Exportauflösung nicht deutlich unterscheiden (z.B. wenn die Exportauflösung `300 dpi` beträgt, werden Eckpunkte, die weniger als `1/600 Zoll` auseinander liegen, entfernt). Dies kann die Größe und Komplexität der Exportdatei reduzieren (sehr große Dateien können u. U. in anderen Anwendungen nicht geladen werden).
   - Wenn Sie das Kontrollkästchen *Datei nach dem Export öffnen* aktivieren, wird die exportierte Datei automatisch mit dem Standard-PDF-Viewer geöffnet.

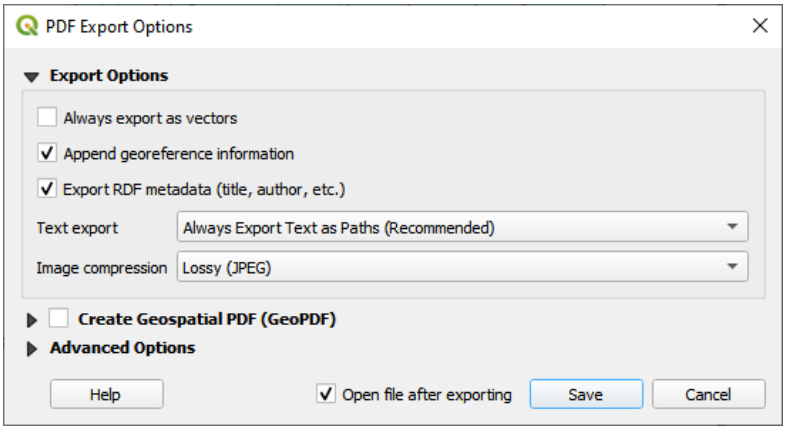


Abb. 18.67: PDF-Exportoptionen

Bemerkung

Der Export von Geospatial PDF wird unterstützt, und es stehen eine Reihe von Geospatial PDF-spezifischen Optionen zur Verfügung:

- *Format* (Geospatial PDF Format - es gibt einige Varianten),
- *Mehrere Kartenthemen einschließen* (geben Sie die einzuschließenden Kartenthemen an),
- guilabel:*Vektorobjektinformation* (wählen Sie die Layer und gruppieren Sie sie in logische PDF-Gruppen).

Bemerkung

Der Export eines Drucklayouts in Formate, die Georeferenzierung unterstützen (z.B. `PDF` und `TIFF`), erzeugt standardmäßig eine georeferenzierte Ausgabe.

Bemerkung

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 Einen Atlas erzeugen

Mit den Atlas-Funktionen können Sie Kartenwerke auf automatisierte Weise erstellen. Atlas verwendet die Merkmale einer Tabelle oder eines Vektor-Layers (*Coverage layer*), um eine Ausgabe für jedes Objekt (**atlas feature**) in der Tabelle / dem Layer zu erstellen. Die häufigste Anwendung ist das Zoomen eines Kartenobjekts auf das aktuelle Atlasobjekt. Weitere Anwendungsfälle sind:

- ein Kartenelement, das für einen anderen Layer nur Objekte anzeigt, die dasselbe Attribut wie das Atlasobjekt haben oder innerhalb seiner Geometrie liegen.
- eine Beschriftung oder ein HTML-Element, dessen Text bei der Iteration von Objekten ersetzt wird
- ein Tabellenelement, das die Attribute der zugehörigen *Eltern- oder Kindobjekte* des aktuellen Atlasobjekts anzeigt…

Für jedes Objekt wird die Ausgabe für alle Seiten und Artikel entsprechend ihren Exporteinstellungen verarbeitet.

> **Tipp**
>
> **Verwenden Sie Variablen für mehr Flexibilität**
>
> QGIS bietet ein umfangreiches Bedienfeld mit Funktionen und Variablen 1, einschließlich atlasbezogener Funktionen, mit denen Sie die Layout-Elemente, aber auch die Symbologie der Layer je nach Atlasstatus manipulieren können. Die Kombination dieser Objekte gibt Ihnen eine große Flexibilität und hilft Ihnen, auf einfache Weise komplexe Karten zu erstellen.

Das *Atlas* Bedienfeld ermöglicht die Erzeugung eines Atlas und den Zugriff auf Atlasparameter. Dieses Bedienfeld enthält die folgenden Funktionen (siehe Abb. 18.68):

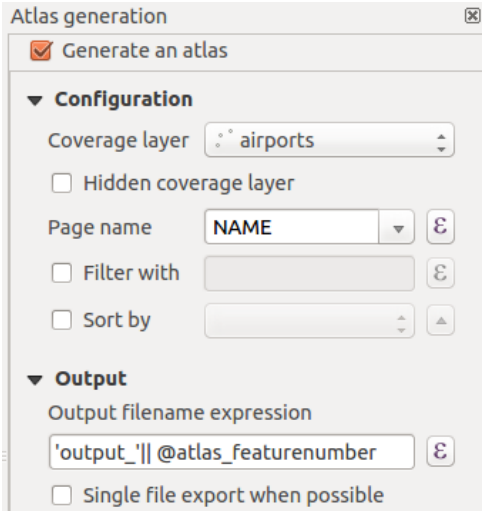


Abb. 18.68: Atlas Bedienfeld

- *Einen Atlas erzeugen* aktiviert oder deaktiviert die Atlaserzeugung.
- *Konfiguration*
  - Ein *Abdeckungslayer* Kombinationsfeld, mit dem Sie den Tabellen- oder Vektor-Layer auswählen können, der die Objekte enthält, über die iteriert werden soll.
  - Ein optionales Kontrollkästchen *Abdeckungslayer verstecken*, das, wenn es aktiviert ist, den Abdeckungslayer (aber nicht die anderen Layer) während der Generierung versteckt.
  - Ein optionales Kombinationsfeld *Seitenname* zur Angabe des Namens für die Objektseite(n). Sie können ein Feld des Abdeckungslayers auswählen oder einen *Ausdruck* setzen. Ist diese Option leer, verwendet QGIS eine interne ID, entsprechend dem Filter und/oder der Sortierreihenfolge, die auf den Layer angewendet wurde.
  - Ein optionales Kontrollkästchen *Filtern nach*, in dem Sie einen Ausdruck für die Filterung von Objekten aus dem Coverage Layer angeben können. Wenn der Ausdruck nicht leer ist, werden nur Objekte verarbeitet, die den Wert `True` haben.
  - Ein optionales Kontrollkästchen *Sortieren nach*, mit dem Sie Objekte des Coverage Layers (und der Ausgabe) anhand eines Feldes des Coverage Layers oder eines Ausdrucks sortieren können. Die Sortierreihenfolge (entweder aufsteigend oder absteigend) wird durch die zweistufige Schaltfläche *Sortierrichtung* festgelegt, die einen Pfeil nach oben oder unten anzeigt.
- *Ergebnis* - hier kann die Ausgabe des Atlas konfiguriert werden:
  - Ein *Ausgabedateiausdruck*-Textfeld, das verwendet wird, um einen Dateinamen für jedes Atlas-Objekt zu erzeugen. Es basiert auf Ausdrücken. Ist nur für das Rendern in mehrere Dateien sinnvoll.
  - Ein Kontrollkästchen *Einzeldatei, wenn möglich*, mit dem Sie die Erzeugung einer einzigen Datei erzwingen können, wenn dies mit dem gewählten Ausgabeformat möglich ist (z.B. `PDF`). Wenn dieses

Feld angekreuzt ist, ist der Wert des Feldes *Ausgabedateiausdruck* bedeutungslos.

- Eine Dropdown-Liste *Bildexportformat* zur Auswahl des Ausgabeformats bei Verwendung der Schaltfläche Atlas als Bilder exportieren… .

Karte über Atlas steuern

Am häufigsten wird der Atlas mit dem Kartenelement verwendet, wobei auf das aktuelle Atlas-Objekt gezoomt wird, wenn die Iteration über den Coverage Layer geht. Dieses Verhalten wird in den Gruppeneigenschaften *Gesteuert durch Atlas* des Kartenelements eingestellt. Siehe *Gesteuert durch Atlas* für verschiedene Einstellungen, die Sie auf das Kartenelement anwenden können.

Beschriftungen mit Ausdruck anpassen

Um Beschriftungen an das Objekt anzupassen, über das der Atlas iteriert, können Sie Ausdrücke einfügen. Stellen Sie sicher, dass Sie den Ausdrucksteil (einschließlich Funktionen, Felder oder Variablen) zwischen `[%` und `%]` platzieren (siehe layout_label_item` für weitere Details).

Für einen Städte-Layer mit den Feldern `CITY_NAME` (Städtename) und `ZIPCODE` (Postleitzahl) könnte man zum Beispiel folgendes einfügen:

```
The area of [% concat( upper(CITY_NAME), ',', ZIPCODE, ' is ',
format_number($area/1000000, 2) ) %] km2
```

oder, andere Kombinationen:

```
The area of [% upper(CITY_NAME)%],[%ZIPCODE%] is
[%format_number($area/1000000,2) %] km2
```

Die Information `[% concat( upper(CITY_NAME), ',', ZIPCODE, ' is ', format_number($area/1000000, 2) ) %]` ist ein Ausdruck, der innerhalb der Beschriftung verwendet wird. Beide Ausdrücke würden im erzeugten Atlas zu folgender Art von Beschriftung führen:

```
The area of PARIS,75001 is 1.94 km2
```

Die datendefinierte Übersteuerung in einem Atlas

Es gibt mehrere Stellen, an denen Sie eine Schaltfläche Datendefinierte Übersteuerung verwenden können, um die gewählte Einstellung zu steuern. Dies ist besonders nützlich bei der Atlaserzeugung. Siehe *Einrichten von datendefinierter Übersteuerung* für weitere Details zu diesem Widget.

Für die folgenden Beispiele wird der Layer `Regions` des QGIS-Beispieldatensatzes verwendet und als *Abdeckungslayer* für die Atlaserstellung ausgewählt. Wir nehmen an, dass es sich um ein einseitiges Layout handelt, das ein Kartenelement und ein Beschriftungselement enthält.

Wenn die Höhe (Nord-Süd) einer Regionsausdehnung größer ist als ihre Breite (Ost-West), soll die Ausrichtung *Portrait* (Hochformat) statt *Landscape* (Querformat) verwendet werden, um die Papierausnutzung zu optimieren. Mit der Schaltfläche Datendefinierte Übersteuerung können Sie die Papierausrichtung dynamisch festlegen.

Klicken Sie mit der rechten Maustaste auf die Seite und wählen Sie *Seiteneigenschaften*, um das Bedienfeld zu öffnen. Wir wollen die Ausrichtung dynamisch festlegen, indem wir einen Ausdruck verwenden, der von der Geometrie des Bereichs abhängt. Drücken Sie also die Schaltfläche des Feldes *Orientierung*, wählen Sie *Bearbeiten…*, um den Dialog *Expression Builder* (Ausdrucksgenerator) zu öffnen und geben Sie den folgenden Ausdruck ein:

```
CASE WHEN bounds_width(@atlas_geometry) > bounds_height(@atlas_geometry)
THEN 'Landscape' ELSE 'Portrait' END
```

Wenn Sie nun *den Atlas anzeigen lassen*, richtet sich das Papier automatisch aus, aber die Platzierung der Elemente ist möglicherweise nicht ideal. Für jede Region müssen Sie auch die Position der Layoutelemente neu festlegen. Für die Kartenelemente können Sie die Schaltfläche der Eigenschaft *Breite* verwenden, um sie mit dem folgenden Ausdruck dynamisch einzustellen:

```
@layout_pagewidth - 20
```

Verwenden Sie ebenfalls die Schaltfläche der Eigenschaft *Höhe*, um den folgenden Ausdruck zur Einschränkung der Größe des Kartenelements bereitzustellen:

```
@layout_pageheight - 20
```

Um sicherzustellen, dass das Kartenelement auf der Seite zentriert ist, setzen Sie seinen *Referenzpunkt* auf das obere linke Optionsfeld und geben Sie `10` für seine *X* und *Y* Positionen ein.

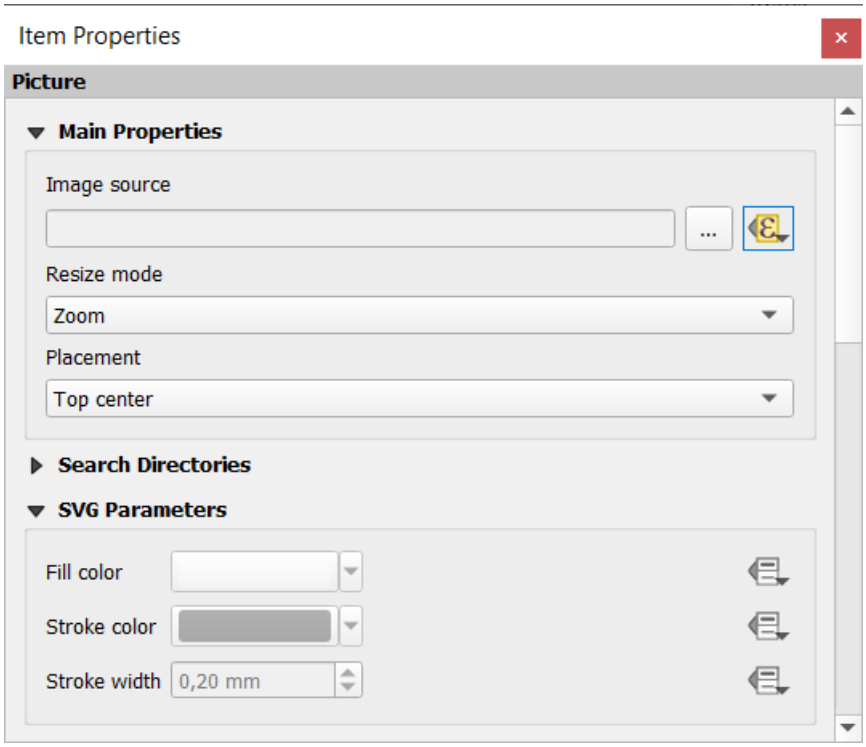
Fügen wir einen Titel oberhalb der Karte in der Mitte der Seite hinzu. Wählen Sie das Beschriftungselement und setzen Sie die horizontale Ausrichtung auf *Zentriert*. Verschieben Sie die Beschriftung an die rechte Position, wählen Sie die mittlere Schaltfläche für *Referenzpunkt* und geben Sie den folgenden Ausdruck für das Feld *X* ein:

```
@layout_pagewidth / 2
```

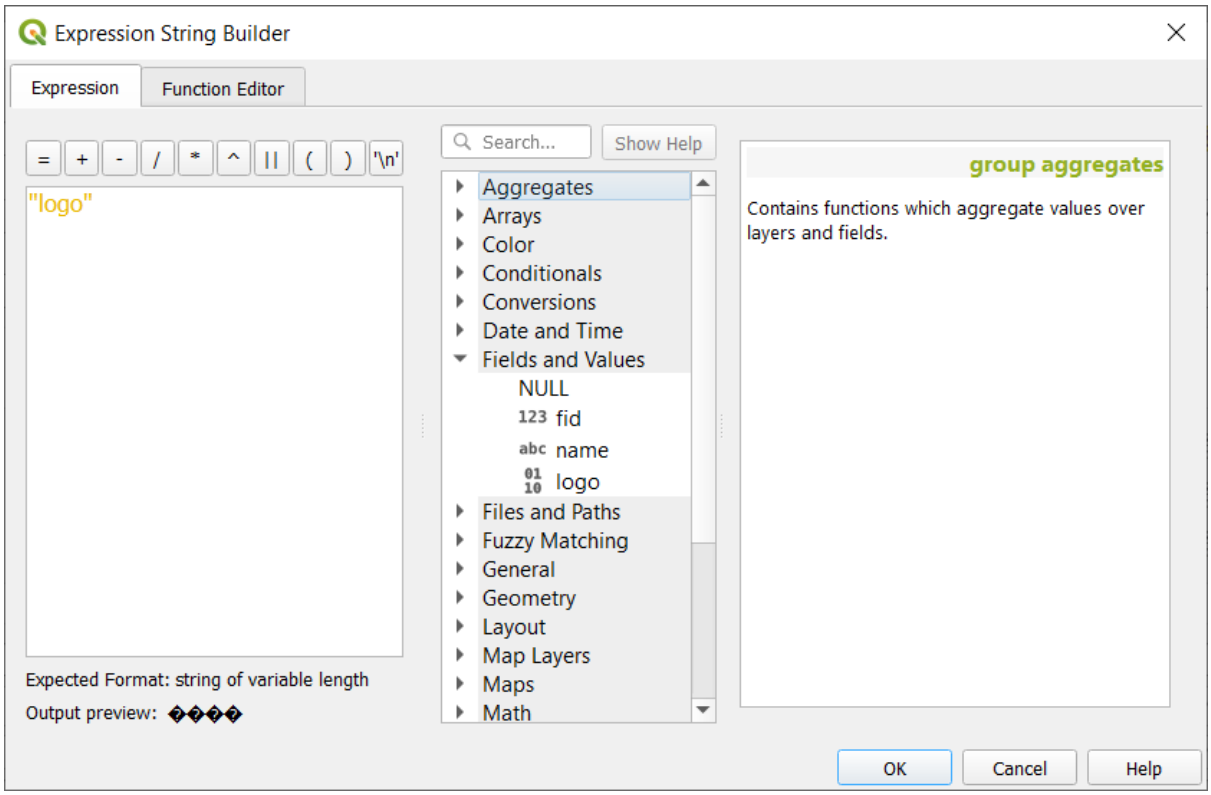
Für alle anderen Layoutelemente können Sie die Position auf ähnliche Weise festlegen, so dass sie sowohl im Hoch- als auch im Querformat korrekt positioniert sind. Sie können auch weitere Anpassungen vornehmen, z. B. den Titel mit Objektattributen anpassen (siehe Beispiel *Beschriftungen mit Ausdruck anpassen*), Bilder ändern, die Anzahl der Legendenspalten entsprechend der Seitenausrichtung anpassen, …

Die hier bereitgestellten Informationen sind ein Update des ausgezeichneten Blogs (in Englisch und Portugiesisch) über die Data Defined Override Optionen Multiple_format_map_series_using_QGIS_2.6 .

Ein weiteres Beispiel für die Verwendung von datendefinierter Übersteuerung ist die Verwendung eines dynamischen Bildes. Für die folgenden Beispiele verwenden wir einen Geopackage Layer, der ein BLOB-Feld namens `logo` mit dem Feldtyp binary enthält (siehe vector_create_geopackage`). Für jedes Objekt wird ein anderes Bild definiert, damit der Atlas darüber iterieren kann, wie in *Vorschau und Erstellung eines Atlas* beschrieben. Alles, was Sie tun müssen, ist, ein Bild im Drucklayout hinzuzufügen und im Atlaskontext zu seinem *Elementeigenschaften* zu gehen. Dort finden Sie im Bereich *Bild* für ein Rasterbild das Auswahlmenü für den Dateipfad und dahinter die Schaltfläche für die *Datendefinierte Übersteuerung*.



Wählen Sie im sich öffnenden Fenster *Bearbeiten…*, sodass sich der *Expression Builder* (Ausdrucksgenerator) öffnet. Im Bereich *Felder und Werte* finden Sie das BLOB-Feld, das im Layer geopackage definiert wurde. Doppelklicken Sie auf den Feldnamen `logo` und klicken Sie auf *OK*.



Der Atlas iteriert über die Einträge im BLOB-Feld, vorausgesetzt, Sie wählen den Layer des Geopakets als *Abdeckungslayer* (weitere Anweisungen finden Sie in *Vorschau und Erstellung eines Atlas*).

Dies sind nur zwei Beispiele dafür, wie Sie einige erweiterte Einstellungen mit Atlas verwenden können.

Vorschau und Erstellung eines Atlas

Abb. 18.69: Atlas-Vorschauwerkzeuge

Sobald die Atlaseinstellungen konfiguriert und Layoutelemente (Karte, Tabelle, Bild…) damit verknüpft sind, können Sie eine Vorschau aller Seiten erstellen, indem Sie *Atlas* ► *Atlas-Vorschau anzeigen* wählen oder auf das Symbol Atlas-Vorschau anzeigen klicken. Sie können dann mit den Pfeilen durch alle Objekte navigieren:

- Erstes Objekt
- Vorheriges Objekt
- Nächstes Objekt
- Letztes Objekt

Sie können auch das Auswahlmenü verwenden, um ein bestimmtes Objekt auszuwählen und in der Vorschau anzuzeigen. Das Auswahlmenü zeigt die Namen der Objekte im Atlas entsprechend dem in der Option atlas *Seitenname* festgelegten Ausdruck an.

Wie bei einfachen Karten-Zusammenstellungen kann ein Atlas auf verschiedene Weise erzeugt werden (siehe *Eine Ausgabe erzeugen* für weitere Informationen - verwenden Sie einfach Werkzeuge aus dem Menü *Atlas* oder der Werkzeugleiste anstelle des Menüs *Layout*.

Das bedeutet, dass Sie Ihre Zusammenstellungen mit *Atlas* ► *Atlas drucken* direkt ausdrucken können. Sie können auch ein PDF mit *Atlas* ► *Atlas als PDF exportieren…* erstellen: Sie werden nach einem Verzeichnis gefragt, in dem alle erzeugten PDF-Dateien gespeichert werden sollen, es sei denn, das Kontrollkästchen *Einzeldateiexport wenn möglich* wurde aktiviert. In diesem Fall werden Sie aufgefordert, einen Dateinamen anzugeben.

Mit dem Werkzeug *Atlas* ► *Atlas als Bilder exportieren…* oder *Atlas* ► *Atlas als SVG exportieren…* werden Sie auch aufgefordert, einen Ordner auszuwählen. Jede Seite jedes Objekts wird in das im Bedienfeld *Atlas* eingestellte Bilddateiformat oder in SVG exportiert.

Bemerkung

Bei mehrseitiger Ausgabe verhält sich ein Atlas wie ein Layout, da nur die Seite, die die *Allgemeine Einstellungen* enthält, eine World-Datei erhält (für jedes ausgegebene Objekt).

Tipp

Drucken eines bestimmten Atlasobjektes

Wenn Sie nur ein bestimmtes Objekt aus dem Atlas ausdrucken oder exportieren möchten, starten Sie einfach die Vorschau, wählen das gewünschte Objekt in der Dropdown-Liste aus und klicken auf *Layout* ► *Drucken* (oder *Exportieren…* in ein beliebiges unterstütztes Dateiformat).

Verwendung projektdefinierter Beziehungen für die Erstellung von Atlanten

Für Benutzer mit HTML- und Javascript-Kenntnissen ist es möglich, mit GeoJSON-Objekten zu arbeiten und projektdefinierte Beziehungen aus dem QGIS-Projekt zu verwenden. Der Unterschied zwischen diesem Ansatz und der Verwendung von Ausdrücken, die direkt in den HTML-Code eingefügt werden, besteht darin, dass Sie ein vollständiges, unstrukturiertes GeoJSON-Objekt erhalten, mit dem Sie arbeiten können. Das bedeutet, dass Sie bestehende Javascript-Bibliotheken und Funktionen verwenden können, die mit GeoJSON-Objekten arbeiten.

Der folgende Code enthält alle zugehörigen untergeordneten Objekte der definierten Beziehung. Mit der JavaScript-Funktion `setFeature` können Sie flexible HTML-Dateien erstellen, die Beziehungen in jedem beliebigen Format darstellen (Listen, Tabellen usw.). Im Codebeispiel erstellen wir eine dynamische Aufzählung der zugehörigen untergeordneten Objekte.

```
// 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>
```

Bei der Erstellung des Atlas wird über den Layer mit den übergeordneten Objekten iteriert. Auf jeder Seite sehen Sie eine Aufzählung der zugehörigen untergeordneten Objekte, die der Kennung des übergeordneten Objekts folgt.

18.4 Einen Bericht erstellen

Dieser Abschnitt hilft Ihnen, einen Bericht in QGIS zu erstellen.

18.4.1 Was ist ein Bericht?

Per Definition ist ein GIS-Bericht ein Dokument, das Informationen in Form von Karten, Text, Grafiken, Tabellen usw. enthält. Ein Bericht kann ad hoc, periodisch, wiederkehrend, regelmäßig oder nach Bedarf erstellt werden. Berichte können sich auf bestimmte Zeiträume, Ereignisse, Vorkommnisse, Themen oder Orte beziehen.

In QGIS ist ein *Bericht* eine Erweiterung eines *Layouts*.

Berichte ermöglichen es den Nutzern, ihre GIS-Projekte auf einfache, schnelle und strukturierte Weise auszugeben.

Ein Bericht kann mit *Projekt* ► *Neuer Bericht* oder im Fenster *Projekt* ► *Layout-Verwaltung* erstellt werden.

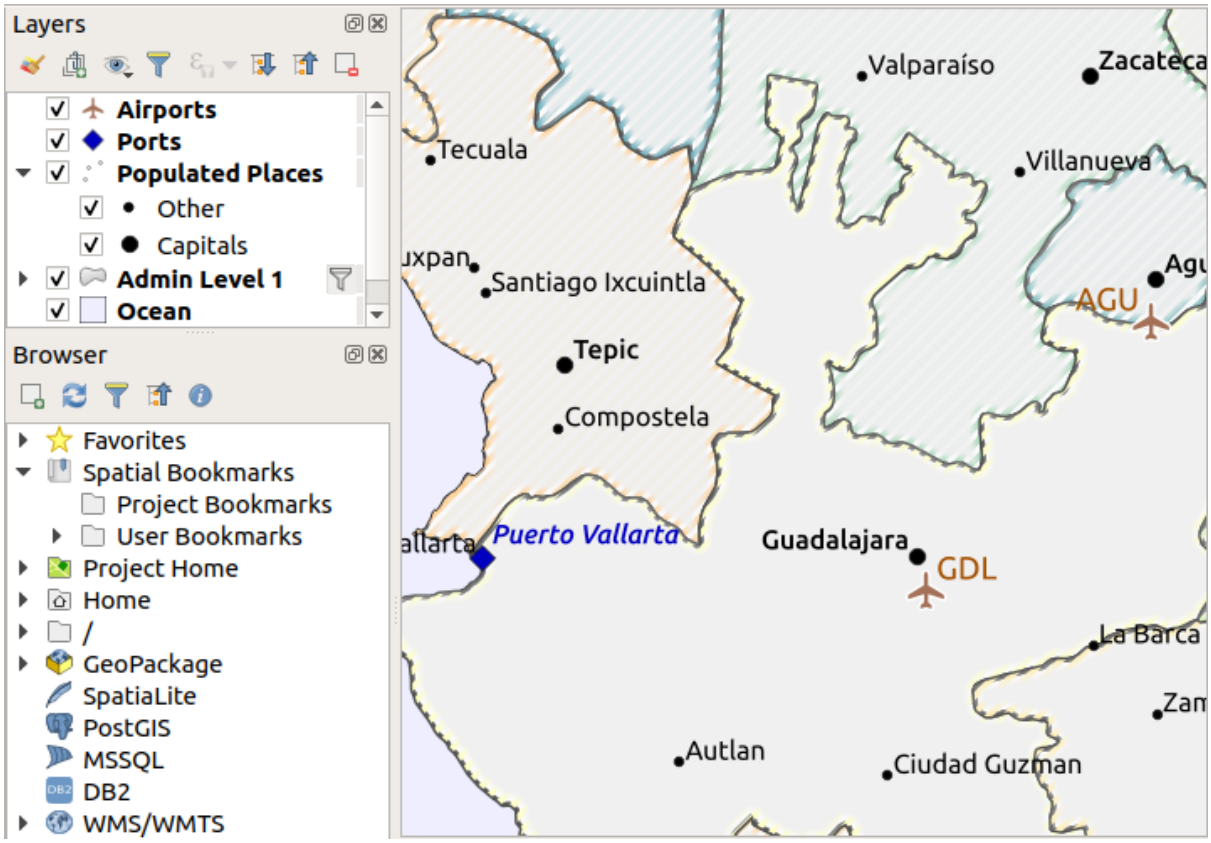
Bemerkung

Die Karten in QGIS-Berichten verhalten sich genauso wie die Karten in Drucklayouts und Atlanten. Wir werden uns auf die Besonderheiten von QGIS-Berichten konzentrieren. Details zur Handhabung von Karten finden Sie in den Abschnitten *Druck-Layouts* und *Atlanten*.

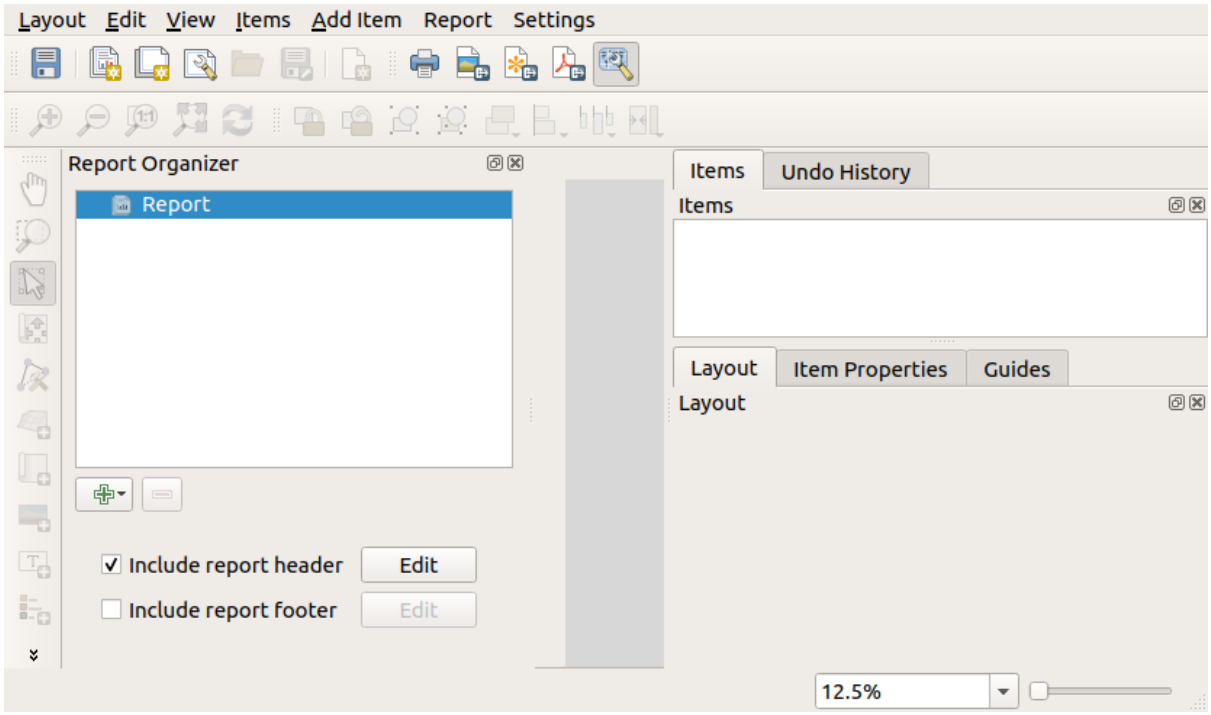
18.4.2 Erste Schritte

Im Dialog *Layout-Verwaltung* kann ein Bericht unter *Neu aus Vorlage* erstellt werden, indem die Dropdown-Option *Leerer Bericht* ausgewählt und die Schaltfläche *Erstellen…* betätigt wird.

Für dieses Beispiel verwenden wir einige Verwaltungsgrenzen, besiedelte Orte, Häfen und Flughäfen aus dem Natural EarthDatensatz (1:10M).

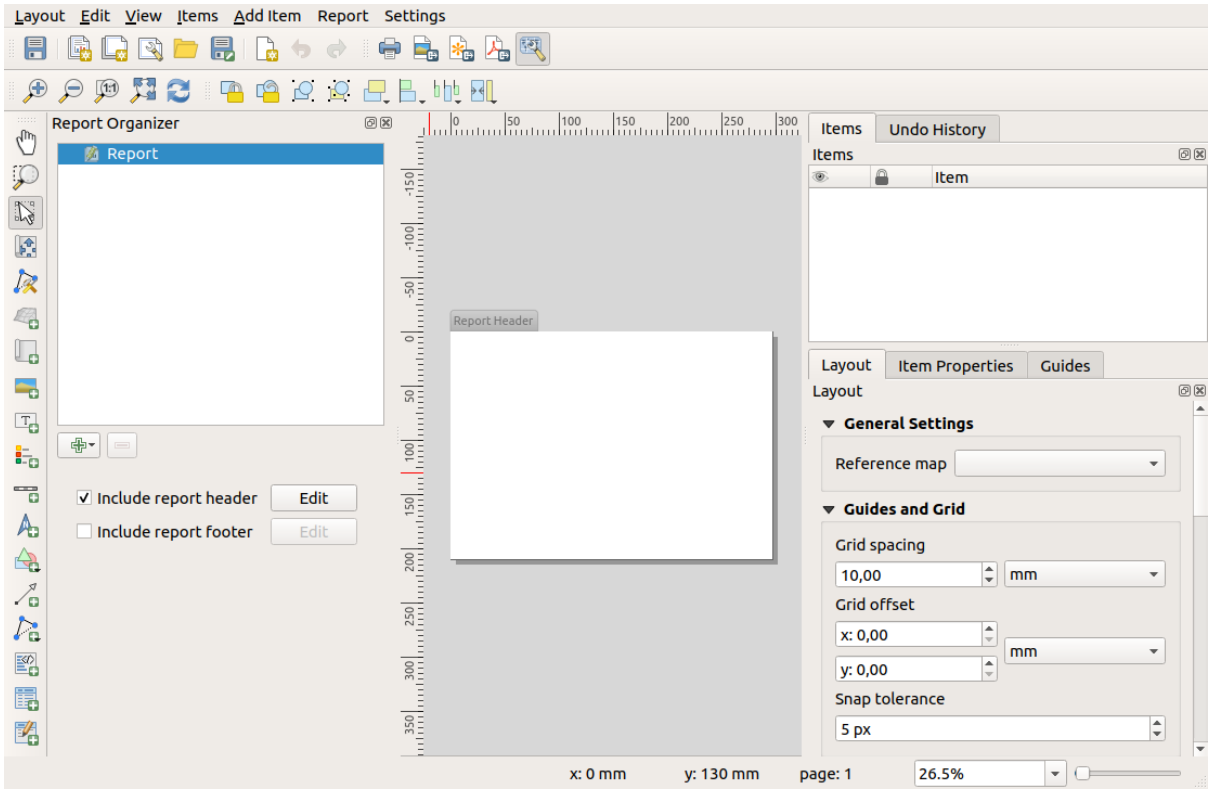


Mit dem Befehl *Projekt* ► *Neuer Bericht* erstellen wir einen leeren Bericht. Der Dialog, der angezeigt wird, ist dem Fenster zum Erstellen eines Druck-Layouts ähnlich, zusätzlich gibt es aber das Bedienfeld *Berichtsorganisation* auf der linken Seite:



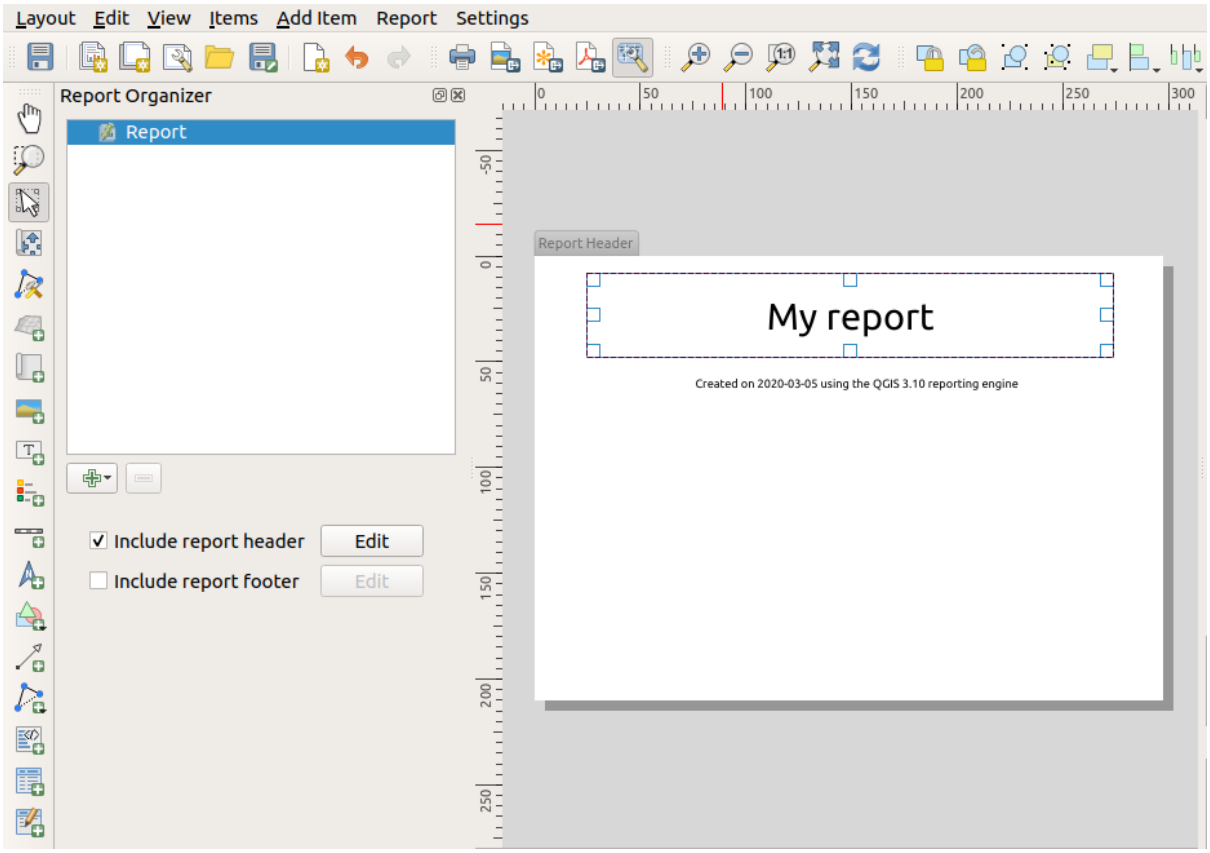
18.4.3 Arbeitsbereich für Layout-Berichte

QGIS-Berichte können aus mehreren, verschachtelten Abschnitten bestehen. In unserem neuen leeren Bericht gibt es zunächst nur den Hauptberichtsteil. Die einzigen Optionen für diesen Berichtsabschnitt sind *Berichtskopf einschließen* und *Berichtsfuß einschließen*. Wenn wir diese Optionen aktivieren, wird eine Kopfzeile als erste Seite(n) in den Bericht eingefügt (einzelne Berichtsteile können auf Wunsch mehrseitig sein), und eine Fußzeile wird die letzte(n) Seite(n) bilden. Aktivieren Sie die Kopfzeile (*Berichtskopf einschließen*), und klicken Sie auf die Schaltfläche *Bearbeiten* daneben:

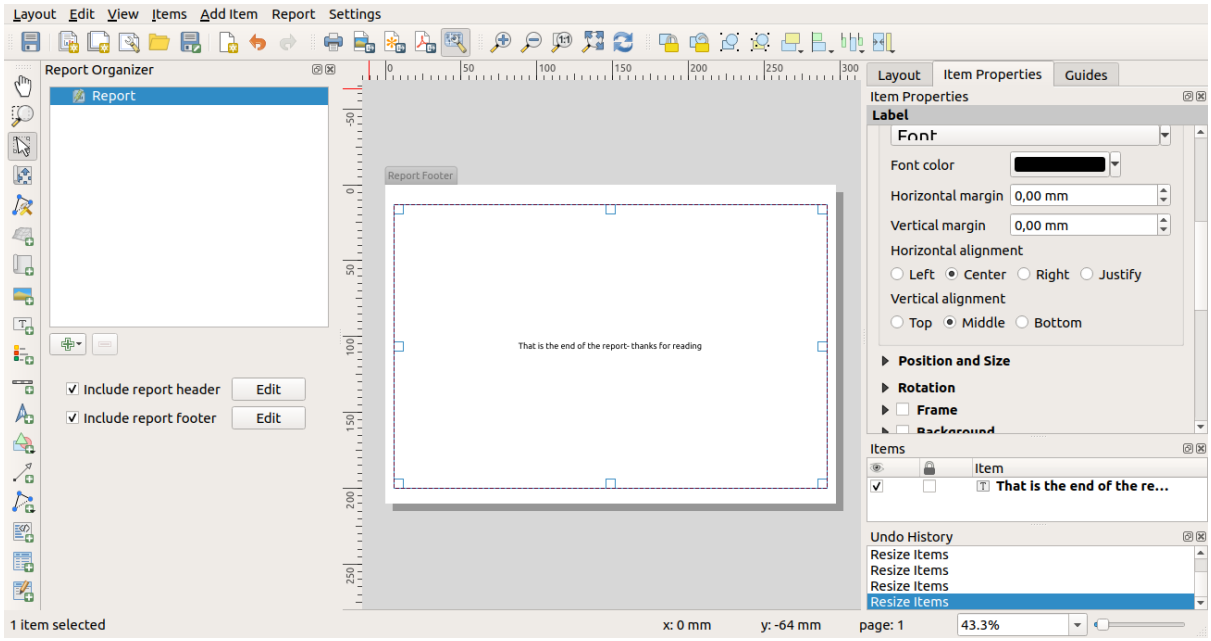


Infolgedessen geschehen einige Dinge. Erstens wird ein Bearbeitungsstift neben *Bericht* im *Berichtsorganisator* angezeigt, was darauf hinweist, dass der Berichtsabschnitt gerade im Designer bearbeitet wird. Wir sehen auch eine neue Seite mit einem kleinen *Berichtskopf*-Titel. Die Seite ist standardmäßig im Querformat ausgerichtet, aber dies (und andere Eigenschaften der Seite) können Sie ändern, indem Sie mit der rechten Maustaste auf die Seite klicken und *Seiteneigenschaften* wählen. Dadurch wird die Registerkarte *Elementeigenschaften* für die Seite aufgerufen, und die Seitengrößen *Größe*, *Breite*, *Höhe* und mehr können festgelegt werden.

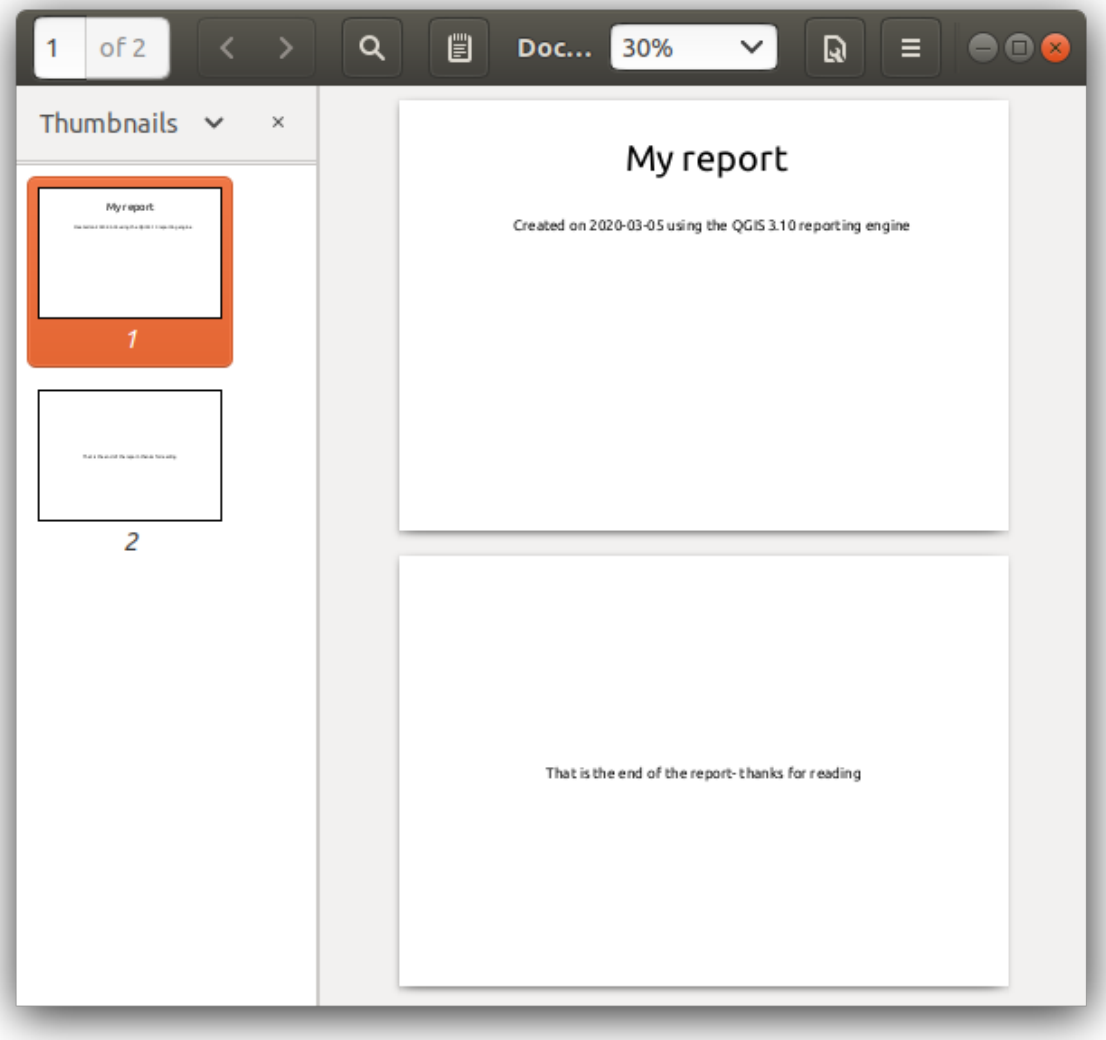
In QGIS-Berichten besteht jeder Bestandteil des Berichts aus einzelnen Layouts. Sie können mit denselben Werkzeugen wie für Standard-Drucklayouts erstellt und geändert werden - Sie können also jede gewünschte Kombination von Beschriftungen, Bildern, Karten, Tabellen usw. verwenden. Fügen wir zur Veranschaulichung einige Elemente zu unserem Berichtskopf hinzu:



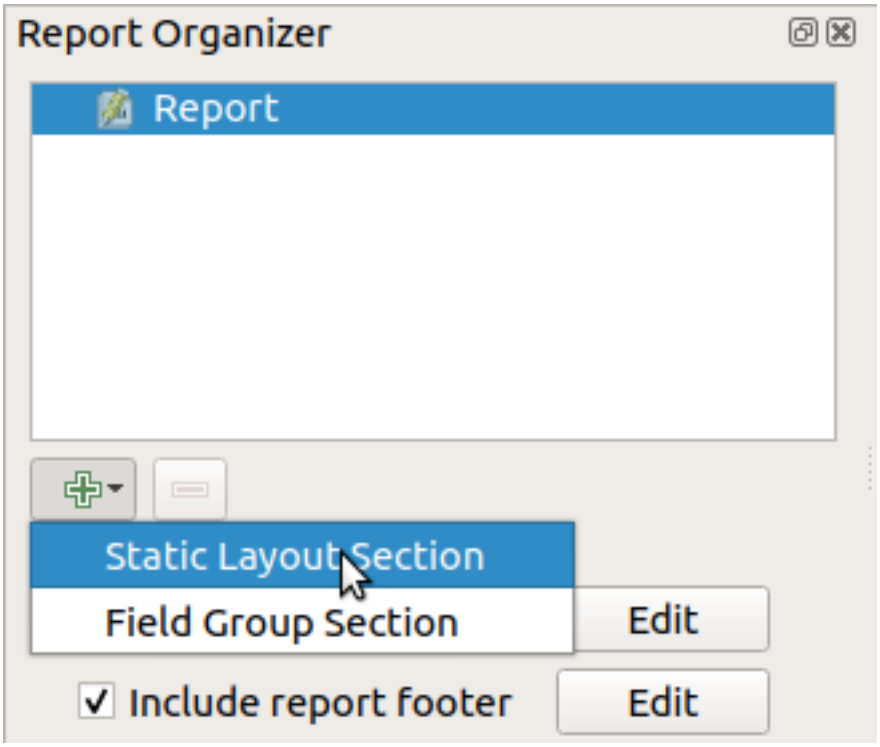
Wir werden auch eine einfache Fußzeile für den Bericht erstellen, indem wir die Option *Berichtsfuß einschließen* aktivieren und auf *Bearbeiten* klicken.



Bevor wir weitermachen, exportieren wir diesen Bericht und sehen, was wir erhalten. Das Exportieren erfolgt über das Menü *Bericht* - in diesem Fall wählen wir *Bericht als PDF exportieren…*, um den gesamten Bericht in eine PDF-Datei zu übertragen. Hier ist das nicht sehr beeindruckende Ergebnis - ein zweiseitiges PDF, das aus unserer Kopf- und Fußzeile besteht:



Lassen Sie uns die Dinge interessanter gestalten. Wenn Sie auf die Schaltfläche Abschnitt hinzufügen in *Berichtsorganisation* klicken, haben Sie die Möglichkeit, neue Abschnitte zu Ihrem Bericht hinzuzufügen.

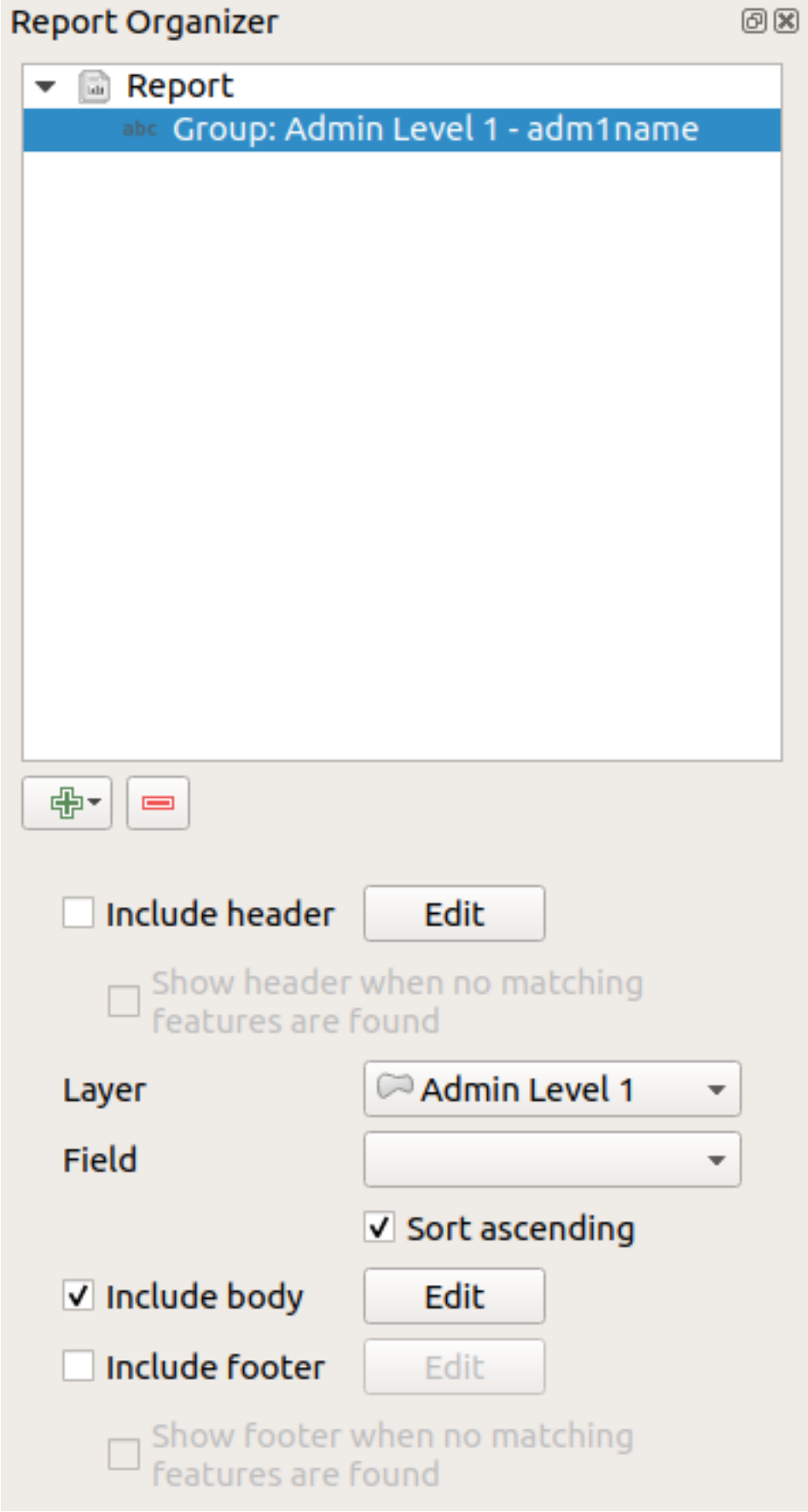


Es gibt zwei Optionen: *Statischer Layoutabschnitt* und *Feldgruppenabschnitt*.

Der *Statische Layoutabschnitt* ist ein einzelnes, statisches Layout-Körper. Dieser kann verwendet werden, um statische Layouts in der Mitte eines Berichts einzubetten.

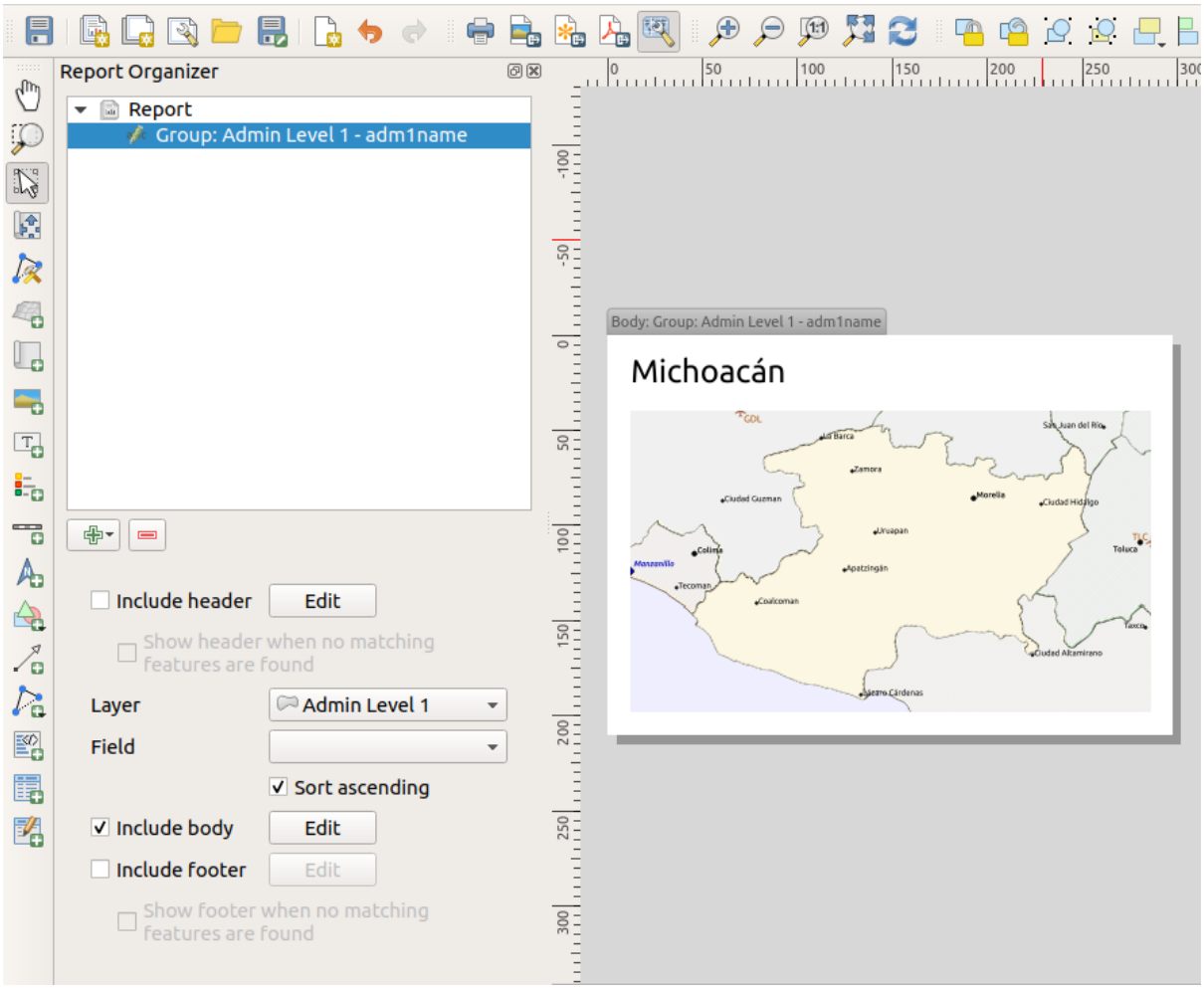
Der *Feldgruppenabschnitt* wiederholt das Layout für jedes Objekt eines Layers. Die Objekte werden nach dem ausgewählten Feld sortiert (mit der Option für aufsteigende/absteigende Sortierung). Wenn ein Feldgruppenabschnitt untergeordnete Abschnitte hat (z. B. einen anderen Feldgruppenabschnitt mit einem anderen Feld), wird nur über Objekte mit eindeutigen Werten für das ausgewählte Feld iteriert. Dies ermöglicht verschachtelte Berichte.

Für den Moment werden wir unserem Bericht einen *Feldgruppenabschnitt* hinzufügen. Grundsätzlich kann man sich einen *Feldgruppenabschnitt* als Äquivalent zu einem *Atlas* vorstellen: Sie wählen einen Layer aus, über den Sie iterieren, und der Bericht fügt für jedes gefundene Objekt einen Abschnitt ein. Wenn Sie den neuen *Feldgruppenabschnitt* einfügen, erhalten Sie eine Reihe von neuen, damit verbundenen Einstellungen:



In diesem Fall haben wir unsere Feldgruppe so eingerichtet, dass wir alle Zustände des Layers *Admin Level 1* unter

Verwendung der Werte des Feldes *adm1name“ durchlaufen. Die gleichen Optionen für Kopf- und Fußzeile sind vorhanden, zusammen mit einer neuen Option, um ein :guilabel:`Körper* für diesen Abschnitt einzufügen. Wir werden das tun und den Textkörper bearbeiten:

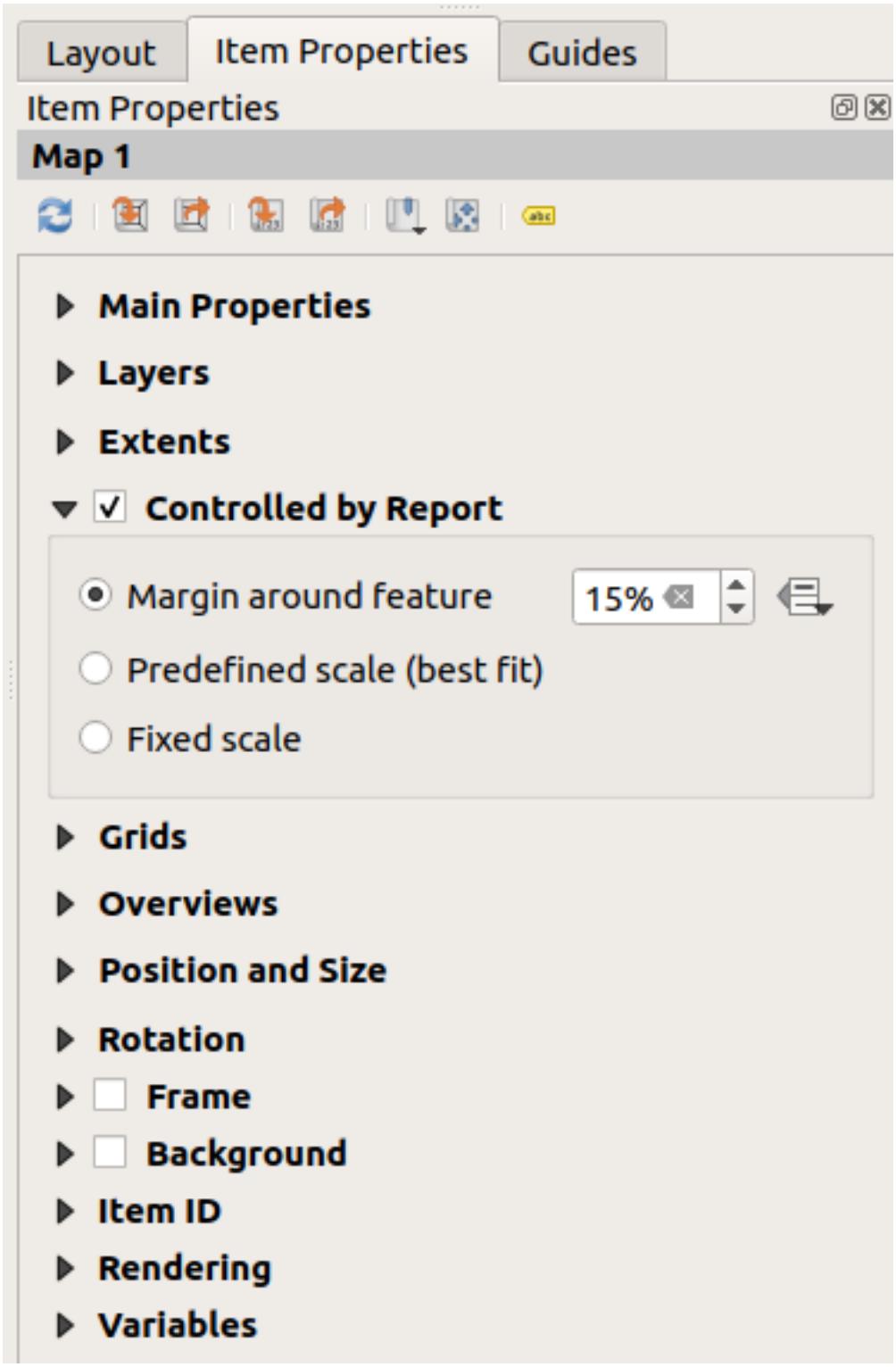


Unser Körper besteht nun aus einer Karte und einer Beschriftung mit dem Namen des Staates. Um den Namen des Staates einzufügen, wählen wir *Element hinzufügen* ► *Beschriftung hinzufügen* und definieren den Text unter *Haupteigenschaften* mit Hilfe von *Einfügen oder Bearbeiten eines Ausdrucks*….

Das Ergebnis ist der folgende Ausdruck (*name* ist der Name des Attributs im Layer *Admin Level 1*, das den Namen des Zustands enthält):

```
[% "name" %]
```

Die Karte wird so eingestellt, dass sie dem aktuellen Berichtsobjekt folgt (aktiviert durch Ankreuzen von *Gesteuert durch Bericht* - so wie ein Kartenelement in einem Atlas dem aktuellen Atlasobjekt folgt, wenn *Gesteuert durch Atlas* aktiviert ist):



Wenn wir unseren Bericht jetzt exportieren würden, bekämen wir in etwa das folgende Ergebnis:

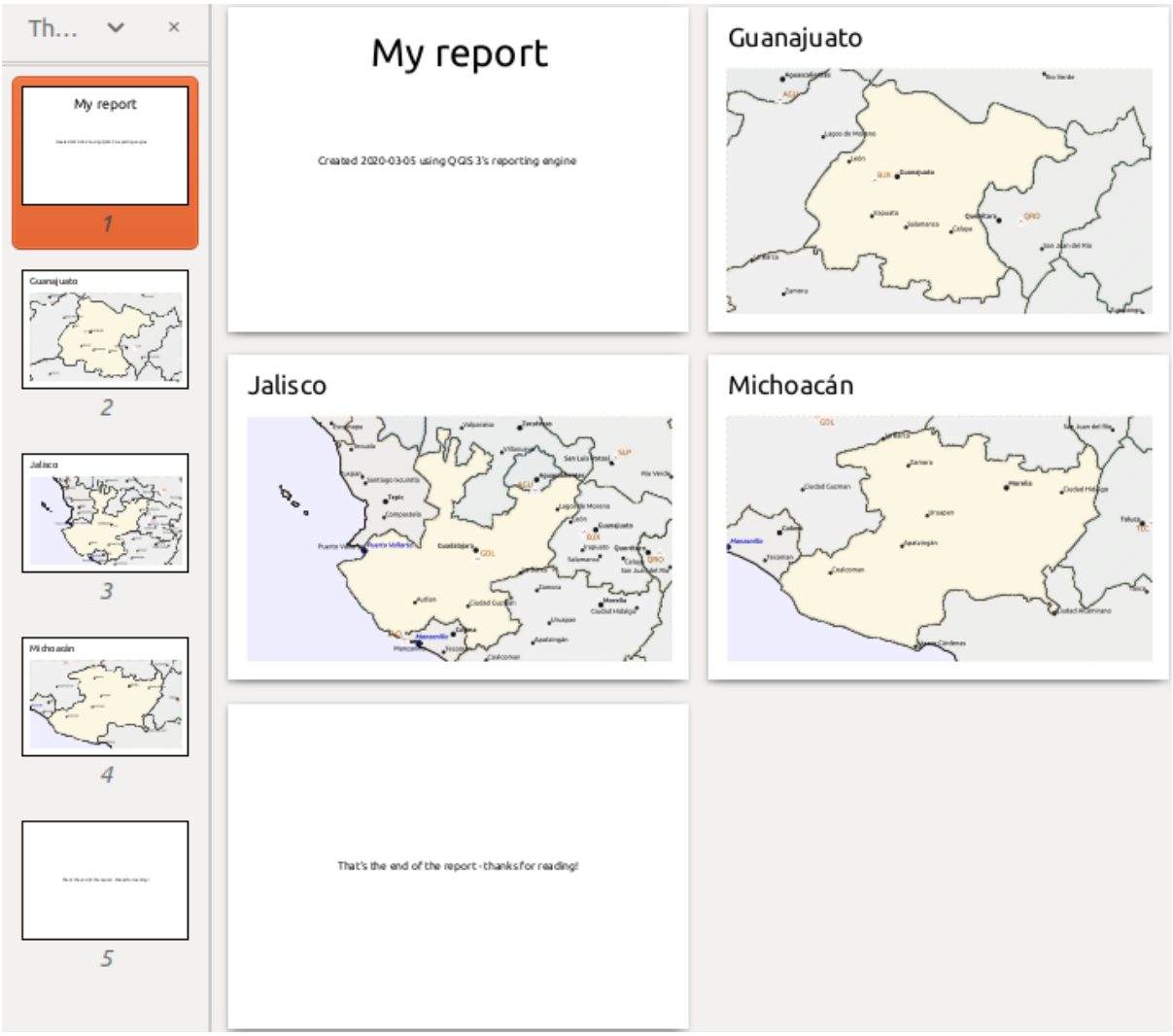
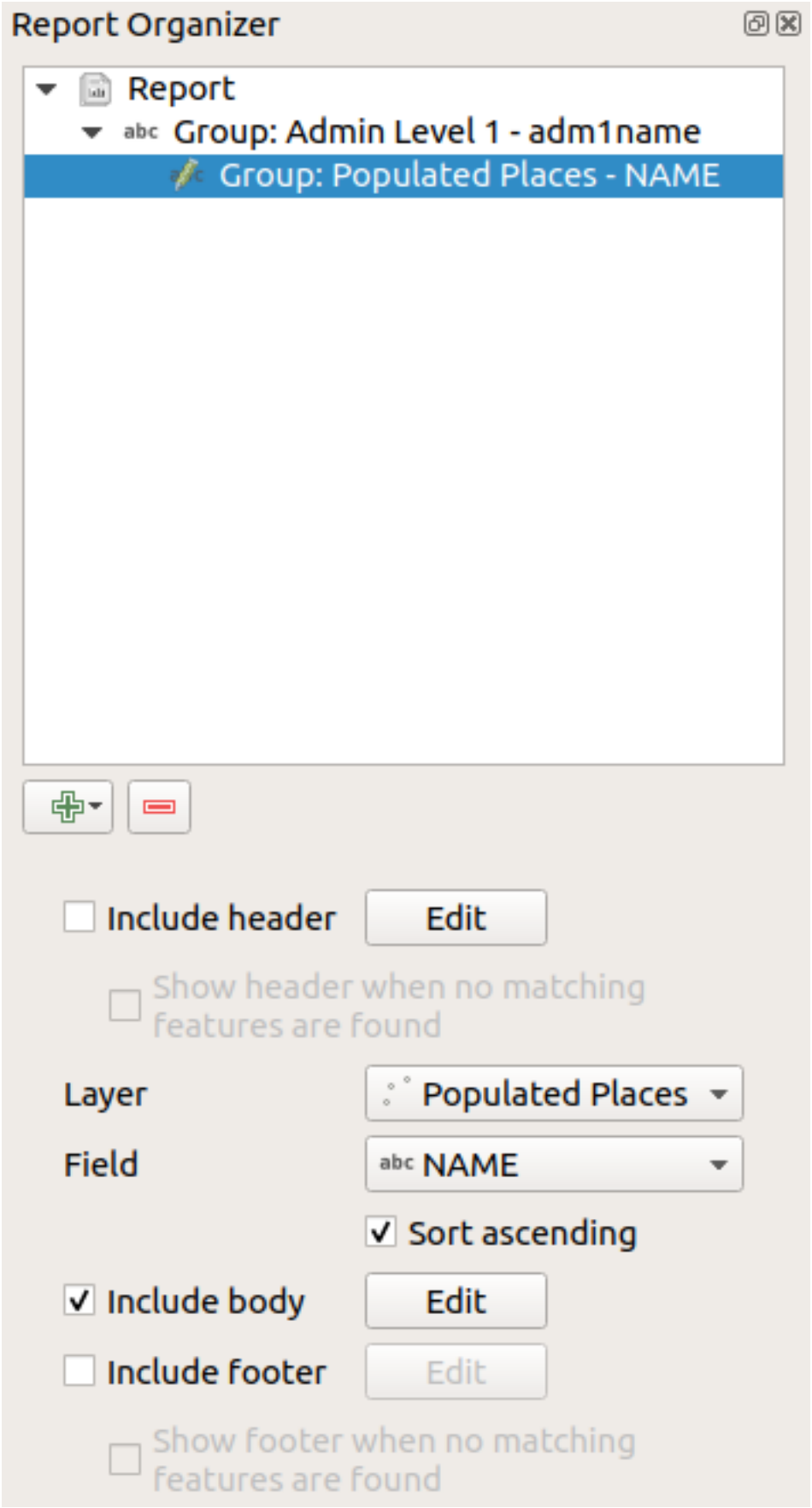


Abb. 18.70: Die Kopfzeile des Berichts, eine Seite für jeden Staat und die Fußzeile des Berichts.

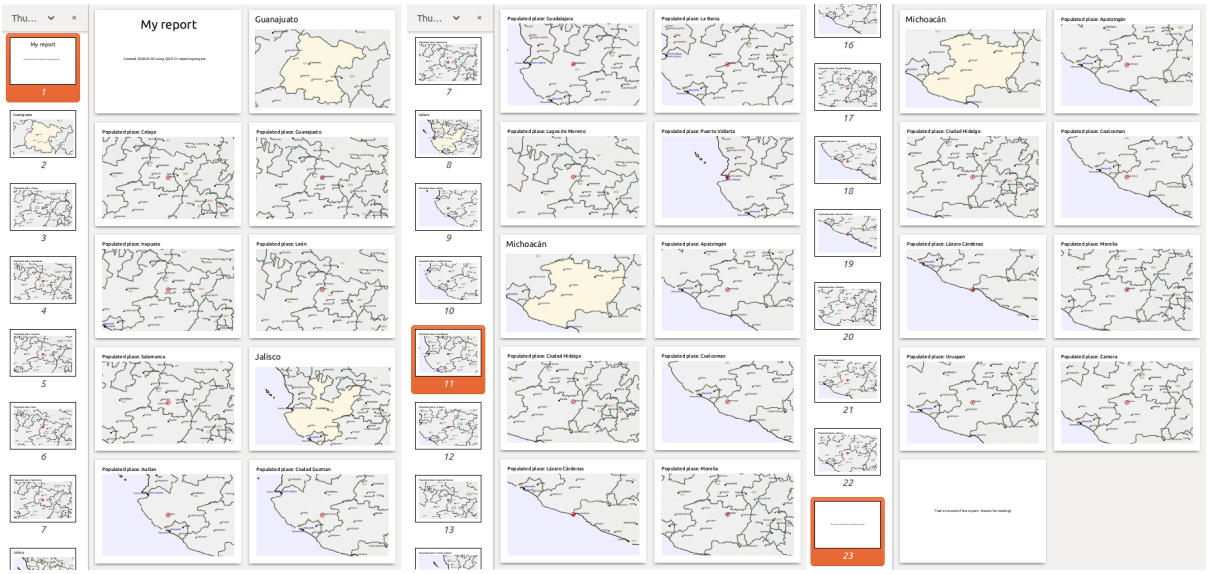
Also mehr oder weniger ein Atlas, aber mit einer Kopf- und Fußseite.

Um die Sache noch interessanter zu machen, fügen wir unserer Statusgruppe einen Unterabschnitt hinzu. Dazu wählen wir zunächst die Feldgruppe *Admin Level 1* in der Berichtsorganisation aus, klicken dann auf die Schaltfläche Abschnitt hinzufügen und fügen einen neuen *Feldgruppenabschnitt* hinzu:



Bei der Iteration über die Objekte eines *Feldgruppenabschnitts* werden die Objekte so gefiltert, dass sie dem

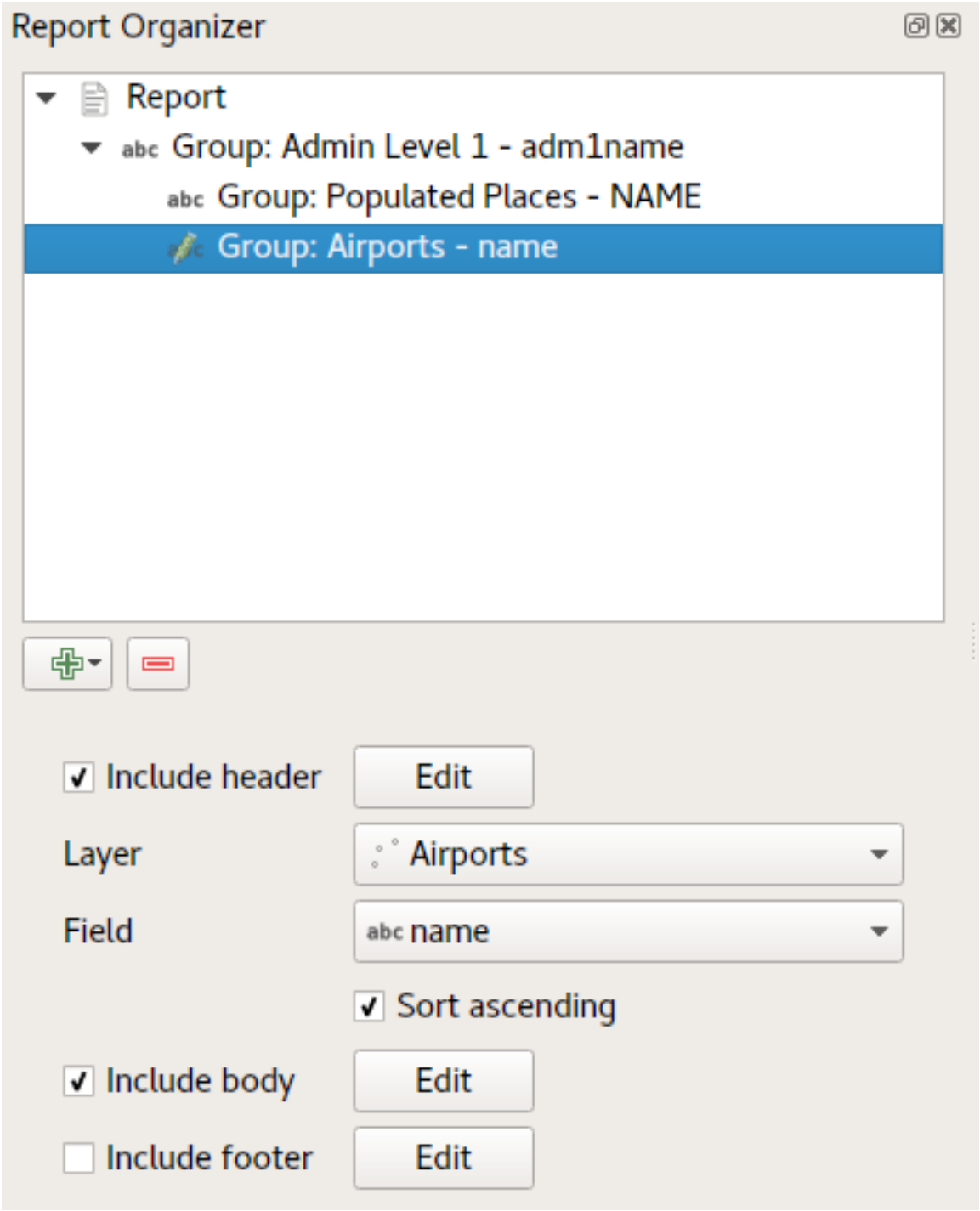
definierenden Feld der übergeordneten Gruppe entsprechen (in diesem Fall `adm1name`). In diesem Fall durchläuft der hinzugefügte Unterabschnitt den Layer *Populated Places*, der für jeden gefundenen Ort einen eigenen Abschnitt enthält. Der Clou dabei ist, dass der Layer *Populated Places* ein Attribut mit dem gleichen Namen wie das definierende Feld im übergeordneten Layer, *adm1name*, hat, das jeden Ort mit dem Zustand kennzeichnet, in dem er sich befindet (wenn Sie Glück haben, sind Ihre Daten bereits so strukturiert - wenn nicht, führen Sie den Verarbeitungsalgorithmus Attribute nach Ort verschneiden aus und erstellen Sie Ihr eigenes Feld). Wenn wir diesen Bericht exportieren, holt sich QGIS den ersten Status aus dem Layer *Admin Level 1* und durchläuft dann alle *Populated Places* mit einem passenden *adm1name* Wert. Das Ergebnis ist wie folgt:



Hier haben wir ein Grundgerüst für die Gruppe „Populated Places“ erstellt, einschließlich einer Karte des Ortes und einer Tabelle mit einigen Ortsattributen. Unser Bericht besteht nun aus einer Kopfzeile, einer Seite für den ersten Bundesstaat, gefolgt von einer Seite für jeden besiedelten Ort in diesem Bundesstaat, dann die restlichen Bundesstaaten mit ihren besiedelten Orten und schließlich die Fußzeile des Berichts. Wenn wir eine Kopfzeile für die Gruppe „Populated Places“ hinzufügen würden, würde diese unmittelbar vor der Auflistung der besiedelten Orte für jedes Bundesland eingefügt, wie in der folgenden Abbildung gezeigt.

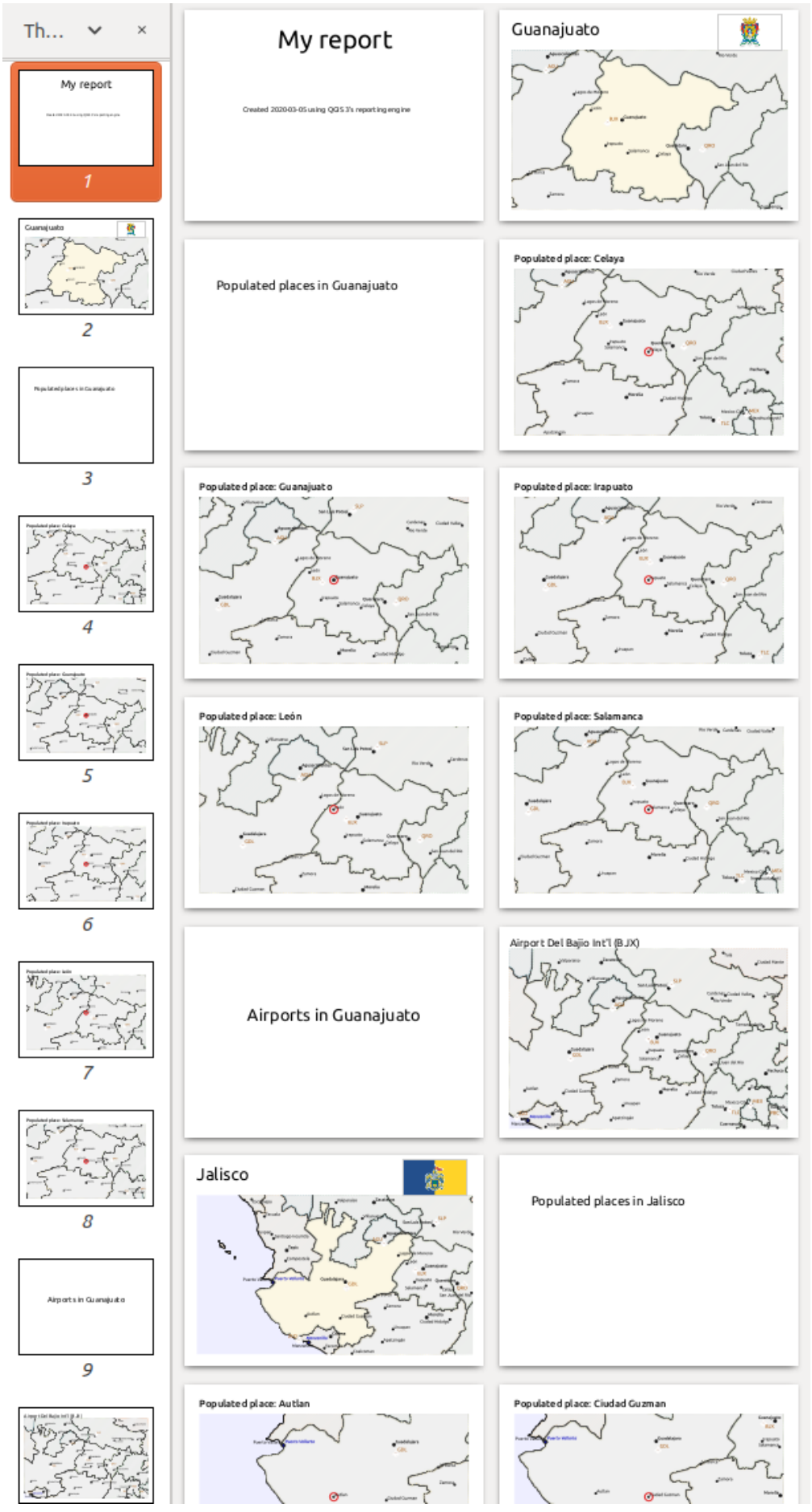
In ähnlicher Weise würde eine Fußzeile für die Gruppe „Populated Places“ eingefügt, nachdem der letzte Ort für jeden Staat aufgenommen wurde.

Zusätzlich zu den verschachtelten Unterabschnitten können Unterabschnitte in einem Bericht auch nacheinander eingefügt werden. Wenn wir der Gruppe *Admin Level 1* für *Airports* einen zweiten Unterabschnitt hinzufügen, dann (wenn der Layer *Airports* ein Attribut *adm1name* hat, das ihn mit der übergeordneten Gruppe verknüpfen kann) wird unser Bericht zuerst ALLE bewohnten Orte für jeden Staat auflisten, gefolgt von allen Flughäfen innerhalb dieses Staates, bevor wir mit dem nächsten Staat fortfahren.



Wichtig ist hier, dass unsere Gruppe *Airports* eine Untergruppe der Gruppe *Admin Level 1* ist - nicht der Gruppe *Populated Places*.

In diesem Fall wäre unser Bericht wie folgt aufgebaut (beachten Sie, dass auch Statuskennzeichen enthalten sind - das Verfahren zum Hinzufügen von objektspezifischen Bildern auf diese Weise wird weiter unten beschrieben):


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

Einfügen von Bildern in einen Bericht

Bilder können in Berichten sehr nützlich sein, und QGIS erlaubt Bilder sowohl in den statischen als auch in den dynamischen Teilen eines Berichts. Bilder werden auf die gleiche Weise hinzugefügt wie bei Standard-Drucklayouts, und für die statischen Berichtsteile (und statische Bilder in dynamischen Teilen) gibt es nicht mehr zu tun.

Wenn Sie jedoch Illustrationen wünschen, die auf die Objekte des Berichts zugeschnitten sind, muss Ihr Layer über ein Attribut verfügen, mit dem Sie das einzubeziehende Bild definieren können.

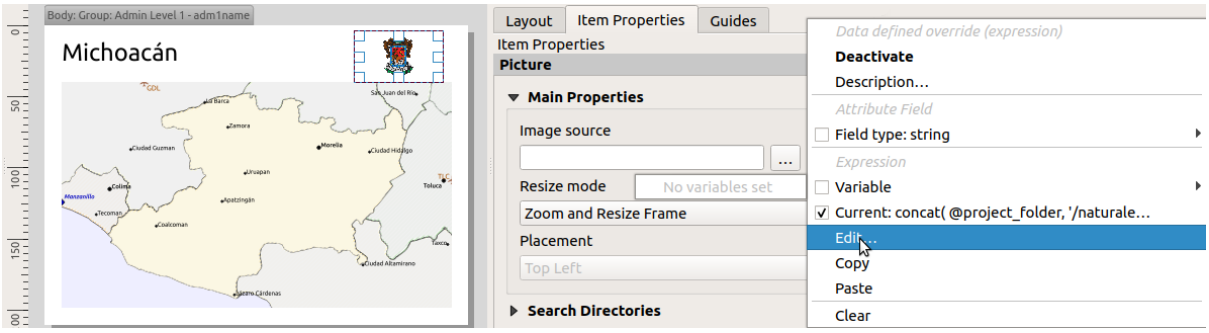
QGIS ist auf vollständige Dateipfade für Bilder in Berichten angewiesen.

Bei dynamischen Bildern fügen Sie zunächst wie gewohnt ein Bild in den Body-Teil der Gruppe ein. In den *Elementeigenschaften* des Bildes setzen Sie die *Bildquelle* mit der Schaltfläche Datendefinierte Übersteuerung, und wählen entweder ein Attribut, das den absoluten Pfad der Bilder enthält, oder *Bearbeiten…* (um einen Ausdruck einzugeben, der den absoluten Bildpfad erzeugt).

Nachfolgend ein Beispielausdruck, der die String-Verkettung verwendet, um den absoluten Pfad zu den Bildern anzugeben, wobei das Verzeichnis, in dem sich die Projektdatei befindet, (`@project_path`) und ein Attribut (`adm1name`) verwendet werden, aus dem der Dateiname generiert wird (in diesem Fall durch Umwandlung des Strings im Attribut `adm1name` in Großbuchstaben und Anhängen von ‚_flag.png'):

```
concat(@project_folder, '/naturalearth/pictures/' ,
       upper("adm1name"), '_flag.png')
```

Das bedeutet, daß sich die Bilder im Unterverzeichnis `naturalearth/pictures` des Projektdateiverzeichnisses befinden.



Hervorheben des aktuellen Berichtsobjekts in einer Karte

Im obigen Bericht werden die Objekte des Berichts in den Karten durch Hervorhebung (Staat) und Kreise (besiedelte Orte) herausgestellt. Um die Objekte in den Karten hervorzuheben (abgesehen davon, dass sie in der Mitte der Karten platziert werden), müssen Sie den Stil durch einen Vergleich zwischen seiner `@id` und der `@atlas_featureid` definieren, wie bei Atlanten.

Wenn Sie z.B. für das jeweilige Berichts-Objekt eine dickere Linie / Umrandung als für die anderen Objekte verwenden möchten, können Sie die Linienstärke definieren:

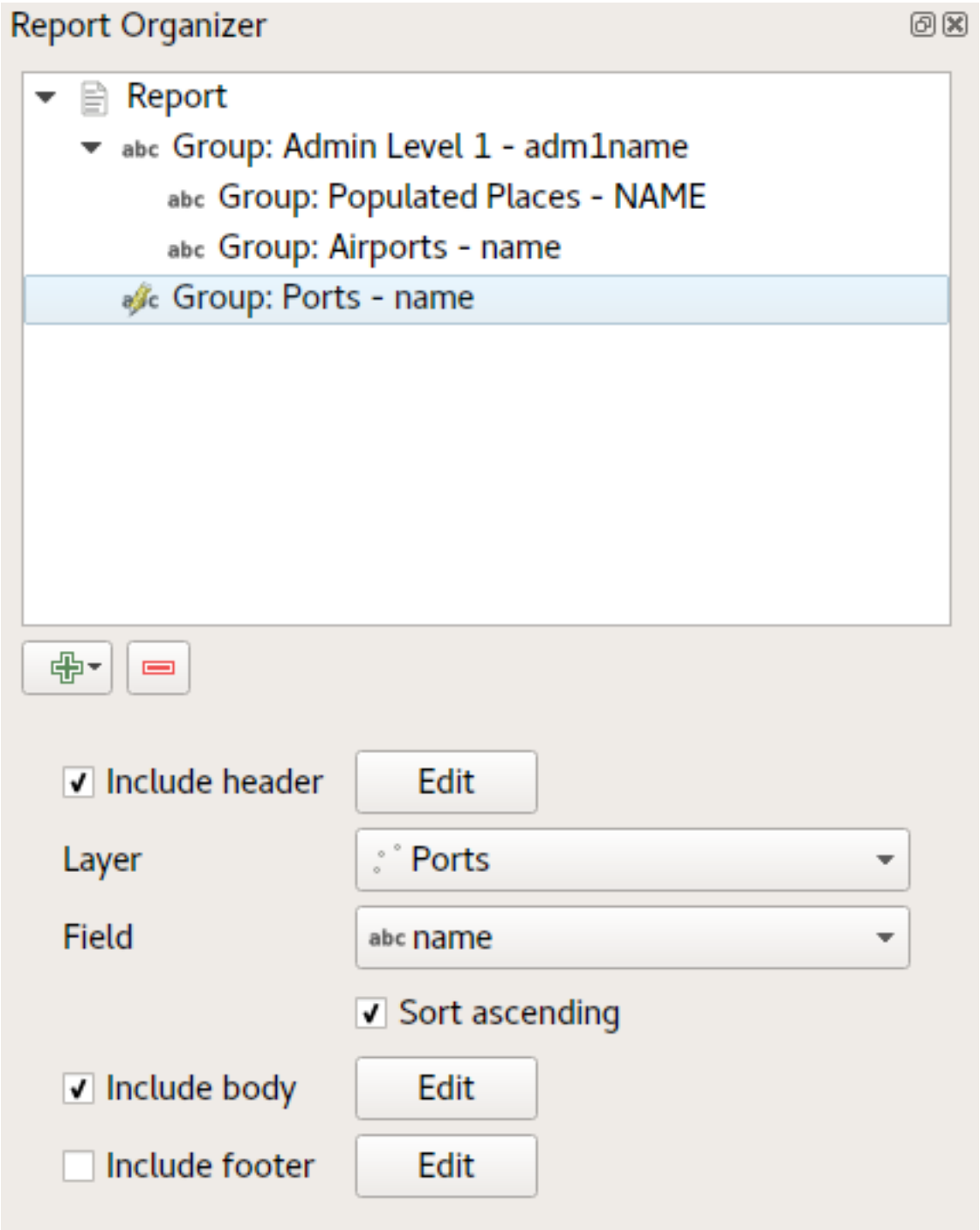
```
if($id=@atlas_featureid, 2.0, 0.1)
```

Das Berichts-Objekt erhält einen 2 Einheiten breiten Polygonumriss, während alle anderen Objekte eine 0,1 Einheiten breite Linie erhalten. Es ist auch möglich, die Farbe zu definieren (nicht-transparentes dunkles Magenta für das Berichtsmerkmal und halbtransparentes Hellgrau für die anderen Objekte):

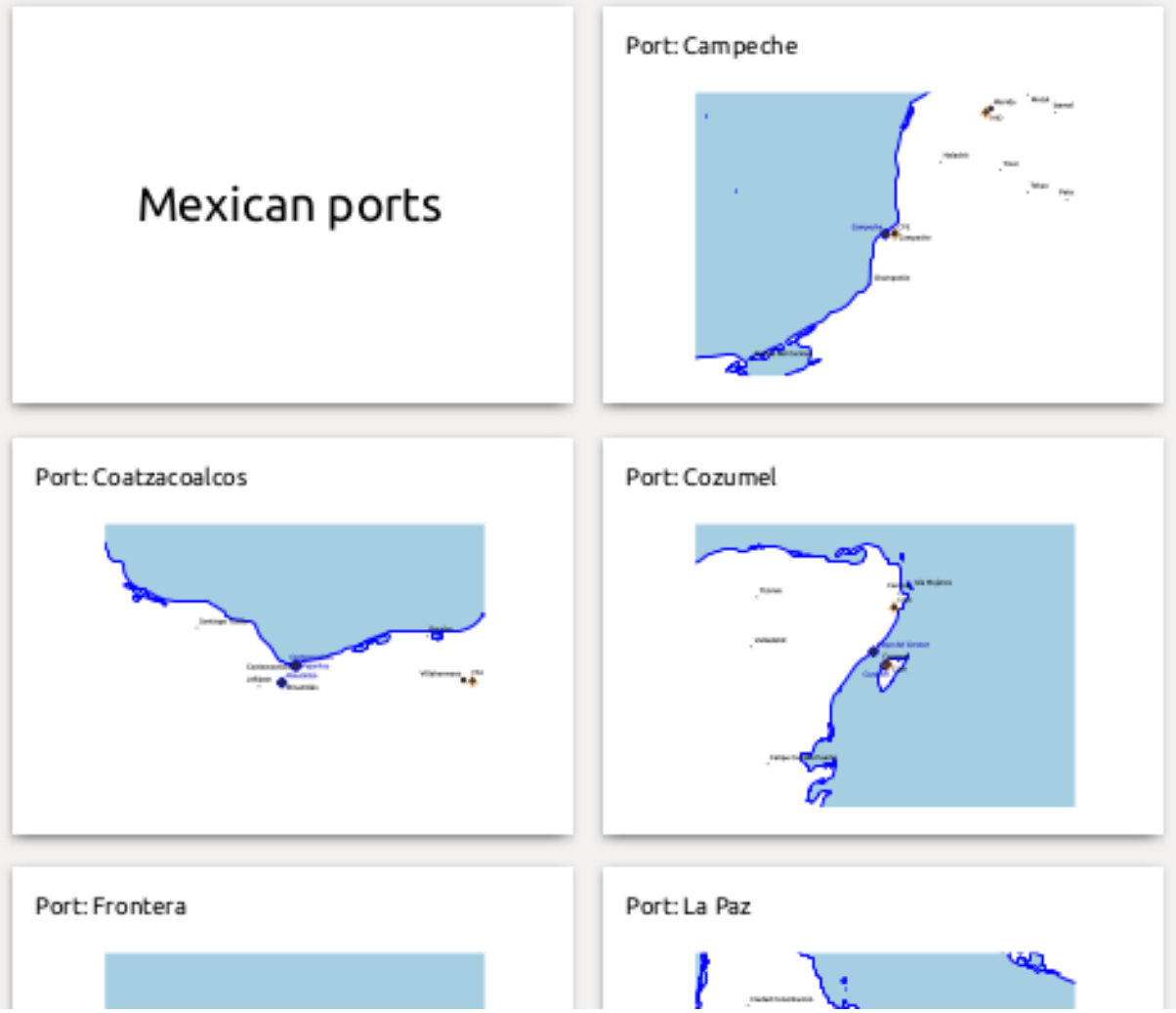
```
if($id=@atlas_featureid, '#FF880088', '#88CCCCCC')
```

Weitere Gruppen der obersten Ebene

Die Kombination von verschachtelten und aufeinanderfolgenden Abschnitten sowie von Abschnittsüberschriften und -fußzeilen ermöglicht eine große Flexibilität. Im folgenden Bericht fügen wir zum Beispiel eine weitere Feldgruppe als untergeordnetes Element des Hauptberichts für den Layer *Ports* hinzu. Nach der Auflistung der Bundesstaaten mit ihren bewohnten Orten und Flughäfen erhalten wir nun eine zusammenfassende Liste aller Häfen in der Region:



Dies führt dazu, dass der letzte Teil unseres Berichts wie folgt exportiert wird:



18.4.4 Exporteinstellungen

Wenn Sie einen Bericht exportieren (*Bericht* ► *Bericht als Bilder exportieren… / SVG… / PDF…*), werden Sie nach einem Dateinamen gefragt, und dann haben Sie die Möglichkeit, die Exporteinstellungen so anzupassen, dass Sie die gewünschte Ausgabe erhalten.

Wie Sie sehen, sind die Berichte in QGIS extrem leistungsfähig und flexibel.

Bemerkung

Die aktuellen Informationen wurden aus einem Blog von North Road übernommen: Exploring Reports in QGIS 3.0 - the Ultimate Guide!

KAPITEL 19

Arbeiten mit OGC / ISO-Protokollen

Das Open Geospatial Consortium (OGC) ist eine internationale Organisation mit mehr als 300 Mitgliedern aus kommerziellen, behördlichen Bereichen, aus der Forschung sowie aus Non-Profit Organisationen Vereinen. Die Mitglieder entwickeln und implementieren Standards für den Austausch räumlicher Daten, GIS-Datenprocessing und standardisierte Bereitstellung von Geodaten.

Das OGC beschreibt ein grundlegendes Datenmodell für geografische Objekte und entwickelt eine wachsende Zahl von Spezifikationen, um spezifische Anforderungen an interoperable Standort- und Geodaten-Technologien, einschließlich GIS, zu erfüllen. Weitere Informationen finden Sie unter https://www.ogc.org/.

Wichtige von QGIS unterstützte OGC Spezifikationen sind:

- **WMS** — Web Map Service (*WMS/WMTS Client*)
- **WMTS** — Web Map Tile Service (*WMS/WMTS Client*)
- **WFS** — Web Feature Service (*WFS und WFS-T Klient*)
- **WFS-T** — Web Feature Service - Transactional (*WFS und WFS-T Klient*)
- **WCS** — Web Coverage Service (*WCS Client*)
- **WPS** — Web Processing Service
- **CSW** — Catalog Service for the Web
- **SFS** — Simple Features for SQL (*PostgreSQL Database*)
- **GML** — Geography Markup Language
- **SensorThings API** — SensorThings API (*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*).

Sie können Ihre Karten und Daten auch über die Protokolle WMS, WMTS, WFS, WFS-T und WCS teilen, indem Sie einen Webserver benutzen, auf dem QGIS Server, UMN MapServer oder GeoServer installiert ist.

19.1 WMS/WMTS Client

19.1.1 Übersicht über die WMS-Unterstützung

Derzeit kann QGIS als WMS-Klient eingesetzt werden. Es unterstützt die Versionen 1.1, 1.1.1 und 1.3 der WMS-Server. Gut getestet wurden die öffentlich verfügbaren Server wie beispielsweise DEMIS.

WMS-Server liefern Daten aufgrund einer Anfrage eines Klienten (hier QGIS) als Rasterbild aus. Dabei spielen Ausdehnung, Anzahl der angefragten Layer, Symbolisierungen und Transparenz eine Rolle. Der WMS-Server holt die benötigten Daten dann aus seiner Datenquelle hervor, rendert diese in eine Rasterkarte und sendet das fertige Bild zurück zum Klienten. Das für QGIS typische Rasterformat ist in aller Regel JPEG oder PNG.

WMS ist ein komplett auf Übertragung ausgelegter Dienst (REST = Representational State Transfer). Daraus resultiert die Tatsache, dass die von QGIS generierte URL für das Bild auch in einem Browser eingesetzt werden kann. Das Resultat dieser Anfrage sieht in der Regel genauso aus wie in QGIS. Das ist besonders hilfreich, wenn es beim Einsatz von WMS Probleme geben sollte. Da es sehr viele unterschiedle WMS-Server-Anbieter am Markt gibt (und alle die WMS-Spezifikation etwas unterschiedlich interpretieren), ist eine Überprüfung im Browser sehr hilfreich.

WMS-Layer können sehr einfach hinzugefügt werden, solange man die URL des Servers kennt, eine Verbindung über HTTP zu diesem Server besteht und der angefragte Server auch HTTP versteht.

Darüber hinaus speichert QGIS Ihre WMS-Antworten (d.h. Bilder) für 24 Stunden, solange die GetCapabilities-Anfrage nicht ausgelöst wird. Die GetCapabilities-Anfrage wird jedes Mal ausgelöst, wenn die Schaltfläche *Verbinden* im Dialog *WMS/WMTS* verwendet wird, um die WMS-Server-Fähigkeiten abzufragen. Dies ist ein automatisches Objekt, um die Ladezeit des Projekts zu optimieren. Wenn ein Projekt mit einem WMS Layer gespeichert wird, werden die entsprechenden WMS Kacheln beim nächsten Öffnen des Projekts aus dem Cache geladen, sofern sie nicht älter als 24h sind.

19.1.2 Überblick über die WMTS Unterstützung

|qg| kann auch als WMTS Client fungieren. WMTS ist ein OGC Standard zum Bereitstellen von Tile Sets von räumlichen Daten. Dies ist ein schnellerer und effizienterer Weg als die Bereitstellung über einen WMS weil bei WMTS die Tile Sets vorgeneriert werden und der Client nur die Übermittlung von Tiles, nicht Ihre Erstellung abfragt. Ein WMS beinhaltet typischerweise sowohl die Erstellung als auch die Übermittlung der Daten. Ein bekanntes Beispiel eines nicht-OGC Standards zum Darstellen von gekachelten räumlichen Daten ist Google Maps.

Um die Daten in einer Vielzahl von Maßstäben je nach Anforderung darzustellen werden die WMTS Tile Sets in mehreren verschiedenen Maßstäben erstellt und dem GIS Client zur Abfrage bereitgestellt.

Das Diagramm veranschaulicht das Konzept der Tile Sets:

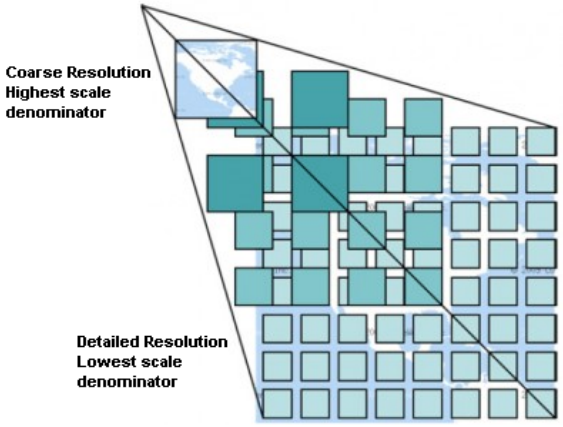


Abb. 19.1: Konzept der WMTS Tile Sets

Die zwei Typen von WMTS Schnittstellen die QGIS unterstützt sind die über Key-Value-Pairs (KVP) und RESTful. Diese zwei Schnittstellen sind unterschiedlich und Sie müssen diese für QGIS unterschiedlich spezifizieren.

1. Um einen **WMTS KVP** Service anzubinden muss ein QGIS Benutzer die WMS/WMTS Schnittstelle öffnen und die folgenden Zeichenkette zu der URL des WMTS Tile Service hinzufügen:

   ```
   "?SERVICE=WMTS&REQUEST=GetCapabilities"
   ```

   Ein Beispiel für diese Art von Adresse ist:

   ```
   https://opencache.statkart.no/gatekeeper/gk/gk.open_wmts?service=WMTS&
   ↪request=GetCapabilities
   ```

   Zu Testzwecken funktioniert der topo2 Layer in diesem WMTS gut. Indem man diese Zeichenfolge hinzufügt gibt man an dass ein WMTS Web Service anstatt eines WMS Service benutzt werden soll.

2. Der **RESTful WMTS** Service erfordert eine andere Form, eine einfache URL. Das von der OGC empfohlene Format ist:

   ```
   {WMTSBaseURL}/1.0.0/WMTSCapabilities.xml
   ```

   Anhand dieses Formats können Sie erkennen, dass es sich um eine RESTful-Adresse handelt. Auf ein RESTful WMTS wird in QGIS zugegriffen, indem man einfach die Adresse im WMS Setup in das URL-Feld des Formulars einträgt. Ein Beispiel für diese Art von Adresse für eine österreichische Basiskarte ist:

   ```
   https://maps.wien.gv.at/basemap/1.0.0/WMTSCapabilities.xml
   ```

Bemerkung

Es gibt immer noch einige alte WMS-C-Dienste. Diese sind ähnlich wie WMTS (d.h. sie dienen dem selben Zweck, funktionieren aber anders). Sie können Sie genau so verwalten wie WMTS-Dienste. Fügen Sie einfach `?tiled=true` an das Ende der URL an. Siehe https://wiki.osgeo.org/wiki/Tile_Map_Service_Specification für weitere Informationen über diese Spezifikation.

Wenn Sie WMTS lesen können Sie auch an WMS-C denken.

19.1.3 WMS/WMTS Server auswählen

Wenn Sie das Objekt WMS/WMTS zum ersten Mal in QGIS verwenden, sind noch keine Server definiert.

Dann müssen Sie Verbindungen zu den Servern, auf die Sie zugreifen wollen, definieren:

1. Gehen Sie zum Tab *WMS/WMTS* des Dialogs *Datenquellenverwaltung* indem Sie:

   - entweder auf die Schaltfläche Datenquellenverwaltung öffnen klicken (oder Tastenkürzel `Strg+L`) und dann den Tab aktivieren
   - oder auf die Schaltfläche WMS/WMTS-Layer hinzufügen in der *Layerverwaltungswerkzeugleiste* klicken
   - oder im Menü *Layer* ► *Layer hinzufügen* ► den Menüpunkt *WMS/WMTS-Layer hinzufügen…* anwählen

2. Klicken Sie auf *Neu* im Tab *Layer*. Der Dialog *Neue WMS/WMTS-Verbindung anlegen* erscheint.

Tipp

Ein Rechtsklick auf den Eintrag *WMS/WMTS* innerhalb des *Bedienfelds Browser* und die Auswahl *Neue Verbindung…* öffnet ebenfalls den Dialog *Neue WMS/WMTS-Verbindung anlegen*.

3. Geben Sie dort die Verbindungsparameter zu Ihrem gewünschten WMS-Server ein, so wie im Folgenden beschrieben:

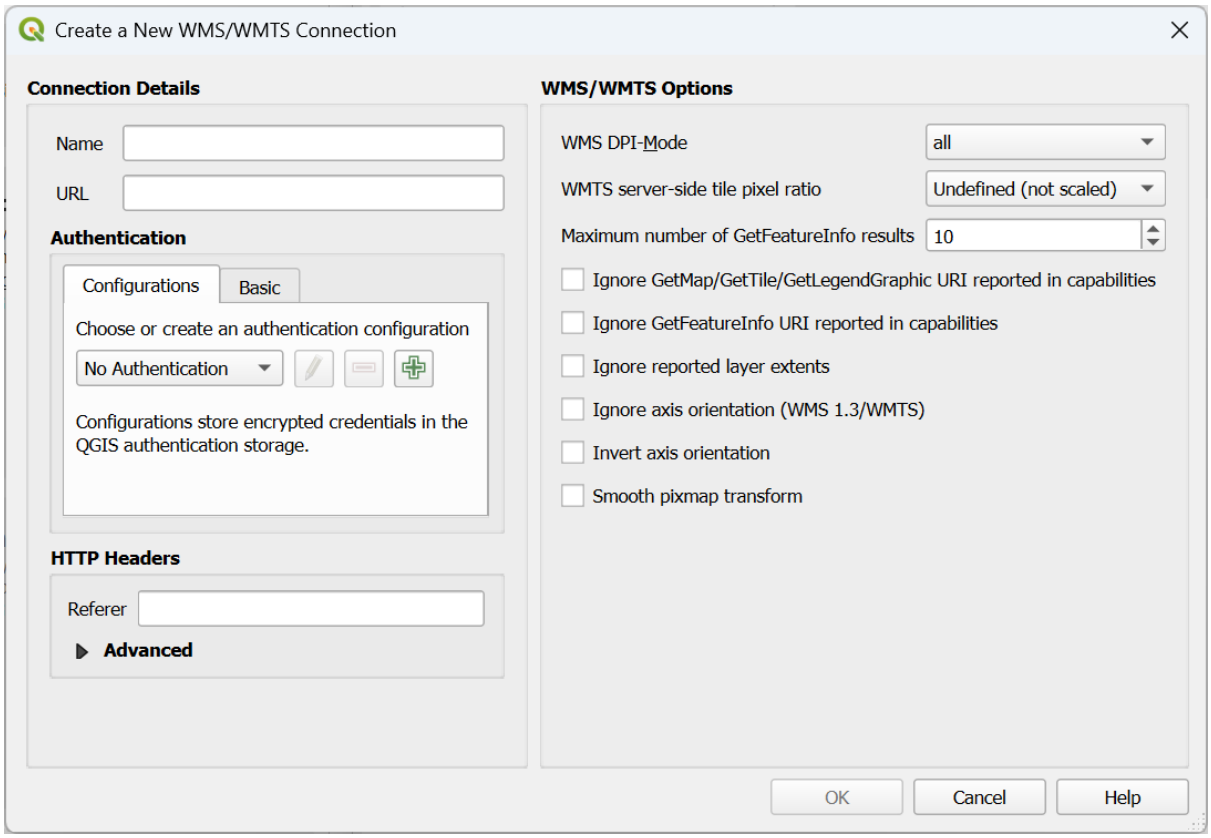


Abb. 19.2: Verbindung mit einem WMS-Server anlegen

- *Name*: Der Name der Verbindung. Der Name wird in der Auswahlliste der Serververbindungen benutzt, so dass Sie die WMS-Server unterscheiden können.
- *URL*: DIe URL des Servers. Es muss sich um einen auflösbaren Hostnamen handeln – im selben Format, wie Sie es für eine Telnetverbindung benutzen würden oder um einen Host anzupingen, d.h. nur die Basis-URL. Ihre URL sollte also keine Teile wie `request=GetCapabilities` oder `version=1.0.0` enthalten.
- *Authentifizierung* (optional): Benutzen Sie eine *gespeicherte Konfiguration* oder eine Basic-Authentifizierung mit *Benutzername* und *Passwort*.

> **Warnung**
>
> Durch die Eingabe von **Benutzername** und **Passwort** auf der Registerkarte *Authentifizierung* werden ungeschützte Anmeldedaten in der Verbindungskonfiguration gespeichert. Diese **Anmeldeinformationen sind sichtbar**, wenn Sie z.B. die Projektdatei mit jemandem teilen. Daher ist es ratsam, Ihre Anmeldeinformationen stattdessen in einer *Authentifizierungskonfiguration* zu speichern (Registerkarte *Konfigurationen*). Siehe *Authentifizierungssystem* für weitere Details.

- HTTP *Referer*
- *WMS DPI-Mode*: Verfügbare Optionen sind **alle**, **aus**, **QGIS**, **UMN** und **GeoServer**.
- *WMTS serverseitige Pixelauflösung*: Ermöglicht beim Rendern von WMTS Layern die Vergrößerung oder Verkleinerung der Kacheln auf der Grundlage der DPI des Gerätebildschirms. Verfügbare Optionen sind *Undefiniert (nicht skaliert)*, *Standard (96 DPI)* und *Hoch (192 DPI)*.

- *Maximale Anzahl von GetFeatureInfo-Ergebnissen*: Gibt einen Standardwert für die maximale Anzahl von Ergebnissen an, die pro Layer von einer GetFeatureInfo-Anfrage unter Verwendung dieser Verbindung zurückgegeben werden (siehe Parameter FEATURE_COUNT 1). Der Standardwert ist `10`. Setzen Sie den Wert auf `0`, um den Standardwert des Servers (normalerweise `1`) zu verwenden: Der Parameter FEATURE_COUNT wird dann nicht zur Anfrage hinzugefügt.
- ☐ *Gemeldete GetMap/GetTile/GetLegendGraphic-URI aus Diensteigenschaften ignorieren*: wenn aktiviert, wird der angegebene URI aus dem obigen Feld *URL* verwendet.
- ☐ *Gemeldete GetFeatureInfo-URI ignorieren*: falls Sie das anhaken, wird die im *URL* Feld eingegebene URI benutzt.
- ☐ *Berichtete Layerausmaße ignorieren*: die von Rasterlayern berichteten Layerausmaße können kleiner sein, als die Fläche, die tatsächlich gerendert wird (dies ist insbesondere bei WMS-Servern der Fall, die Symbole enthalten, die mehr Platz brauchen als die Daten selbst). Haken Sie diese Option an, um zu verhindern, dass Rasterlayer am Rand abgeschnitten werden, was zu abgeschnittenen Symbolen in den Randbereichen dieser Layer führen kann
- ☐ *Achsenorientierung ignorieren (WMS 1.3/WMTS)*
- ☐ *Achsenorientierung invertieren*
- ☐ *Pixmap-Transformation glätten*

4. Klicken Sie auf *OK*

Sobald die neue WMS/WMTS-Serververbindung erstellt wurde, bleibt sie für zukünftige QGIS-Sitzungen erhalten. Beachten Sie, dass es auch möglich ist, die Verbindungsparameter aus einer `.XML`-Datei zu *Laden* oder in einer solchen zu *Speichern*.

Wenn Sie einen Proxyserver konfigurieren müssen, um WMS-Dienste aus dem Internet zu beziehen, können Sie Ihren Proxyserver in den Optionen hinzufügen. Wählen Sie im Menü *Einstellungen* ► *Optionen* und klicken Sie auf den Tab *Netzwerk*. Dort können Sie Ihre Proxyeinstellungen einfügen und sie durch Anhaken von ☑ *Proxy für Webzugriff verwenden* aktivieren. Stellen Sie sicher, dass Sie den richtigen *Proxytyp* [...▾] aus der Auswahl eingestellt haben.

19.1.4 WMS/WMTS Layer laden

Sobald Sie einmal erfolgreich Ihre Parameter eingegeben haben, können Sie die Schaltfläche *Verbinden* benutzen, um die Eigenschaften des ausgewählten Servers herunterzuladen. Dazu gehören die Bildkodierung, die Layer und ihre Stile sowie die Bezugssysteme. Da es sich um eine Netzwerkanfrage handelt, hängt die Antwortgeschwindigkeit von der Qualität Ihrer Netzwerkverbindung zum WMS-Server ab. Während Daten vom WMS-Server heruntergeladen werden, wird der Downloadfortschritt in der unteren linken Ecke des QGIS-Hauptfensters dargestellt.

Ihr Bildschirm sollte nun so ähnlich wie im Bild Abb. 19.3 aussehen, das die Antwort eines WMS-Servers darstellt.

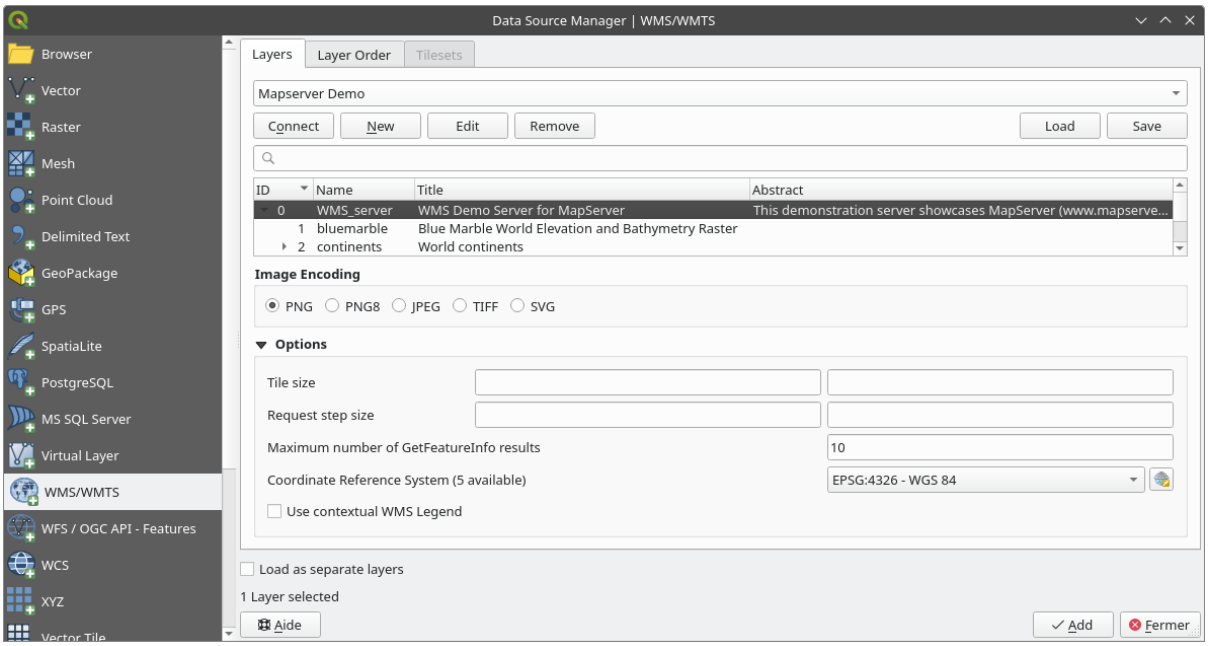


Abb. 19.3: Daten von einem WMS-Server hinzufügen, mit Auflistung der verfügbaren Layer

Der obere Teil des Tabs *Layer* im Dialog zeigt eine Baumstruktur, die Layergruppen mit ihren enthaltenen Layern und dem (bzw. den) zugehörigen Stil(en) enthält, die vom Server angeboten werden. Jeder Eintrag wird durch:

- eine *ID*
- einen *Namen*
- einen *Titel*
- und eine *Zusammenfassung* identifiziert.

Sie können die Liste mit dem Suchfeld rechts oben filtern.

Bildkodierung

Der *Bildkodierung* Bereich listet die Formate die sowohl vom Client als auch vom Server unterstützt werden auf. Wählen Sie eines abhängig von Ihren Anforderungen an die Bildqualität aus.

Tipp

Bildkodierung

In der Regel bieten WMS-Server JPEG oder PNG als Bildkodierung an. JPEG hat eine bildverschlechternde Kompression, während PNG zumeist die Qualität der ursprünglichen Rasterdaten widerspiegelt.

Verwenden Sie JPEG wenn Sie damit rechnen das die WMS Daten photographischen Charakter haben und/oder Sie ein Verlust an Bildqualität nicht stört. Dieser Kompromiss reduziert typischerweise die Datentransferbedingungen um das fünffache verglichen mit PNG.

Verwenden Sie PNG wenn Sie eine genaue Wiedergabe der Originaldaten erzielen wollen und die erhöhten Datentransferbedingungen Sie nicht stören.

Optionen

Der Dialogbereich Optionen enthält Möglichkeiten, die WMS-Abfrage zu konfigurieren. Sie können folgendes einstellen:

- *Kachelgröße* wenn Sie die Kachelgröße (bspw., 256x256) einstellen wollen, um die WMS-Abfrage in viele einzelne Abfragen aufzuteilen.

- *Abfrageschrittgröße*: Wenn Sie den Effekt der abgeschnittenen Beschriftungen an den Kachelrändern verringern wollen, erzeugt eine Erhöhung der Schrittgröße größere Abfragen, weniger Kacheln und weniger Ränder. Der Standardwert ist 2000.
- Die *Maximale Anzahl von GetFeatureInfo-Ergebnissen*: gibt die maximale Anzahl von Ergebnissen an, die von einer GetFeatureInfo-Anfrage für die geladenen Layer zurückgegeben werden. Ermöglicht es, den *Standardwert der Verbindung* für bestimmte Layer zu überschreiben.
- Grundsätzlich kann jeder WMS Layer mehreren Koordinatensystemen haben, je nach den Möglichkeiten des WMS Servers. Wenn Sie einen WMS aus der Liste auswählen, erscheint ein Feld mit der vom Webserver bereitgestellten Standardprojektion. Drücken Sie das Widget Koordinatensystem, um die Standardprojektion des WMS durch ein anderes vom WMS-Server unterstütztes System zu ersetzen.

  Sie sehen einen Dialog ähnlich wie im Bild Abb. 6.4. Der Hauptunterschied zur WMS-Version des Dialogs ist, dass er nur KBS anzeigt, die vom WMS-Server unterstützt werden.
- Schließlich können Sie noch *Kontextbezogene WMS-Legende verwenden* anhaken, falls der WMS-Server dies unterstützt. Dann wird nur die für Ihren aktuellen Kartenausschnitt relevante Legende gezeigt, wodurch Legendenelemente für Objekte ausgeblendet werden, die Sie im aktuellen Kartenausschnitt gar nicht sehen.

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.

Layerreihenfolge

Der Tab *Layerreihenfolge* listet die ausgewählten Layer des verbundenen WMS-Server auf.

Die WMS-Layers werden vom Server in der Reihenfolge übereinander gerendert, wie sie in der Liste im Tab *Layer* erscheinen, von oben nach unten. Möchten Sie die Reihenfolge ändern, können Sie die Schaltflächen *Auf* und *Ab* im Tab *Layerreihenfolge* verwenden.

Transparenz

Die Einstellung für die *Globale Deckkraft* in den *Layereigenschaften* ist hartkodiert, um stets aktiv zu sein, falls verfügbar.

19.1.5 Tilesets

Bei der Verwendung von WMTS-Diensten (Cached WMS) können Sie die vom Server vorgegebene Registerkarte *Tilesets* durchsuchen. Zusätzliche Informationen wie Kachelgröße, Formate und unterstützte KBS sind in dieser Tabelle aufgeführt.

```
# example of WMTS service

https://opencache.statkart.no/gatekeeper/gk/gk.open_wmts?service=WMTS&
↪request=GetCapabilities
```

Wenn Sie einen Layer zum Laden auswählen, ist es auch möglich, eine *Interpretationsmethode* anzuwenden, die den Layer in einen Raster-Layer vom Typ „Single Band Float“ umwandelt, der mit den in QGIS üblichen *Raster-Renderern* gestaltet werden kann.

In Kombination mit diesem Objekt können Sie den Schieberegler für die Kachelskala verwenden, indem Sie *Ansicht* ► *Bedienfelder* (oder *Einstellungen* ► *Bedienfelder*) und dann *Kachelmaßstab* wählen. Dies zeigt Ihnen die verfügbaren Skalen des Kachelservers mit einem angedockten Schieberegler.

19.1.6 Das Objekte abfragen Werkzeug

Nachdem Sie einen Layer von einem WMS-Server geladen haben, können Sie die Layer mit dem Werkzeug Objekte Abfragen abfragen, sofern der WMS-Server diese Funktion unterstützt. Ein Klick auf einen Pixel stellt dann eine Abfrage an den WMS-Server für diesen Pixel. Das Ergebnis wird in Textform geliefert. Die Formatierung hängt von dem jeweilig verwendeten WMS-Server ab.

Formatauswahl

Wenn durch den Server mehrere Formate unterstützt werden wird dem Objekte abfragen Dialog automatisch eine Kombobox mit den unterstützten Formaten hinzugefügt und das ausgewählte Format kann im Projekt für den Layer gespeichert werden.

GML Formatunterstützung

Das Objekte abfragen Werkzeug unterstützt WMS Server Response (GetFeatureInfo) im GML Format (es wird Feature in der QGIS GUI in diesem Zusammenhang genannt). Wenn das „Feature“ Format vom Server unterstützt wird und ausgewählt ist, sind die Ergebnisse des Objekte abfragen Werkzeugs Vektorobjekte, wie bei einem normalen Vektorlayer. Wenn ein einzelnes Objekt im Baum ausgewählt wird wird es in der Karte hervorgehoben und kann in die Zwischenablage kopiert werden und in einen anderen Vektorlayer eingefügt werden. Sehen Sie sich für die Unterstüzung von GetFeatureInfo im GML Format das Beispielsetup des UMN Mapservers unten an.

```
# 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 Anzeigen von Eigenschaften

Sobald Sie einen WMS-Server hinzugefügt haben, können Sie seine Eigenschaften anzeigen, indem Sie mit der rechten Maustaste auf ihn in der Legende klicken und *Eigenschaften* wählen. Die Eigenschaften von WMS/WMTS-Layern sind den Eigenschaften von Raster-Layern sehr ähnlich. Eine ausführliche Beschreibung finden Sie unter *Der Eigenschaften-Dialog von Rasterlayern*. Es gibt jedoch einige Unterschiede, die im Folgenden erläutert werden.

Informationseigenschaften

Reiter Metadaten

Der Reiter *Metadaten* im Kontextmenü zeigt eine Vielzahl von Informationen über den WMS-Server. Diese Infos sind dem Capabilities-Dokument des Servers entnommen. Viele Definitionen können reduziert werden indem man den WMS Standard liest (siehe OPEN-GEOSPATIAL-CONSORTIUM *Literatur und Internetreferenzen*), hier sind dazu einige praktische Definitionen:

- **Servereigenschaften**
  - **WMS Version**– Die WMS-Version, die vom Server unterstützt wird.
  - **Bildformate** — Eine Liste der MIME-Typen mit denen der Server antworten kann. QGIS unterstützt jedes Format, welches die darunterliegende Bibliothek QT unterstützt, mindestens aber `image/png` und `image/jpeg`.
  - **Abfrageformate** — Eine Liste der MIME-Typen mit denen der Server auf Pixel-Abfragen antworten kann. Derzeit wird von QGIS nur der Typ `text-plain` unterstützt.
- **Layereigenschaften**
  - **Ausgewählt** — Gibt an, ob dieser Layer während des Hinzufügens des Server ausgewählt war.
  - **Sichtbar** — Gibt an, ob der Layer in der Legende angezeigt wird oder nicht. (noch nicht verwendet in der aktuellen Version von QGIS.)
  - **Kann abfragen** — Gibt an, ob der Layer auf Abfragen Ergebnisse zurückgibt.
  - **Kann Transparenz** — Gibt an, ob der Layer transparent gezeichnet werden kann. Diese QGIS Version verwendet ein hardcodiertes `Ja`, sofern die Bildkodierung Transparenz bietet
  - **Kann reingezoomt werden** —Gibt an, ob dieser Layer gezoomt werden kann. Diese Version von QGIS verwendet standardmäßig `Ja`. Daher kann es sein, dass einige Layer komisch aussehen, die diese Funktion nicht unterstützen.
  - **Kaskadierend** — WMS-Server können als Proxy zwischen anderen WMS-Servern agieren, um Rasterdaten für einen Layer anzufordern. Dieser Eintrag gibt an, wieviele WMS-Server angefragt werden müssen, um die Daten zu bekommen.
  - **Fixierte Höhe, Fixierte Breite** —Gibt an, ob der Layer eine feste Pixeldimension hat. Diese Version von QGIS nimmt an, dass alle WMS-Layer diesen Wert nicht gesetzt haben. Daher kann es sein, dass einige Layer komisch aussehen, die diese Funktion nicht unterstützen.
  - **WGS 84 Boundingbox** —Gibt die Boundingbox eines Layers in WGS84-Koordinaten an. Einige WMS-Server setzen diese Werte nicht korrekt (z.B. stehen darin manchmal UTM-Koordinaten), sodass bei solchen Layern in QGIS der Eindruck entsteht, sehr weit herausgezoomt zu sein. Der Webmaster des WMS-Servers sollte dann auf dieses Problem aufmerksam gemacht werden. Das WMS XML-Element ist `LatLonBoundingBox`, `EX_GeographicBoundingBox` oder die CRS:84 `BoundingBox`.
  - **Verfügbare Koordinatensysteme** — Die Projektionen, in denen dieser Layer dargestellt werden kann. Diese sind dem Capabilities-Dokument des Servers entnommen.
  - **Verfügbare Stile** — Die Bildstile, in denen dieser Layer dargestellt werden kann.

Zeitliche-Eigenschaften

Raster *Zeitliche Eigenschaften* (nämlich *Dynamische Zeitsteuerung*) können für WMS und WMTS Layer gesetzt werden. Wenn ein WMS- oder WMTS-Layer mit Zeitdimension zum Projekt hinzugefügt wird, wird er standardmäßig im Bedienfeld *Layer* mit dem Symbol Zeitlicher Layer angezeigt. Ihre *Zeitlich*-Eigenschaften sind standardmäßig auf den temporalen Modus *Automatisch* eingestellt, was bedeutet, dass der Layer standardmäßig dem aktuellen Zeitbereich des temporalen Controllers folgt.

Sie können dann einen bestimmten statischen Zeitwert für den Layer anzeigen lassen, indem Sie das Kontrollkästchen *Dynamische Zeitsteuerung* deaktivieren und eine Option unter *Statischer WMS-T Zeitbereich* wählen:

- *Servervorgabe*
- *Vordefiniertes Datum* mit einem Server, der Daten für nicht zusammenhängende Zeiträume bereitstellt, oder *Vordefinierter Bereich* mit einem Server, der einen Bereich von verfügbaren Daten bereitstellt. Im letzteren Fall sind ein *Anfangsdatum* und ein *Enddatum* erforderlich. Ihre erwartete Formatierung kann aus der Option „Referenzzeit“ abgeleitet werden (siehe unten). je nachdem, ob der Anbieter über Daten für einen zusammenhängenden Zeitraum verfügt oder nicht.
- *Folgt dem zeitlichen Bereich des Projekts* wie im Eigenschaften-Dialog des Projekts definiert

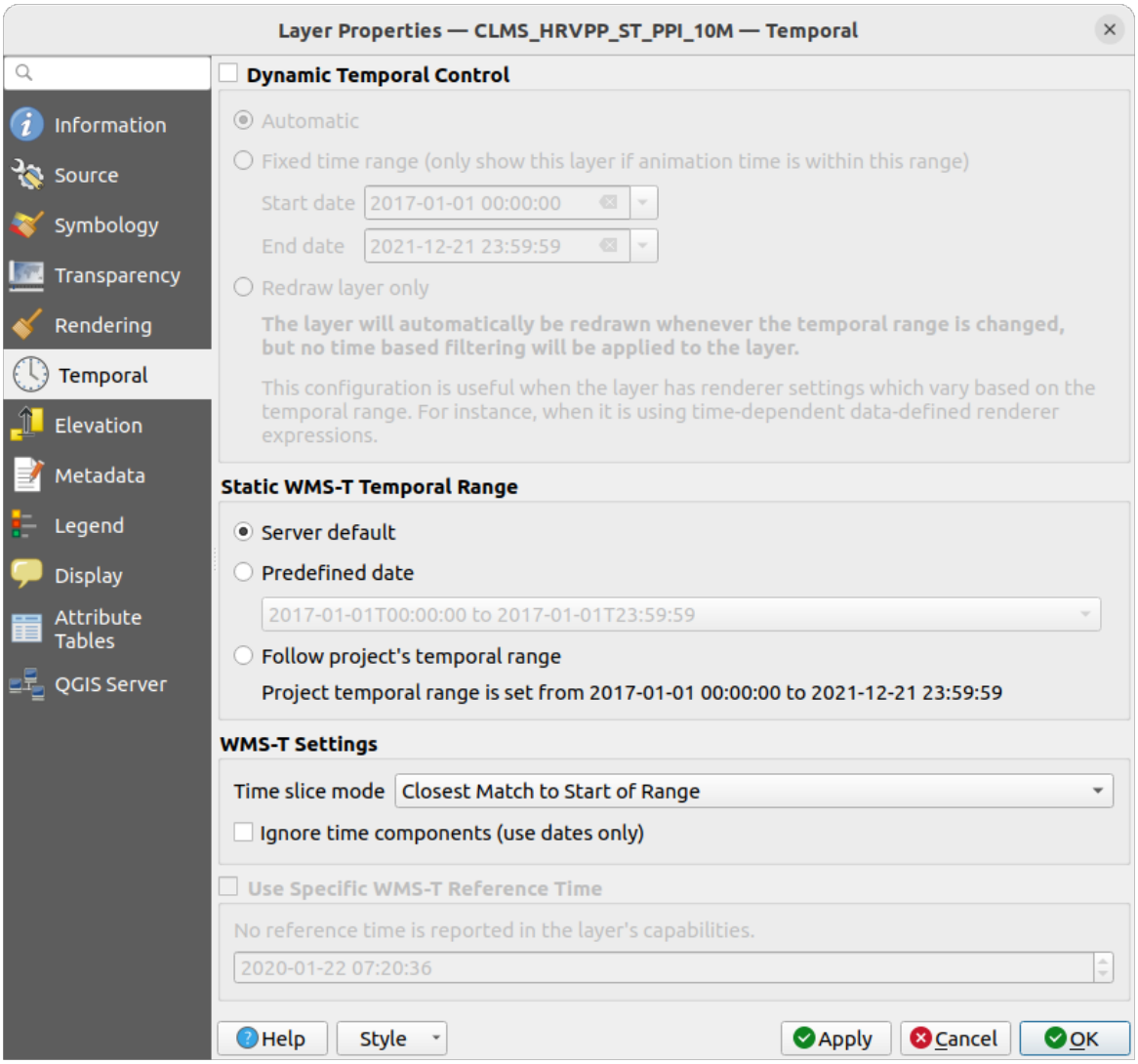


Abb. 19.4: Zeitliche Eigenschaften eines WMTS Layers

Unabhängig davon, welche Zeitdatensteuerung verwendet wird, gibt es einige *WMS-T Einstellungen*, die helfen, die

richtigen Zeitdaten anzuzeigen:

- *Zeitscheibenmodus*, der sein kann:
  - *Ganzen zeitlichen Bereich verwenden*
  - *Anpassung an den Beginn des Bereichs*
  - *Anpassung an das Ende des Bereichs*
  - *Nächste Übereinstimmung mit dem Beginn des Bereichs*
  - *Nächste Übereinstimmung mit dem Ende des Bereichs*
- *Zeitkomponenten ignorieren (nur Daten verwenden)*: Wenn diese Option aktiviert ist, wird die Zeitkomponente von zeitlichen Abfragen ignoriert und nur die Datumskomponente in Serveranfragen verwendet.

Sie können auch *Spezifische WMS-T-Referenzzeit verwenden* aus den Zeiten auswählen, die in den Fähigkeiten des Layers angegeben sind. Praktisch für Server, die einen nicht zusammenhängenden Satz von Datumszeitinstanzen (anstelle eines Datumsbereichs) bereitstellen.

QGIS Server-Eigenschaften

Zusätzlich zum Raster-Layer *Eigenschaften* werden bei der Veröffentlichung eines WMS/WMTS Layers mit QGIS Server folgende Optionen angezeigt:

- *WMS Drucklayer*: Ermöglicht die Festlegung eines alternativen WMS Layers, der für den Druck verwendet werden soll (in der Antwort GetProjectSettings). Praktisch für WMTS Layer, die im Allgemeinen nicht für den Druck geeignet sind.
- *WMS/WMTS-Datenquelle-URI veröffentlichen*: Ermöglicht es einem Web-Client, die WMS/WMTS-Daten direkt abzurufen
- *Als Hintergrundlayer anbieten*

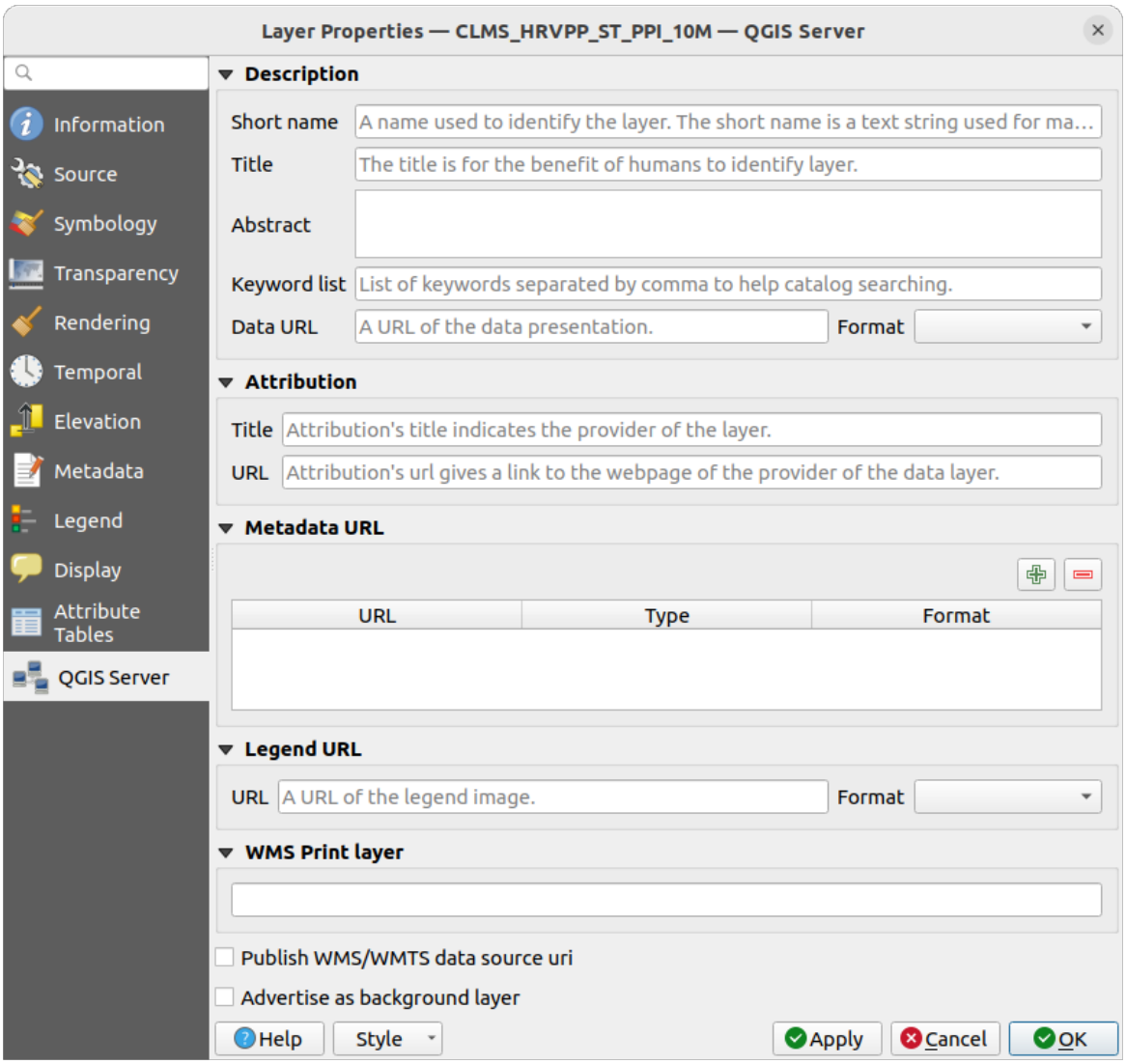


Abb. 19.5: QGIS Server Eigenschaften eines WMS/WMTS Layers

19.1.8 WMS-Legendengrafik im Inhaltsverzeichnis und Layout anzeigen

Der QGIS WMS Datenprovider ist in der Lage, eine Legendengrafik in der Layer-Liste des Inhaltsverzeichnisses und im Drucklayout anzuzeigen. Die WMS-Legende wird nur angezeigt, wenn der WMS-Server über die GetLegendGraphic-Fähigkeit verfügt und für den Layer die getCapability-Url angegeben ist, so dass Sie zusätzlich ein Styling für den Layer auswählen müssen.

Wenn eine legendGraphic verfügbar ist, wird diese unter dem Layer angezeigt. Sie ist klein und Sie müssen darauf klicken um Sie in Ihrer tatsächlichen Größe (gemäß den QgsLegendInterface Architekturlimitationen) zu öffnen. Das Klicken auf die Legende des Layers öffnet einen Rahmen mit einer Legende in voller Auflösung.

Im Drucklayout wird die Legende in ihrer ursprünglichen (heruntergeladenen) Größe integriert. Die Auflösung der Legendengrafik kann in den Artikeleigenschaften unter *Legende* ► *WMS LegendGraphic* an Ihre Druckanforderungen angepasst werden.

Die Legende wird Kontextinformationen basiert auf dem aktuellen Maßstab darstellen. Die WMS Legende wird nur gezeigt wenn der WMS Server eine GetLegendGraphic Capability hat und für dem Layer eine getCapability URL angegeben wurde, also müssen Sie einen Stil auswählen.

19.2 WCS Client

Ein Web Coverage Service (WCS) stellt eine Anbindung zu Rasterdaten in Formaten die nützlich für die client-seitige Darstellung, als Input für wissenschaftliche Modelle und für andere Clients zur Verfügung. Der WCS ist vergleichbar zu WFS und WMS. Als WMS und WFS Service Instanz erlaubt der WCS den Clients Teile von Serverinformationsbeständen basierend auf räumlichen Einschränkungen und Abfragekriterien auszuwählen.

QGIS hat einen nativen WCS Provider und unterstützt sowohl Version 1.0 und 1.1 (welche deutliche Unterschiede aufweisen), aktuell jedoch wird 1.0 vorgezogen da es mit 1.1 viele Probleme gibt (z.B. implementiert jeder Server es auf eine andere Art und Weise mit zahlreichen Besonderheiten).

Der native WCS Provider kümmert sich um alle Netzwerkanfragen und verwendet alle voreingestellten QGIS Netzwerkanfragen (insbesondere proxy). Es ist auch möglich einen Cache auszuwählen (‚Immer cachen', ‚Cache vorziehen', ‚Netzwerk vorziehen', ‚Immer Netzwerk') und der Provider unterstützt ebenfalls das Auswählen der Zeitposition wenn Temporal Domain vom Server angeboten wird.

Loading a WCS Layer

To be able to load a WCS Layer, first create a connection to the WCS server:

1. Öffnen Sie das Dialogfeld *Datenquellenverwaltung* durch Drücken der Schaltfläche Datenquellenverwaltung öffnen.
2. Enable the *WCS* tab
3. Click on *New…* to open the *Create a New WCS Connection* dialog

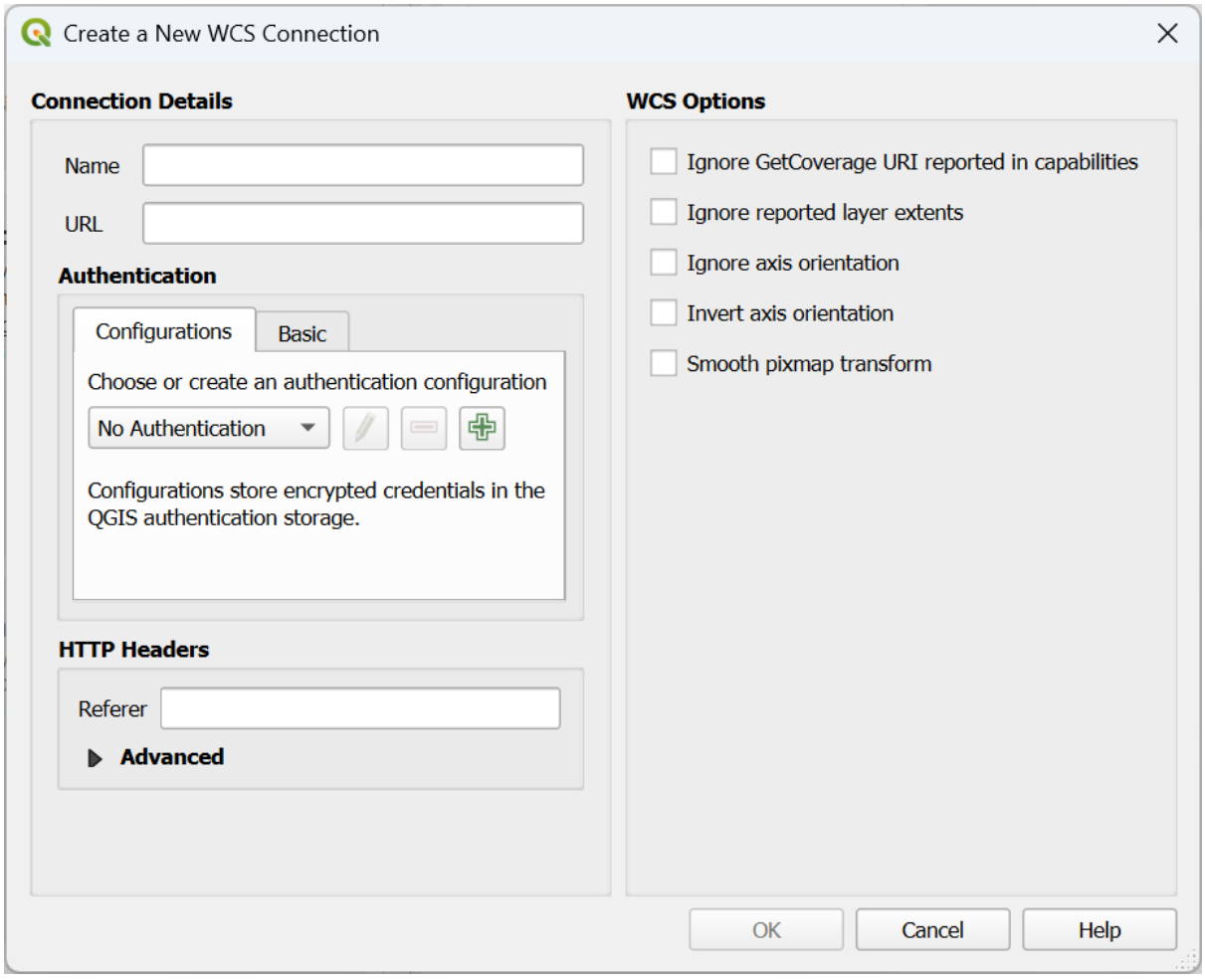


Abb. 19.6: Creating a connection to a WCS server

- *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*: DIe URL des Servers. Es muss sich um einen auflösbaren Hostnamen handeln – im selben Format, wie Sie es für eine Telnetverbindung benutzen würden oder um einen Host anzupingen, d.h. nur die Basis-URL. Ihre URL sollte also keine Teile wie `request=GetCapabilities` oder `version=1.0.0` enthalten.
- *Authentifizierung* (optional): Benutzen Sie eine *gespeicherte Konfiguration* oder eine Basic-Authentifizierung mit *Benutzername* und *Passwort*.

  Warnung

  Durch die Eingabe von **Benutzername** und **Passwort** auf der Registerkarte *Authentifizierung* werden ungeschützte Anmeldedaten in der Verbindungskonfiguration gespeichert. Diese **Anmeldeinformationen sind sichtbar**, wenn Sie z.B. die Projektdatei mit jemandem teilen. Daher ist es ratsam, Ihre Anmeldeinformationen stattdessen in einer *Authentifizierungskonfiguration* zu speichern (Registerkarte *Konfigurationen*). Siehe *Authentifizierungssystem* für weitere Details.

- HTTP *Referer*
- ☐ *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*
- ☐ *Achsenorientierung invertieren*
- ☐ *Pixmap-Transformation glätten*

4. Drücken Sie *OK*, um die Verbindung herzustellen.

Beachten Sie, dass alle Proxy-Einstellungen, die Sie in Ihren Einstellungen festgelegt haben, ebenfalls erkannt werden. Beachten Sie auch, dass es möglich ist, die Verbindungsparameter aus einer `.XML`-Datei zu *Laden* oder in einer solchen zu *Speichern*.

Now we are ready to load WCS layers from the above connection.

19.3 WFS und WFS-T Klient

In QGIS verhält sich ein WFS Layer ziemlich genau wie jeder andere Vektorlayer. Sie können Objekte identifizieren und auswählen und die Attributtabelle anzeigen. QGIS unterstützt WFS 1.0.0, 1.1.0, 2.0 und OGC API - Objekte (OAPIF), einschließlich der Bearbeitung (durch WFS-T). QGIS unterstützt auch das Herunterladen im Hintergrund und das progressive Rendering, das Zwischenspeichern heruntergeladener Objekte auf der Festplatte und die automatische Versionserkennung.

Layer von Servern, die OGC API - Features - Teil 4: Erstellen, Ersetzen, Aktualisieren und Löschen implementieren, können in den Bearbeitungsmodus versetzt werden, um das Erstellen, Ändern und Löschen von Objekten zu ermöglichen. Beachten Sie, dass jedes erstellte/geänderte/gelöschte Objekt eine eigene Netzwerkanfrage erfordert, so dass die Leistung im Falle der gleichzeitigen Änderung von Hunderten oder mehr Objekten leiden kann.

Im Allgemeinen ähnelt das Hinzufügen eines WFS Layers dem Verfahren, das für WMS verwendet wird. Es sind keine Standardserver definiert, sodass Sie Ihre eigenen hinzufügen müssen. Sie können WFS-Server mithilfe des *MetaSearch-Plugins* oder Ihrer bevorzugten Web-Suchmaschine finden. Es gibt eine Reihe von Listen mit öffentlichen URLs, von denen einige gepflegt werden und andere nicht.

Einen WFS-Layer laden

Als Beispiel verwenden wir den Gateway Geomatics WFS-Server und zeigen einen Layer an.

```
https://demo.gatewaygeomatics.com/cgi-bin/wfs_gateway?REQUEST=GetCapabilities&
↪VERSION=1.0.0&SERVICE=WFS
```

Um einen WFS Layer laden zu können, muss zunächst eine Verbindung zum WFS Server hergestellt werden:

1. Öffnen Sie das Dialogfeld *Datenquellenverwaltung* durch Drücken der Schaltfläche Datenquellenverwaltung öffnen.
2. Aktivieren Sie die Registerkarte *WFS / OGC API - Objekte*
3. Klicken Sie auf *Neu…*, um den Dialog *Neue WFS-Verbindung anlegen* zu öffnen
4. In the *Connection Details* group, enter `Gateway Geomatics` as name
5. Geben Sie die URL ein (siehe oben)
6. Optionally, enter the *Authentication* settings

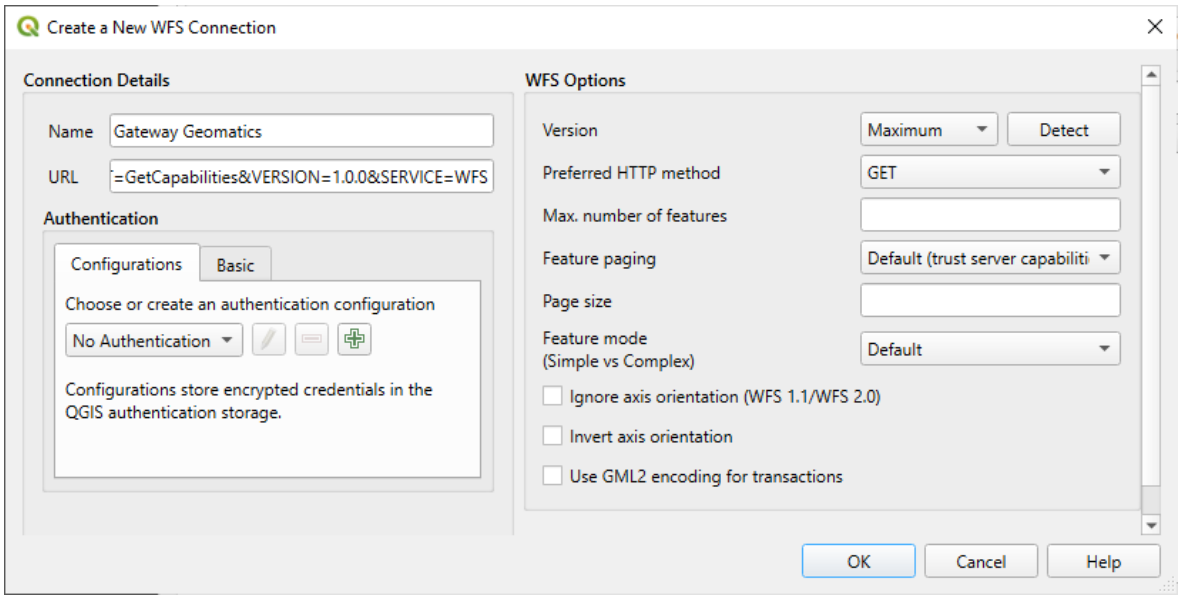


Abb. 19.7: Herstellen einer Verbindung zu einem WFS-Server

> **Bemerkung**
>
> 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:
   - Geben Sie die WFS-Version des Servers an. Ist sie nicht bekannt, drücken Sie die Schaltfläche *Bestimmen*, um sie automatisch abzurufen.
   - 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.
  - *Achsenorientierung ignorieren (WFS 1.1/WFS 2.0)* erzwingen
  - *Achsenorientierung invertieren*.
  - *GML2-Kodierung für Transaktionen verwenden*.

> **Warnung**
>
> Durch die Eingabe von **Benutzername** und **Passwort** auf der Registerkarte *Authentifizierung* werden ungeschützte Anmeldedaten in der Verbindungskonfiguration gespeichert. Diese **Anmeldeinformationen sind sichtbar**, wenn Sie z.B. die Projektdatei mit jemandem teilen. Daher ist es ratsam, Ihre Anmeldeinformationen stattdessen in einer *Authentifizierungskonfiguration* zu speichern (Registerkarte *Konfigurationen*). Siehe *Authentifizierungssystem* für weitere Details.

8. Drücken Sie *OK*, um die Verbindung herzustellen.

Beachten Sie, dass alle Proxy-Einstellungen, die Sie in Ihren Einstellungen festgelegt haben, ebenfalls erkannt werden. Beachten Sie auch, dass es möglich ist, die Verbindungsparameter aus einer `.XML`-Datei zu *Laden* oder in einer solchen zu *Speichern*.

Jetzt können wir die WFS Layer aus der obigen Verbindung laden.

1. Wählen Sie „Gateway Geomatics“ aus der Dropdown-Liste *Serververbindungen*.
2. Klicken Sie auf *Verbinden*
3. Wählen Sie den Layer *Parks* in der Liste
4. Sie können auch wählen, ob Sie:
   - *Nur Objekte anfordern, die sich mit dem Ansichtsbereich überschneiden*
   - *Ändern…* des KBS des Layers in ein anderes, vom Dienst unterstütztes KBS
   - oder eine Abfrage erstellen, um bestimmte Objekte aus dem Dienst abzurufen: Doppelklicken Sie auf die Zeile des Layers, um den Dialog *SQL-Abfrageerstellung* zu öffnen. Dieser Dialog bietet Widgets zum Schreiben einer erweiterten SQL-Abfrage, die auf den verfügbaren Tabellen und Spalten des Dienstes basiert, mit Sortierung und Filterung und einer Reihe von SQL-Funktionen, räumlichen Prädikaten und Operatoren.

     Die von Ihnen erstellte Abfrage wird nach der Validierung in der Spalte *SQL* in der Tabelle *WFS / OGC API - Features* angezeigt, und der gefilterte Layer wird im Bedienfeld *Layer* mit dem Symbol versehen. Es ist also jederzeit möglich, die Abfrage anzupassen.
5. Click *Add* to add the layer to the map, using its default name from the queried service.

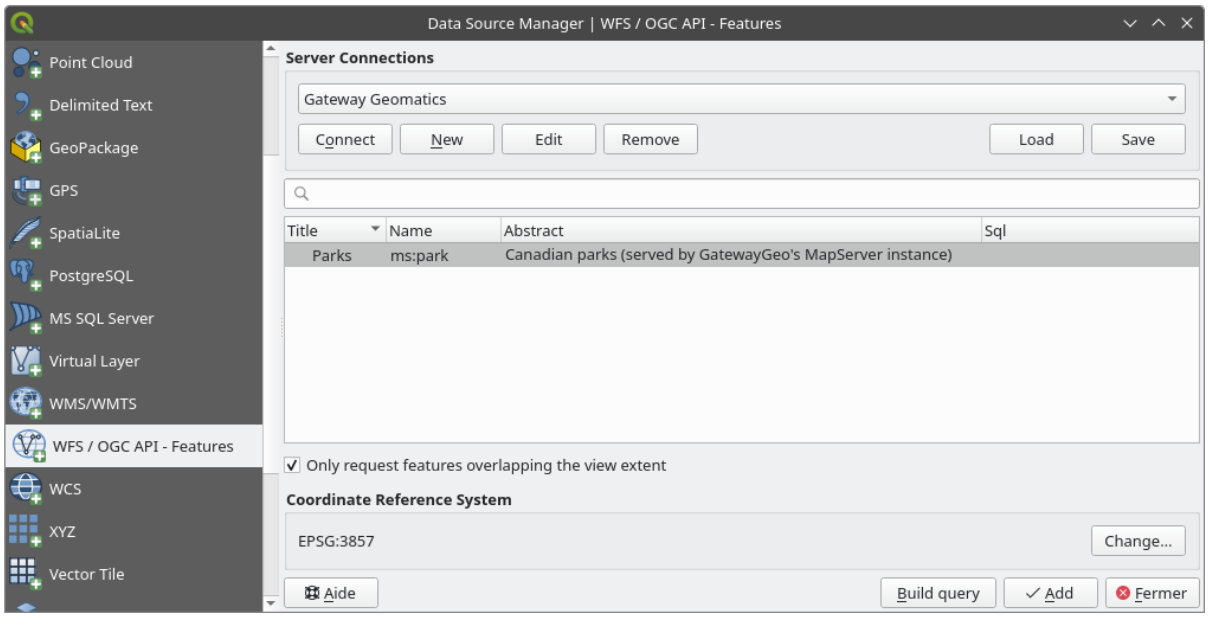


Abb. 19.8: Einen WFS Layer hinzufügen

Sie werden feststellen, dass der Fortschritt des Downloads unten links im QGIS-Hauptfenster angezeigt wird. Sobald der Layer geladen ist, können Sie einige Objekte identifizieren und auswählen und die Attributtabelle anzeigen.

19.4 SensorThings

QGIS unterstützt Verbindungen zu OGC SensorThings API, einem Standard, der ein offenes und einheitliches Framework für die Vernetzung von IoT-Sensoren, Daten und Anwendungen über das Internet bereitstellt. Es handelt sich um einen offenen Standard, der sich mit der syntaktischen und semantischen Interoperabilität des Internets der Dinge befasst. Er basiert auf dem Datenmodell Observations and Measurements, einem standardisierten Modell für Beobachtungen und für Funktionen, die bei der Durchführung von Beobachtungen mit Probenahmen verbunden sind.

19.4.1 Verbindung herstellen

Um SensorThings-Daten zu QGIS hinzuzufügen, verwenden Sie die Registerkarte *SensorThings* im Dialogfeld *Datenquellenverwaltung*.

Um eine neue Verbindung herzustellen, klicken Sie auf *Neu* (oder *Neue SensorThings-Verbindung* im Browser-Bedienfeld) und geben Sie *Name* und *URL* ein. Erweiterte Optionen wie *Authentifizierung* und ein *Referer* können ebenfalls konfiguriert werden.

Drücken Sie *OK*, um die Verbindung herzustellen. Anschließend können Sie:

- *Bearbeiten* der Verbindungseinstellungen für SensorThings.
- *Entfernen* der SensorThings-Verbindung.

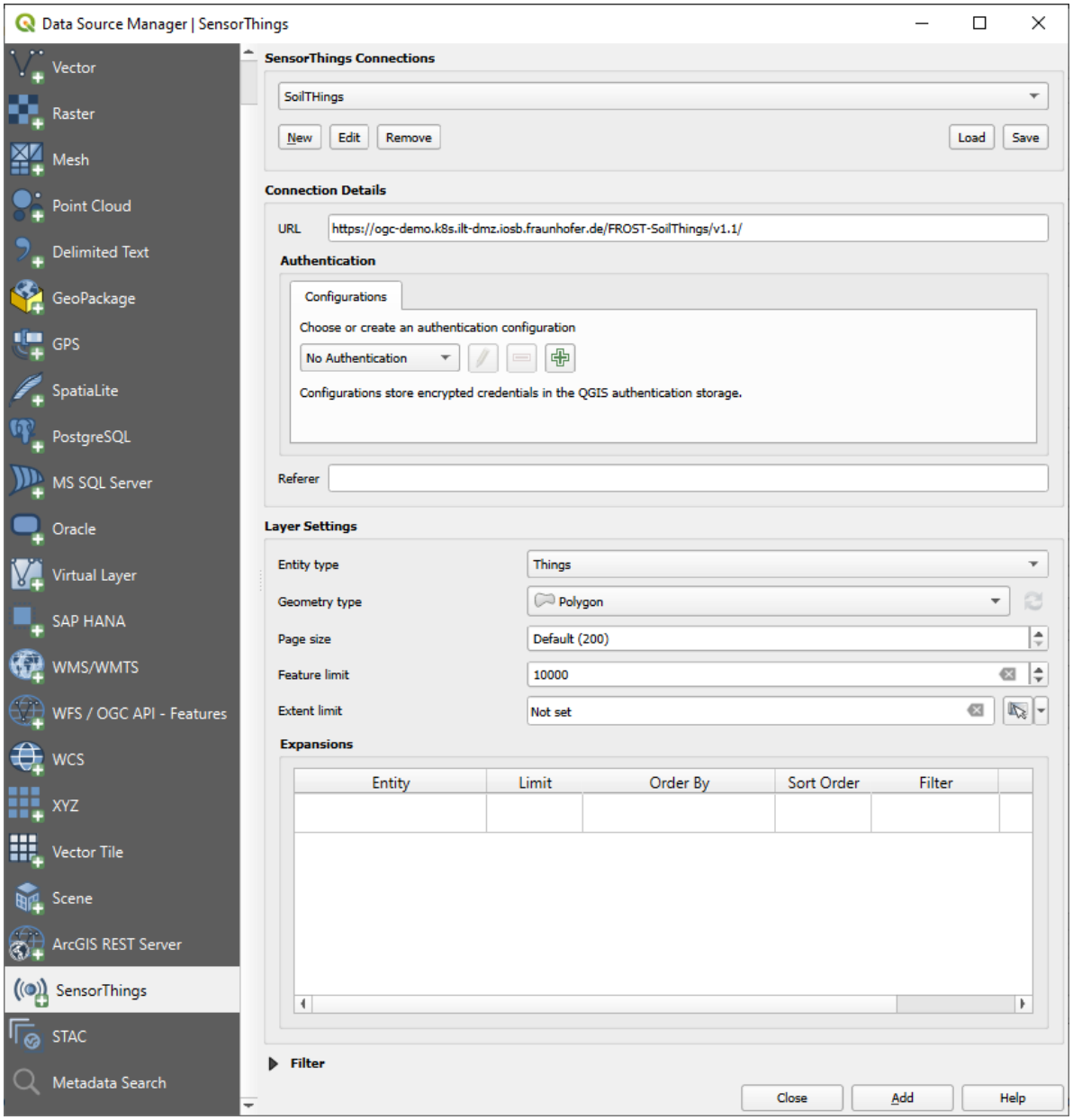


Abb. 19.9: SensorThings-Verbindungsdialog

Konfigurationen können über den Eintrag *SensorThings* im Dialogfeld *Datenquellenverwaltung* oder über das Kontextmenü im *Browser*-Bedienfeld (*Verbindungen speichern*) in einer `.XML`-Datei gespeichert werden (*Speichern*). Ebenso können Konfigurationen aus einer Datei hinzugefügt werden (*Laden*).

19.4.2 Laden von SensorThings-Daten

Die Beziehungen zwischen den in einem SensorThings-Datensatz gespeicherten Layern (sogenannten Entitäten) sind in der folgenden Abbildung dargestellt.

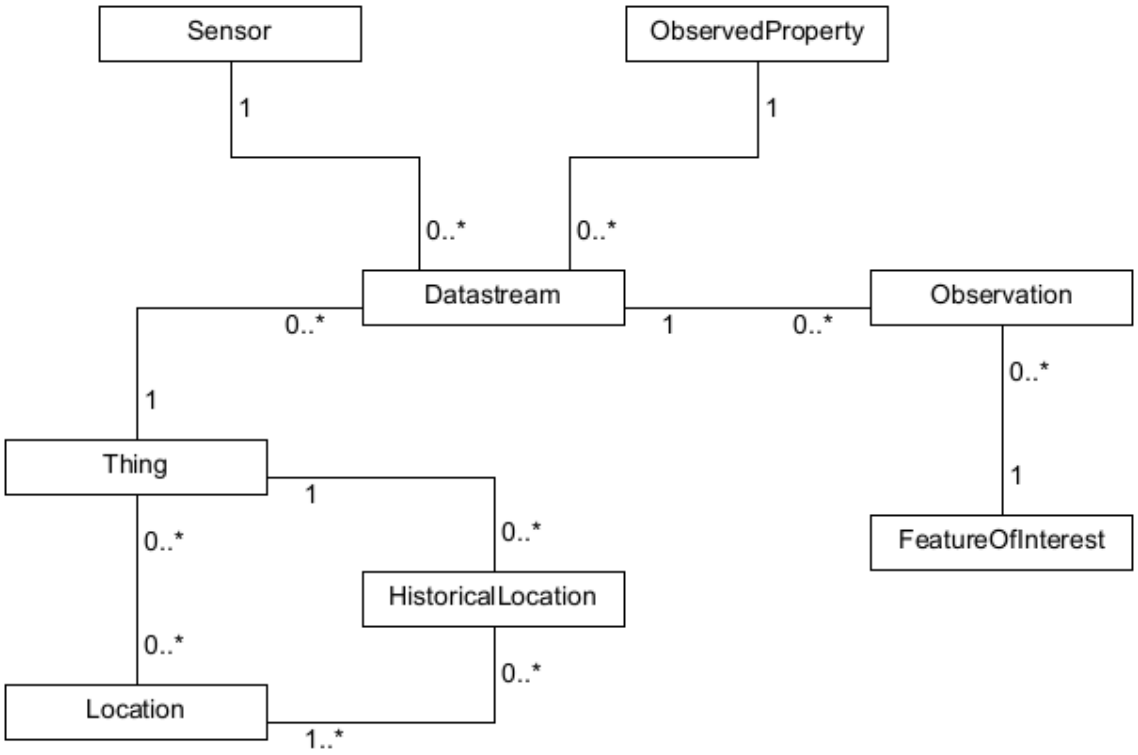


Abb. 19.10: Datenmodell Beobachtungen und Messungen (Quelle: „Sensor Web Tutorial“ von SIST network <https://sist.pages.in2p3.fr/anf21-sos52north/hands-on/09_sta-example-request/>_)

Jede Art von Entität kann in QGIS geladen werden, aber nicht alle sind räumliche Daten. Um eine Entität zu laden, gibt es die *Layer-Einstellungen*, die konfiguriert werden können:

- *Entitätstyp*: Die Entität, die aus dem Datenmodell als Layer in QGIS geladen werden soll.
- *Geometrie-Typ*: Der Geometrie-Typ des ausgewählten Elements, das geladen werden soll. Drücken Sie Verfügbare Typen prüfen, um die Liste auf die tatsächlich unterstützten Geometrie-Typen zu beschränken.
- *Seitengröße*
- *Feature Limit* legt eine maximale Anzahl von Features fest, die vom Dienst angefordert werden können.
- *Extent Limit* legt eine maximale Ausdehnungsgrenze für den Layer fest, sodass nur Features innerhalb dieser Ausdehnung angefordert werden.
- *Erweiterungen*: Das Datenmodell von SensorThings bietet einen Mechanismus zur Erweiterung der Ergebnisse auf verwandte Entitäten, ähnlich wie Tabellen in einer relationalen Datenbank miteinander verknüpft werden. Mit diesem Ansatz können Sie den ausgewählten Layer um Daten aus anderen Elementen erweitern. Dadurch wird die Beziehung vereinfacht, es werden ebenso viele übergeordnete Objekte wie untergeordnete Objekte erstellt und zusätzliche Eigenschaften werden als Spalten in die Attributtabelle aufgenommen.

Verwenden Sie *Filter*, um eine Abfrage zum Filtern der Daten unter Verwendung der SensorThings-Filtersyntax zu erstellen.

Bemerkung

Die oben genannten Einstellungen und Filteroptionen können auch im Dialogfeld Layereigenschaften unter der Registerkarte *Quelle* aktualisiert werden, sobald sie in QGIS geladen sind.

Drücken Sie *Hinzufügen*, um den ausgewählten Entitätstyp als Layer in QGIS zu laden.

19.4.3 Arbeiten mit einem Vektor-Layer aus SensorThings

Ein SensorThings-Layer wird in QGIS als Vektor-Layer geladen. Als solcher zeigt er dieselben Registerkarten in den *Layer-Eigenschaften* an und ermöglicht dieselben Objektinteraktionen mithilfe der Auswahl- oder Identifizierungswerkzeuge. Es gibt jedoch einige Besonderheiten, die Sie bei der Arbeit mit SensorThings-Daten beachten sollten.

Aufgrund des Datenmodells ist die Ergebnis-Eigenschaft einer SensorThings-Beobachtung ein Zeichenfolgenfeld. Wenn Sie die numerische Darstellung beispielsweise in einem abgestuften Stil verwenden möchten, verwenden Sie einen Ausdruck, um den Wert in den Datentyp Realzahl zu konvertieren, und versuchen Sie es mit try(), falls dies fehlschlägt (z. B. `try( to_real(„Observation_result“), Null)`).

Wenn Sie ein Diagramm der Beobachtungen an einem oder mehreren Stellen erstellen möchten, können Sie das QGIS-Plugin *Data Plotly* installieren.

1. Wählen Sie nun die Beobachtungen an einem Punkt in der Kartenansicht aus.
2. Öffnen Sie das Plotly-Fenster und aktivieren Sie das Kontrollkästchen *Nur ausgewählte Objekte verwenden*.
3. Wählen Sie in der x-Spalte eine Datums-/Zeit-Eigenschaft und in der y-Spalte die *Observation_result* aus. Dadurch werden die Beobachtungen an diesem Ort im Zeitverlauf dargestellt.
4. Stellen Sie sicher, dass Sie nach einer einzelnen beobachteten Eigenschaft filtern.
5. Beachten Sie, dass sich die Karte ändert, sobald Sie andere Orte auf der Karte auswählen.

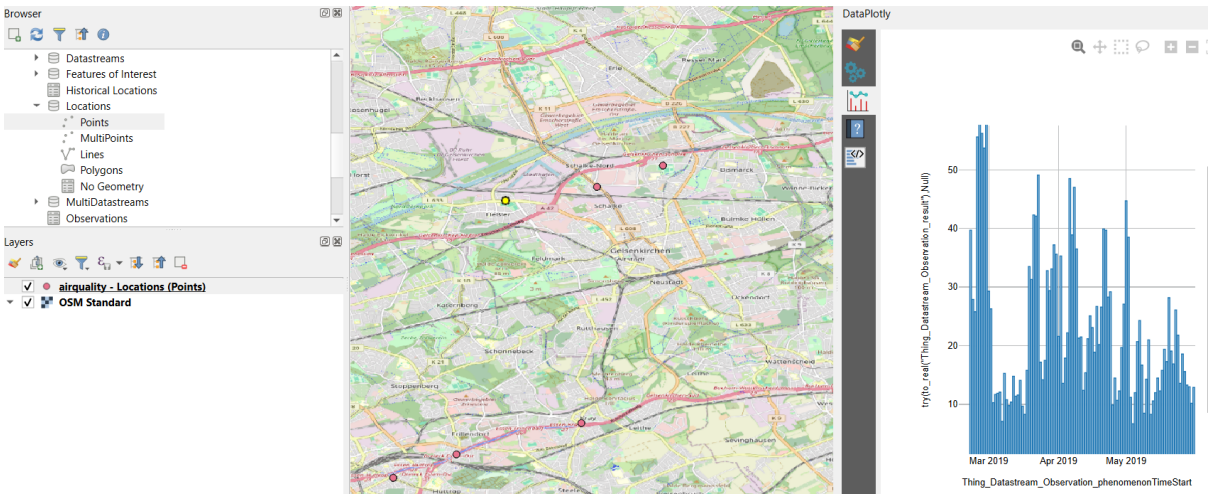


Abb. 19.11: Verwenden Sie Data plotly, um die Luftqualitätsbeobachtungen an einem Standort darzustellen.

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 Verbindung herstellen

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

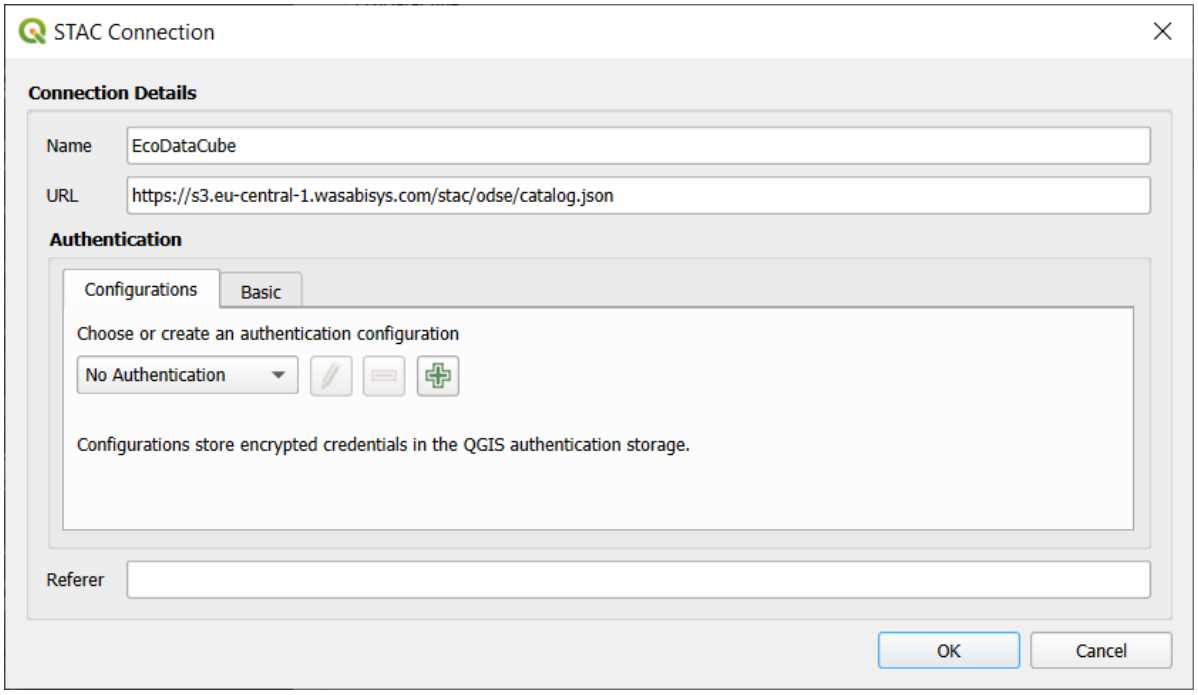


Abb. 19.12: Creating a connection to a STAC server

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.

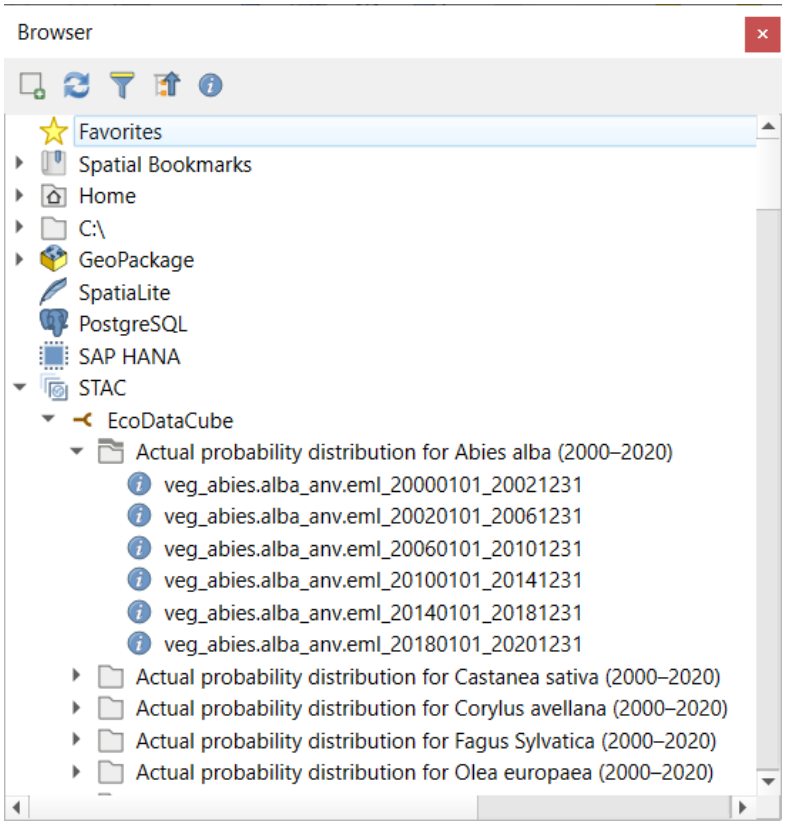


Abb. 19.13: STAC connection expanded in the Browser, showing Collections and Items

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…*

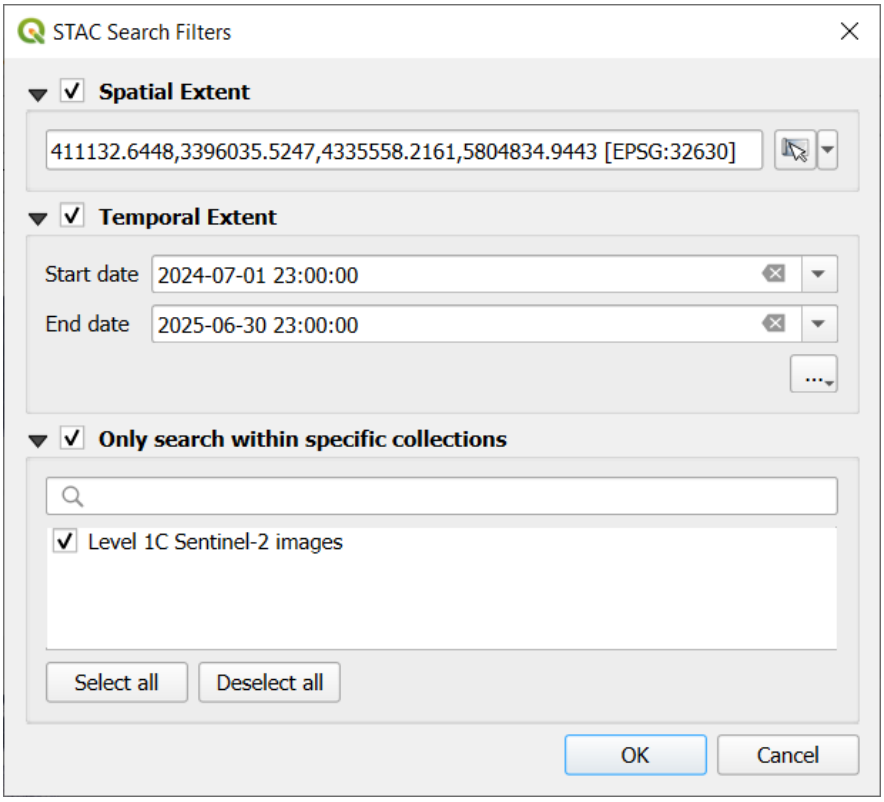


Abb. 19.14: Define spatial and temporal filters for a STAC catalog search

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.

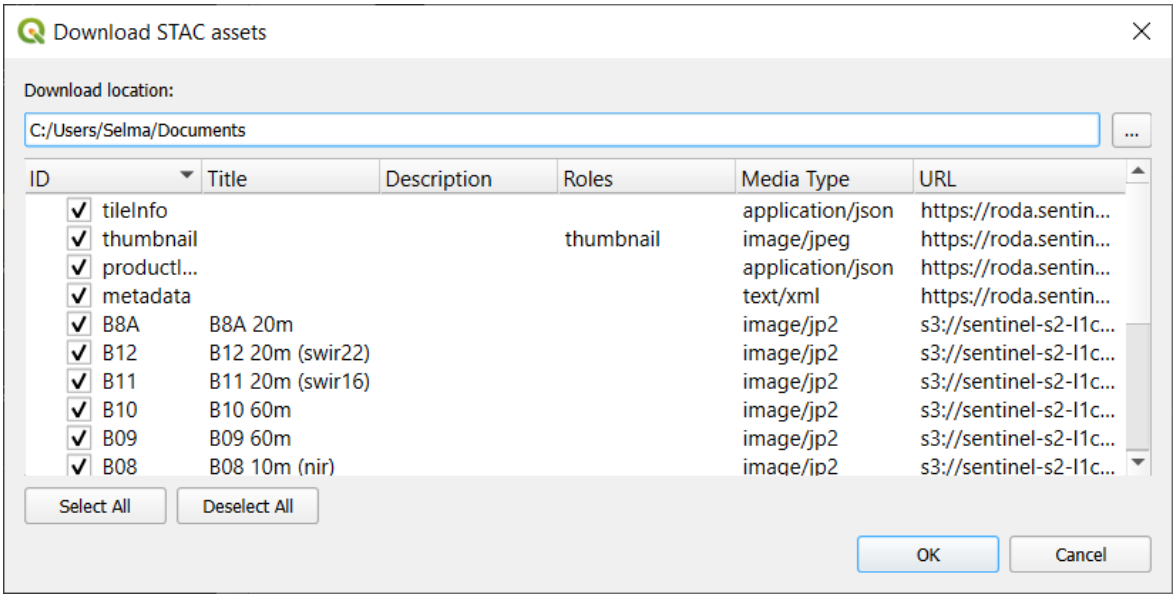


Abb. 19.15: Download STAC item assets

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 *Verwendung von ArcGIS REST-Servern* 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.

KAPITEL 20

Arbeiten mit GPS Daten

20.1 Einführung in GNSS/GPS-Daten

20.1.1 Was ist GPS?

GPS, das **Global Positioning System**, ist ein satellitengestütztes System, mit dem jeder, der einen GPS-Empfänger besitzt, seine genaue Position überall auf der Welt bestimmen kann. GPS wird als Navigationshilfe verwendet, zum Beispiel in Flugzeugen, auf Booten und von Wanderern. Der GPS-Empfänger nutzt die Signale der Satelliten, um seinen Breitengrad, Längengrad und (manchmal) die Höhe zu berechnen. Die meisten Empfänger haben auch die Fähigkeit, Daten zu speichern:

- Orte, auch Wegpunkte genannt (engl. **waypoints**)
- Abfolgen von Wegpunkte, die eine Strecke bilden (**route**).
- und ein **Track**-Protokoll der Bewegungen des Empfängers über die Zeit.

Wegpunkte, Strecken und Tracks sind die drei grundlegenden Objekttypen in GPS-Daten. QGIS zeigt Wegpunkte in Punkt-Layern an, während Routen und Tracks in Linienstring-Layern dargestellt werden.

> **Bemerkung**
>
> QGIS unterstützt auch GNSS-Empfänger (GNNS ist ein Oberbegriff für satellitengestützte Navigation). In dieser Dokumentation verwenden wir jedoch weiterhin den Begriff GPS.

20.1.2 Übertragen oder Laden von GPS-Daten

Laden einer GPX-Datei

Es gibt viele verschiedene Datenformate zum Speichern von GPS-Daten. Das von QGIS unterstützte Format ist GPX (GPS eXchange format), ein Standardformat, welches Wegpunkte, Strecken und Tracks in einer Datei enthalten kann.

Um eine `GPX`-Datei zu laden:

1. Öffnen Sie die Registerkarte *GPS* im Dialog *Datenquellenverwaltung*, d.h.:
   - Klicken Sie auf die Schaltfläche Datenquellenverwaltung öffnen in der Symbolleiste (oder drücken Sie `Strg+L`) und aktivieren Sie die GPS-Registerkarte
   - Oder wählen Sie *Layer* ► *Layer hinzufügen* ► *Neuen GPX Layer hinzufügen…*
2. Verwenden Sie die Schaltfläche … Durchsuchen neben der Option *GPX-Datensatz* zur Auswahl der GPX-Datei
3. Verwenden Sie die Kontrollkästchen, um die *Objekttypen* auszuwählen, die Sie aus der Datei laden möchten. Jedes Objekt (*Wegpunkte*, *Routen* oder *Spuren* = Tracks) wird in einen eigenen Layer geladen.

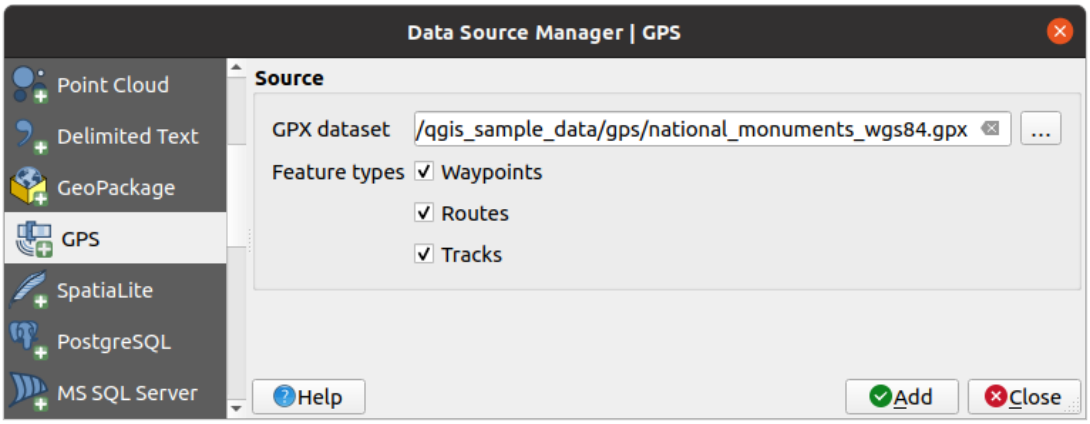


Abb. 20.1: Dialog GPS-Daten laden

Laden auf oder von einem Gerät

Es gibt viele verschiedene Arten von GPS-Geräten und Formaten. Da QGIS GPX-Dateien verwendet, brauchen Sie eine Möglichkeit, andere GPS-Dateiformate in GPX zu konvertieren. QGIS kann dies mit dem kostenlosen Programm GPSBabel tun. GPSBabel kann Ihnen helfen, Wegpunkte, Tracks und Strecken zwischen gängigen GPS-Empfängern wie Garmin oder Magellan und Kartenprogrammen wie Google Earth oder Basecamp zu konvertieren. Es werden buchstäblich Hunderte von GPS-Empfängern und Programmen unterstützt. Es kann auch GPS-Daten zwischen Ihrem Computer und einem GPS-Gerät übertragen.

Unter *Einstellungen* ► *Optionen* ► *GPS* ► *GPSBabel* erlaubt QGIS die Definition eines eigenen Gerätetyps und die Festlegung von Konvertierungsparametern, die später von der *Verarbeitung von GPS-Algorithmen* verwendet werden können.

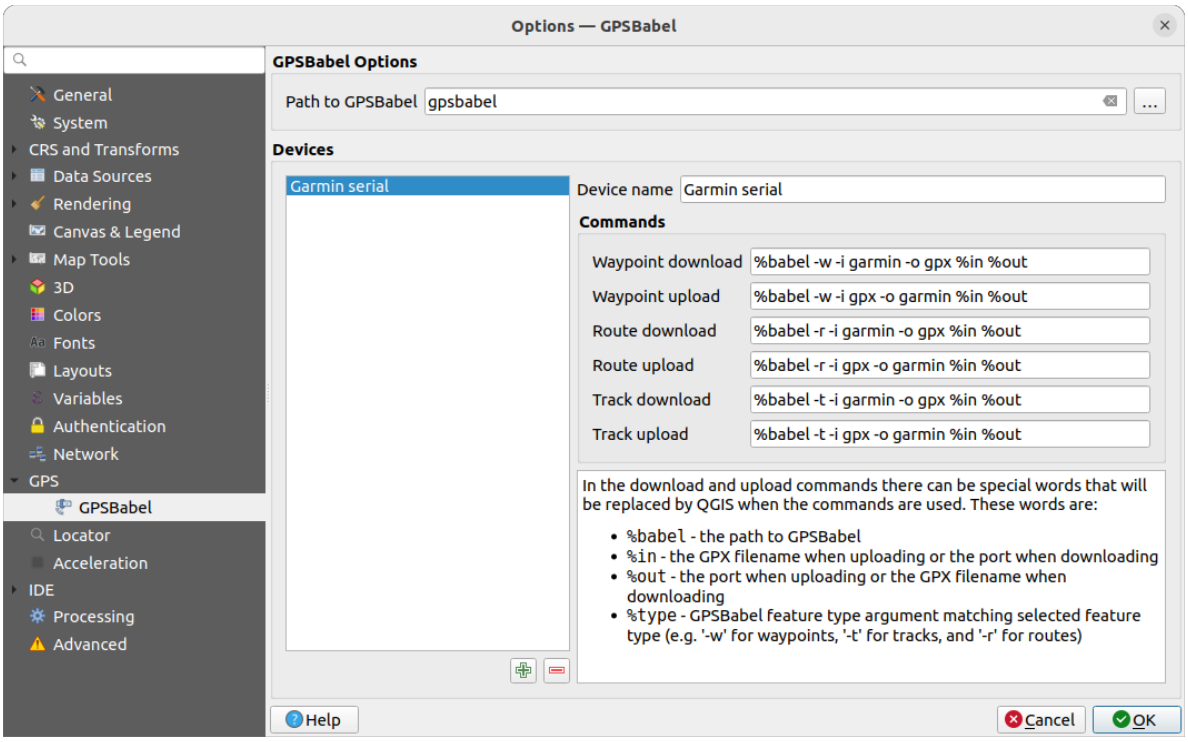


Abb. 20.2: GPSBabel Einstellungen

1. Zuerst müssen Sie den *Pfad zu den GPSBabel* definieren.
2. Dann möchten Sie vielleicht Ihr Gerät hinzufügen. Sie können die Geräteliste mit der Schaltfläche Neues Gerät hinzufügen oder Gerät entfernen aktualisieren.
3. Für jedes Gerät:
   - geben Sie einen *Gerätenamen* an
   - können Sie verschiedene *Befehle* konfigurieren, die QGIS bei der Interaktion mit dem System verwenden wird, wie z.B:
     - *Wegpunkt herunterladen* vom Gerät
     - *Wegpunkt hinaufladen* auf das Gerät
     - *Route herunterladen* vom Gerät
     - *Route hinaufladen* zum Gerät
     - *Spur herunterladen* vom Gerät
     - *Spur hinaufladen* auf das Gerät

   Obwohl es sich bei den Befehlen normalerweise um GPSBabel-Befehle handelt, können Sie auch jedes andere Kommandozeilenprogramm verwenden, das eine GPX-Datei erstellen kann. QGIS wird die Schlüsselwörter `%type`, `` 1%i n`` und `` 2%o ut`` ersetzen, wenn es den Befehl ausführt.

   Wenn Sie z.B. einen Gerätetyp mit dem Download-Befehl `gpsbabel %type -i garmin -o gpx 1%i n 2%o ut` erstellen und diesen dann zum Herunterladen von Wegpunkten von Port `/dev/ttyS0` in die Datei `output.gpx`` verwenden, wird QGIS die Schlüsselwörter ersetzen und den Befehl `gpsbabel -w -i garmin -o gpx /dev/ttyS0 output.gpx` ausführen.

   Lesen Sie das GPSBabel-Handbuch, um die Befehlszeilenoptionen zu erfahren, die für Ihren Anwendungsfall von Bedeutung sind.

Sobald Sie einen neuen Gerätetyp erstellt haben, wird dieser in den Gerätelisten für die GPS-Download- und Upload-Algorithmen angezeigt.

Bemerkung

Mit GPS-Geräten können Sie Daten in verschiedenen Koordinatensystemen speichern. Wenn Sie eine GPX-Datei herunterladen (von Ihrem GPS-Gerät oder einer Website) und sie dann in QGIS laden, stellen Sie sicher, dass die in der GPX-Datei gespeicherten Daten WGS 84 (Breitengrad/Längengrad) verwenden. QGIS erwartet dies, und es ist die offizielle GPX-Spezifikation. Siehe GPX 1.1 Schema Dokumentation.

20.2 Live-GPS-Verfolgung

QGIS kann Ihnen helfen, Feldkartierungen mit einem GPS-Empfänger durchzuführen. Eine solche Live-Tracking-Operation wird mit der *GPS Werkzeugleiste* durchgeführt. Einige *Gerätekonfigurationen* können erforderlich sein, bevor Sie QGIS und den GPS-Empfänger verbinden können.

20.2.1 GPS Werkzeugleiste

Die *GPS-Werkzeugleiste* bietet die wichtigsten Werkzeuge zur Steuerung einer Live-Tracking-Sitzung. Sie kann über *Ansicht* ► *Werkzeugleisten* ► *GPS* aktiviert werden. Sie verfolgt den Status des Projekts, des GPS und des aktuellen GPS-Tracks und aktiviert Aktionen nur dann, wenn sie sinnvoll sind. Sobald ein Gerät von QGIS erkannt wird, können Sie mit ihm interagieren:

- Mit GPS verbinden
- Verbindung zum GPS-Gerät beenden
- Karte auf GPS-Position zentrieren: die Karte wird sofort auf der aktuellen GPS-Position neu zentriert, unabhängig vom Parameter *automatische Neupositionierung*.
- Ziellayer für GPS-digitalisierte Objekte setzen: Standardmäßig wird QGIS *Dem aktiven Layer folgen* für die Digitalisierung von Objekten verwenden. Das bedeutet, dass sich die GPS-Digitalisierungswerkzeuge an den Layer anpassen, der im Bedienfeld *Layer* ausgewählt wurde, und die erstellten Objekte entsprechend in diesem Layer gespeichert werden. Dies kann zwar in manchen Kontexten nützlich sein, erfordert aber auch eine sorgfältige Interaktion mit anderen Layern, um zu vermeiden, dass Objekte versehentlich im falschen Layern gespeichert werden. Mit dieser Option können Sie während der Live-Tracking-Sitzung explizit einen Layer für die Datenspeicherung angeben, den Sie je nach Bedarf wechseln können. Exponierte Werkzeuge passen sich ebenfalls an den gewählten Layer-Typ an.

  Der Layer für das GPS-Ziel wird automatisch editierbar gemacht, wenn ein Objekt erstellt wird, und der Benutzer wird entsprechend informiert.
- Aktionen zur Erstellung von Objekten:
  - Stützpunkt der aktuellen GPS-Position zur GPS-Spur hinzufügen
  - Neues Punktobjekt an der aktuellen GPS-Position erzeugen
  - Neues Linienobjekt an der aktuellen GPS-Position erzeugen
  - Neues Polygonobjekt an der aktuellen GPS-Position erzeugen
  - Spur zurücksetzen
- GPS-Informationsfenster anzeigen: üffent das *GPS-Information* Bedienfeld
- ein Anzeigefeld für den schnellen Zugriff auf einige GPS-Informationskomponenten, die Sie möglicherweise benötigen:

  - ☐ Position anzeigen
  - ☐ Höhe anzeigen (geoid)
  - ☐ Höhe anzeigen (WGS84-Ellipsoid)
  - ☐ Geschwindigkeit über Grund anzeigen
  - ☐ Kurs anzeigen
  - ☐ Spurgesamtlänge anzeigen
  - ☐ Entfernung zum Spurbeginn anzeigen

- Die Schaltfläche 🔧 Einstellungen öffnet ein Dropdown-Menü mit allgemeinen Einstellungen, die während der Sitzung geändert werden können:
  - ☐ *Positionsmarkierung anzeigen*
  - ☐ *Kurslinie anzeigen*
  - ☐ *Karte in GPS-Richtung drehen*
  - Optionen zur Steuerung der Kartenneuausrichtung:
    - ○ *Karte immer neu zentrieren*, wenn die GPS-Position um einen bestimmten Abstand (im Verhältnis zur Kartenansicht) von der Mitte der Kartenansicht abweicht
    - ◉ *Karte beim Verlassen des Bereichs neu zentrieren*
    - ○ *Niemals neu zentrieren*
  - ☐ *Spurstützpunkte automatisch hinzufügen* wenn neue Standorte vom GPS-Gerät empfangen werden.
  - ☐ *Hinzugefügte Objekte automatisch speichern*: Objekte, die anhand von GPS-Positionen erstellt wurden, werden sofort auf die entsprechenden Layer übertragen (der übliche Layer-Bearbeitungspuffer wird übersprungen).
  - *Zeitstempelziel* legt das Feld fest, in dem ein Zeitstempel der Erfassung gespeichert wird
  - ☐ *In GeoPackage/Spatialite protokollieren…*: Nach der Aktivierung wird der Benutzer aufgefordert, eine vorhandene GeoPackage/SpatiaLite-Datei auszuwählen oder einen neuen Dateinamen einzugeben. In der Datei wird eine Tabelle `gps_points` und `gps_tracks` mit einer vordefinierten Struktur erstellt.

    Alle eingehenden GPS-Nachrichten werden im Layer `gps_points` aufgezeichnet, zusammen mit Informationen über Geschwindigkeit, Peilung, Höhe und Genauigkeit des GPS.

    Wenn das GPS abgetrennt (oder QGIS geschlossen) wird, wird der gesamte aufgezeichnete GPS-Track in die Tabelle `gps_tracks` eingetragen (zusammen mit einigen berechneten Informationen wie Tracklänge, Start- und Endzeit).
  - ☐ *NMEA-Sätze protokollieren…*: ermöglicht die Aufzeichnung aller NMEA-Rohdaten des Geräts in eine Textdatei
  - 🔧 *GPS Einstellungen…* für den Zugriff auf den GPS-Reiter in den *globalen Optionen*

> **Tipp**
>
> **Live-Statusleiste mit Informationen zur Position**
>
> Wenn ein GPS-Gerät angeschlossen ist und der Benutzer den Cursor über die Kartenansicht bewegt, zeigt eine Live-Statusleiste die Entfernung und Peilung vom Cursor zur GPS-Position an. Die Einstellungen für Projektentfernung und -peilung werden in dieser Anzeige berücksichtigt. Auf einem Touchscreen-Gerät können Sie die Meldung in der Statusleiste durch Antippen und Halten auslösen.

20.2.2 Das Bedienfeld GPS-Information

Für die vollständige Überwachung eines Live-GPS-Trackings in QGIS müssen Sie möglicherweise das Bedienfeld *GPS-Infomation* aktivieren (*Ansicht* ► *Bedienfelder* ► oder drücken Sie `Strg+0`).

In der oberen rechten Ecke des Bedienfelds *GPS-Information* können Sie QGIS durch Klicken mit einem angeschlossenen GPS-Empfänger *Verbinden* oder die Verbindung zu diesem wieder *Trennen*.

Oben links auf dem Bedienfeld sind folgende Schaltflächen zugänglich:

- Position: Live-Details zur GPS-Position und zu den Sensoren
- Signal: Signalstärke der Satellitenverbindung
- **|Optionen|** Einstellungen: Dropdown-Menü für *Live-Tracking-Optionen*, die Sie während einer Sitzung eventuell ändern müssen

Positionskoordinaten und weitere Attribute

Auf der Registerkarte **|Metadaten|** Position sehen Sie, wenn das GPS Signale von Satelliten empfängt, Ihre Position in geografischer Breite, Länge und Höhe zusammen mit zusätzlichen Attributen.

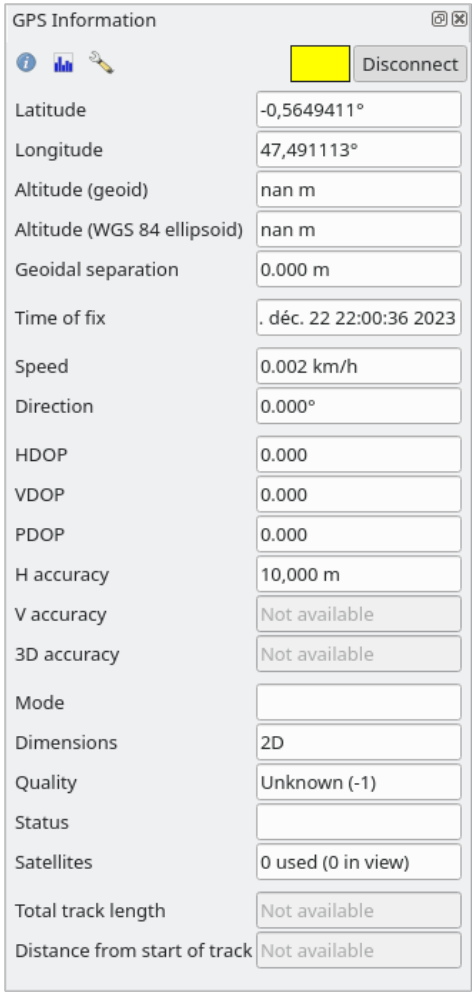


Abb. 20.3: GPS Tracking Position und zusätzliche Attribute

- *Breitengrad*
- *Längengrad*

- *Höhe (geoid)*: Höhe/Höhenlage über oder unter dem mittleren Meeresspiegel
- *Höhe (WGS 84-Ellipsoid)*: Höhe/Höhenlage über oder unter dem WGS-84-Erdellipsoid
- *Geoidale Trennung*: Differenz zwischen dem WGS-84-Erdellipsoid und dem mittleren Meeresspiegel (Geoid), − bedeutet, dass der mittlere Meeresspiegel unter dem Ellipsoid liegt
- *Zeit des Fix*
- *Geschwindigkeit*: Geschwindigkeit über Grund
- *Richtung*: Peilung, gemessen in Grad im Uhrzeigersinn vom geografischen Norden zur Fahrtrichtung
- *HDOP*: Horizontale Verringerung der Genauigkeit (‚Horizontal dilution of precision')
- *VDOP*: Vertikale Verringerung der Genauigkeit (‚Vertical dilution of precision')
- *PDOP*: Verringerung der Positionsgenauigkeit (‚Positional Dilution of precision')
- *H-Genauigkeit*: Horizontale Genauigkeit in Metern
- *V-Genauigkeit*: Vertikale Genauigkeit in Metern
- *3D Genauigkeit*: 3D Root Mean Square (RMS) in Metern
- *Modus*: GPS-Empfängerkonfiguration 2D/3D-Modus, kann `automatisch` oder `manuell` sein
- *Dimensionen*: Positionsfixierung, kann `2D`, `3D` oder `Keine Fixierung` sein
- *Qualität*: Indikator für die Positionierungsqualität
- *Status*: Status der Positionsbestimmung, kann `Gültig` oder `Ungültig` sein
- *Satelliten*: Anzahl der Satelliten, die für die Ermittlung des Fixes verwendet wurden
- *Spurengesamtlänge*: Gesamtlänge des aktuellen GPS-Tracks
- *Entfernung zum Spurbeginn*: Direkte Entfernung vom ersten Punkt des aktuellen GPS-Tracks zum letzten Punkt

Signal

Mit Signal können Sie die Signalstärke der Satelliten sehen, von denen Sie Signale empfangen.

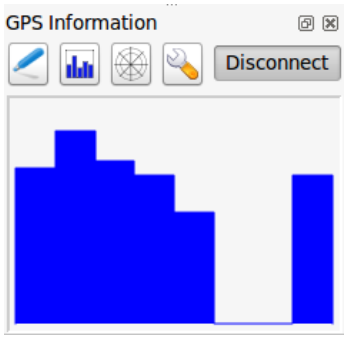


Abb. 20.4: GPS Tracking Signalstärke

20.2.3 Ein Bluetooth GPS fürs Live Tracking anbinden

Mit QGIS können Sie ein Bluetooth GPS für das Aufnehmen von Felddaten anbinden. Um dies durchzuführen, benötigen Sie ein GPS-Gerät mit Bluetooth-Funktion und einen Bluetooth Empfänger an Ihrem Computer.

Als Erstes müssen Sie Ihr GPS Gerät über Bluetooth mit dem Computer verbinden. Stellen Sie das GPS an, gehen Sie zum Bluetooth Icon in Ihrem Infobereich und suchen Sie nach einem neuen Gerät.

Vergewissern Sie sich, dass auf der rechten Seite der Geräteauswahlmaske alle Geräte ausgewählt sind, damit Ihr GPS-Gerät auch unter den verfügbaren Geräten angezeigt wird. Im nächsten Schritt sollte ein serieller Verbindungsdienst verfügbar sein, wählen Sie ihn aus und klicken Sie auf die Schaltfläche *Konfigurieren*.

Merken Sie sich die Nummer des COM-Ports, der der GPS-Verbindung zugewiesen ist, wie sie den Bluetooth-Eigenschaften zu entnehmen ist.

Machen Sie die Kopplung für die Verbindung nachdem das GPS erkannt wurde. Meist ist der Authorisationskode `0000` oder `1234`.

Öffnen Sie nun das Bedienfeld *GPS-Information* und wechseln Sie zum Bildschirm **|Optionen|** GPS-Optionen. Wählen Sie den COM-Port, der der GPS-Verbindung zugewiesen ist, und klicken Sie auf *Verbinden*. Nach einer Weile sollte ein Cursor erscheinen, der Ihre Position anzeigt.

Wenn QGIS keine GPS Daten empfangen kann, sollten Sie Ihr GPS Gerät neustarten, 5-10 Sekunden warten und dann wieder eine Verbindung versuchen. Normalerweise funktioniert diese Lösung. Wenn Sie wieder einen Verbindungsfehler erhalten, vergewissern Sie sich, dass kein anderer Bluetoothempfänger, der an die gleiche GPS Einheit gekoppelt ist, in Ihrer Nähe ist.

20.2.4 Beispiele für die Anbindung von GPS-Geräten in QGIS

GPSMAP 60cs verwenden

MS Windows

Der einfachste Weg, dies zu erreichen, ist die Verwendung einer Middleware (Freeware, nicht offen) namens GPSGate.

Starten Sie das Programm, lassen Sie es nach GPS-Geräten suchen (funktioniert sowohl für USB- als auch für BT-Geräte) und dann in QGIS:

1. In *Einstellungen* ► *Optionen* ► *GPS*, erkennen Sie das eingesteckte Gerät. Sie können den Modus *Autodetect* verwenden.
2. Drücken Sie *Verbinden* im Bedienfeld *GPS-Information*.

Ubuntu/Mint GNU/Linux

Wie unter Windows ist der einfachste Weg einen Server, in diesem Fall GPSD, dazwischen zu benutzen, also:

1. Installieren Sie das Programm:

   ```
   sudo apt install gpsd
   ```

2. Laden Sie dann das `garmin_gps` Kernelmodul

   ```
   sudo modprobe garmin_gps
   ```

3. Dann schließen Sie das Gerät an und überprüfen mit `dmesg` das tatsächliche Laufwerk, das von dem Gerät benutzt wird, zum Beispiel `/dev/ttyUSB0`.
4. Jetzt können Sie gpsd starten

```
gpsd /dev/ttyUSB0
```

5. Und verbinden Sie sich zuletzt mit dem QGIS Live Tracking Tool.

BTGP-38KM Datenlogger verwenden (nur Bluetooth)

Sie können GPSD (unter Linux) oder GPSGate (unter Windows) mühelos verwenden.

BlueMax GPS-4044 Datenlogger verwenden (sowohl BT als auch USB)

MS Windows

Das Live-Tracking funktioniert sowohl für den USB- als auch für den BT-Modus, mit GPSGate oder auch ohne GPSGate, indem Sie einfach den Modus *Autodetect* verwenden oder das Tool auf den richtigen Port richten.

Ubuntu/Mint GNU/Linux

Für USB

Das Live Tracking funktioniert sowohl mit GPSD

```
gpsd /dev/ttyACM3
```

alsauch ohne es, indem man das QGIS Live Tracking Tool direkt mit dem Gerät verbindet (zum Beispiel `/dev/ttyACM3`).

Für Bluetooth

Das Live Tracking funktioniert sowohl mit GPSD

```
gpsd /dev/rfcomm0
```

oder ohne es, indem man das QGIS Live Tracking Tool direkt mit dem Gerät verbindet (zum Beispiel `/dev/rfcomm0`).

KAPITEL 21

Authentifizierungssystem

21.1 Authentifizierungssystem Übersicht

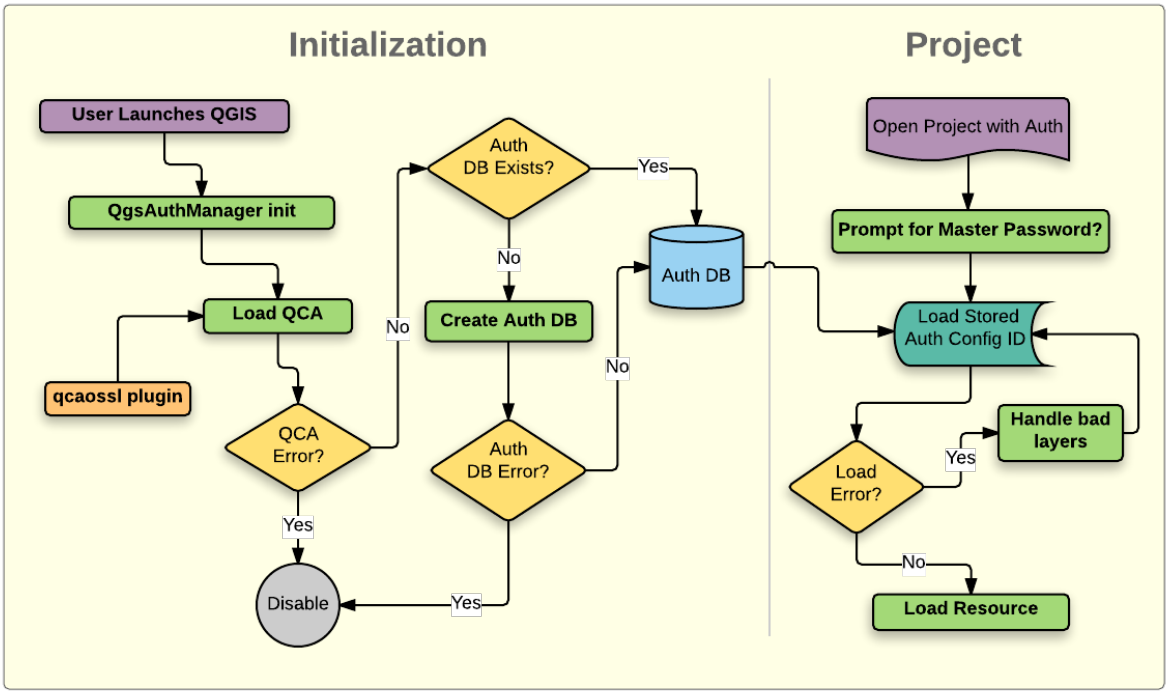


Abb. 21.1: Anatomie des Authentifizierungssystems

21.1.1 Authentifikationsdatenbank

The authentication system stores authentication configurations in an SQLite database file located, by default, at `<profile directory>/qgis-auth.db`.

Diese Authentifizierungsdatenbank kann ohne Auswirkung auf andere aktuelle QGIS Benutzereinstellungen zwischen QGIS Installationen bewegt werden, da sie völlig unabhängig von normalen QGIS-Einstellungen ist. Eine Konfigurations-ID (eine zufällige 7-stellige alphanumerische Zeichenfolge) wird erzeugt, wenn eine Konfiguration zu der Datenbank gespeichert wird. Dies stellt die Konfiguration dar, wodurch die ID im Klartext als Anwendungskomponenten gespeichert wird (z. B. Projekt-, Plugin- oder Einstellungsdateien) ohne Offenlegung ihrer zugehörigen Anmeldeinformationen.

Bemerkung

Das übergeordnete Verzeichnis des *qgis-auth.db`kann mit der folgenden Umgebungsvariablen gesetzt werden, `QGIS_AUTH_DB_DIR_PATH`*, oder mit der `--authdbdirectory` Option während des Starts auf der Kommandozeile.

21.1.2 Custom authentication databases

QGIS can be configured to use a custom authentication database instead of the above mentioned default SQLite one: any database suppported by the Qt SQL module can be used (e.g. PostgreSQL, MySQL, etc), provided that the corresponding Qt SQL driver is available in the system.

This can be useful in scenarios where a user wants to share the same authentication database between multiple QGIS installations, or when a user wants to use a different authentication database than the default SQLite one or when a centralized authentication database is used by QGIS server.

The only way to configure a custom authentication database is by setting the `QGIS_AUTH_DB_URI` environment variable to the URI of the connection, the URI is in the form of `driver://username:password@hostname:port/database?options`.

Where:

- `driver` is the name of the Qt SQL driver to use, e.g. `QPSQL` for PostgreSQL, `QMYSQL` for MySQL, etc.
- `username` is the username to use to connect to the database
- `password` is the password to use to connect to the database
- `hostname` is the hostname of the database server
- `port` is the port of the database server
- `database` is the name of the database to use
- `options` are the options to pass to the driver, e.g. `sslmode=require` for PostgreSQL

Bemerkung

The `schema` can be specified in the URI options, e.g. `QPSQL://username:password@hostname:port/database?schema=schema_name`

The database must exist before starting QGIS, and the user must have the necessary permissions to connect to the database and to create the required tables if they do not exist.

Warnung

The password in the URI is stored in plain text in the environment variable, so it is recommended to use a passwordless user or a user with limited permissions to connect to the database.

Warnung

Any database not based on SQLite is considered to be read-only (this can be changed by Python plugins if necessary).

This is an advanced feature, designed to allow one or more custom authentication databases or even custom Python implementations of credentials storages to be used by QGIS.

The system is also designed to allow for multiple authentication databases to be used but there is currently no user facing interface to manage multiple credential storages so its usage requires manual configuration and management, typically from a Python plugin.

21.1.3 Hauptpasswort

Um vertrauliche Informationen in der Datenbank zu speichern oder darauf zuzugreifen, muss ein *Master Passwort* eingerichtet werden. Ein neues Master-Passwort wird angefordert und verifiziert, wenn erstmals irgendwelche verschlüsselten Daten in der Datenbank gespeichert werden. Wenn auf sensible Informationen zugegriffen wird, wird der Benutzer zur Eingabe des Master-Kennworts aufgefordert. Das Passwort wird dann für den Rest der Sitzung zwischengespeichert (bis die Anwendung beendet wird), es sei denn, der Benutzer wählt manuell eine Aktion aus, um den zwischengespeicherten Wert zu löschen. Bei einigen Vorgängen ist es nich nötig, das Master-Passworts einzugeben - z. B. bei der Auswahl einer vorhandenen Authentifizierungskonfiguration oder beim Anwenden einer Konfiguration auf eine Serverkonfiguration (wie z. B. beim Hinzufügen eines WMS-Layers).

Sie können das Passwort in dem `Wallet/Keyring` Ihres Computer speichern.

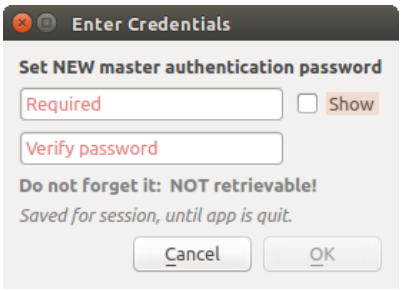


Abb. 21.2: Neues Hauptpasswort eingeben

Bemerkung

Ein Pfad zu einer Datei, die das Hauptpasswort beinhalet, kann mit Hilfe der Umgebungsvariablen eingestellt werden `QGIS_AUTH_PASSWORD_FILE`.

Hauptpasswort verwalten

Einmal eingestellt, kann das Hauptpasswort zurückgesetzt werden, das aktuelle Hauptpasswort wird vor dem Zurücksetzen jedoch benötigt werden. Während dieses Prozesses ist es eine Option, eine vollständige Sicherung der aktuellen Datenbank vorzunehmen.

Abb. 21.3: Hauptpasswort zurücksetzen

Wenn der Benutzer das Hauptpasswort vergisst, gibt es keine Möglichkeit es abzurufen oder außer Kraft setzen. Es gibt auch keine Mittel die verschlüsselten Informationen abzurufen ohne das Master-Passwort zu kennen.

Wenn ein Benutzer das bestehende Password drei Mal falsch eingibt, wird ein Dialog Ihnen anbieten, die Datenbank zu löschen.

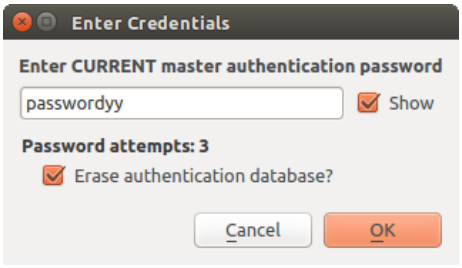


Abb. 21.4: Passwortabfrage nach drei ungültigen Versuchen

21.1.4 Authentifikationskonfiguration

Sie können Authentifizierungseinstellungen über *Konfigurationen* in dem *Authentifizierung* Reiter des QGIS Optionen Dialog (*Einstellungen► Optionen*) vornehmen.

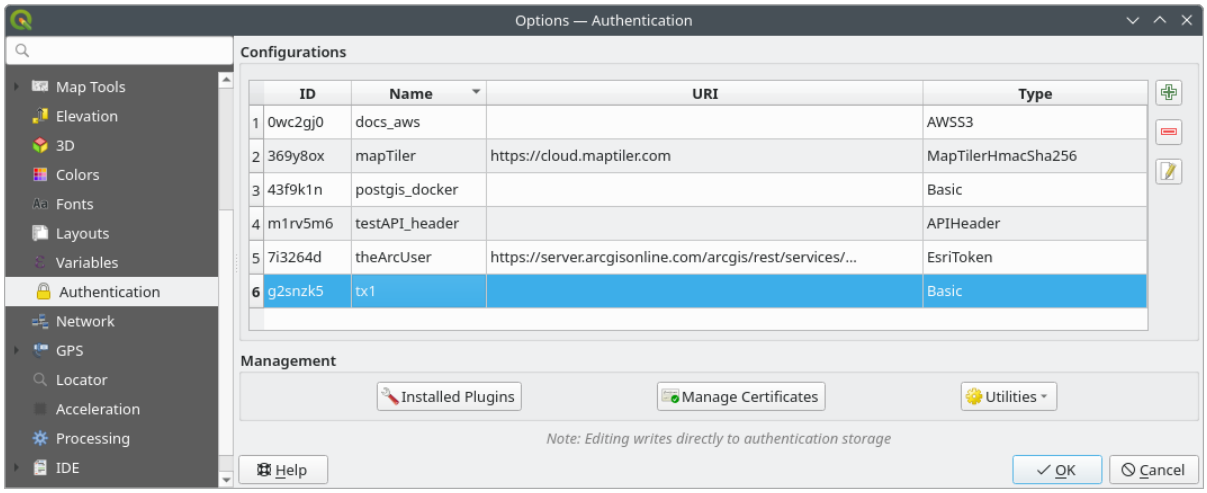


Abb. 21.5: Konfigurationen Editor

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.

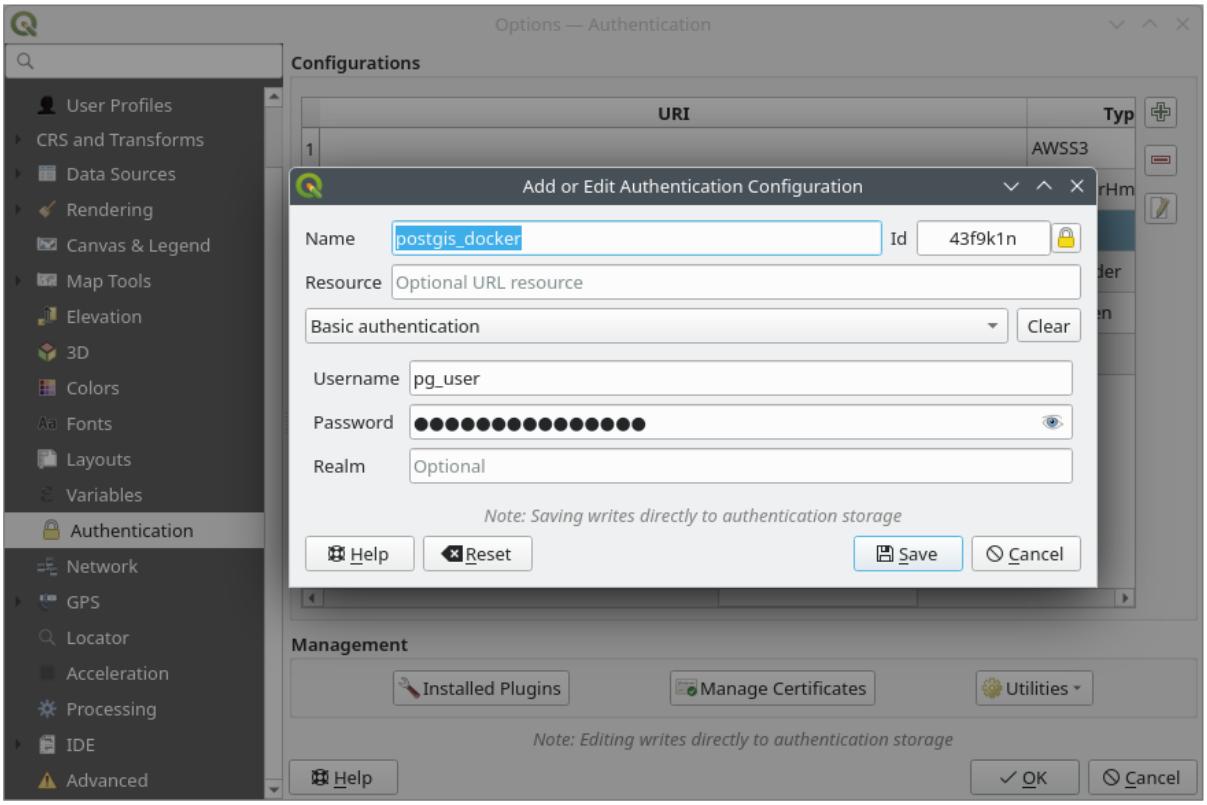


Abb. 21.6: Konfigurationen von innerhalb des Konfigurationseditors hinzufügen

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.

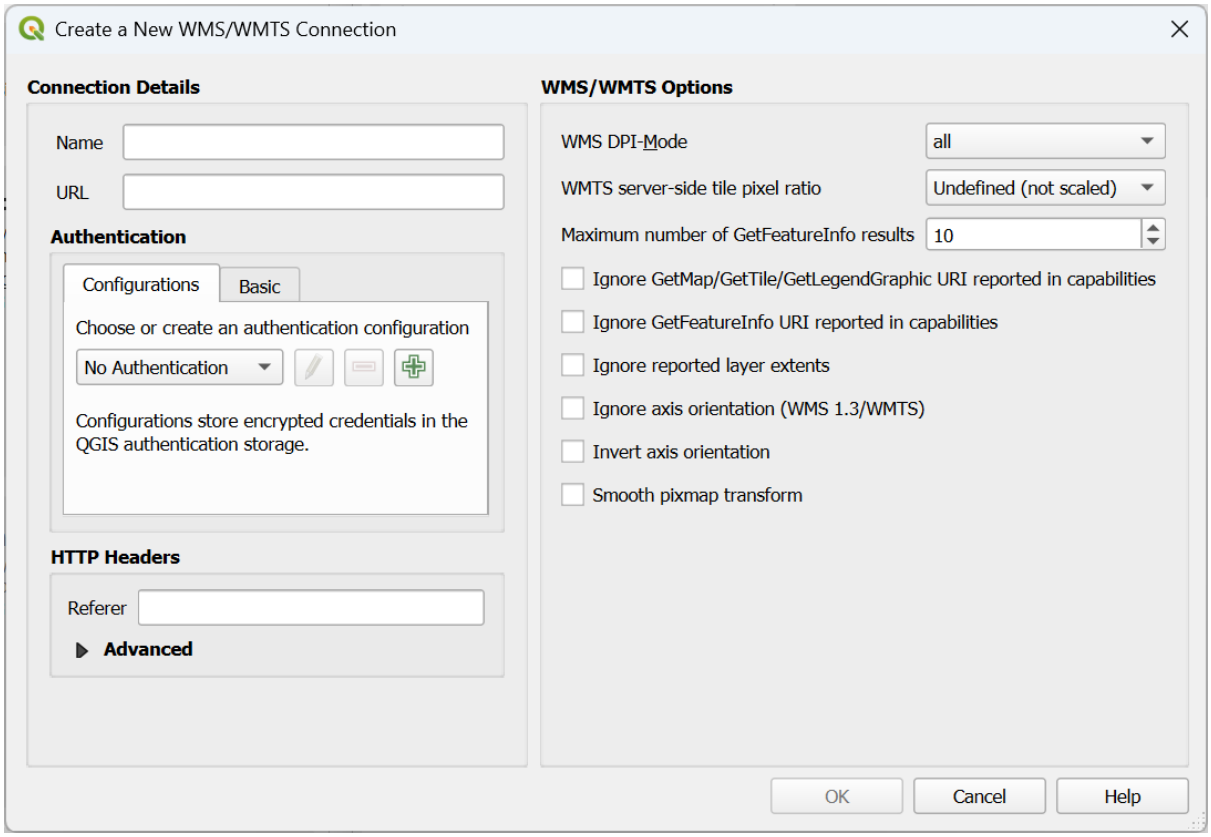


Abb. 21.7: WMS connection dialog showing authentication configuration widget

Beim erstellen oder bearbeiten einer Authentifikationskonfiguration, ist ein Name, eine Authentifizierungsmethode oder eine andere Information, die die Authentifizierungsmethode benötigt, erforderlich (mehr über die verfügbaren Authentifikationstypen in *Authentifizierungsmethoden*).

21.1.5 Authentifizierungsmethoden

Verfügbare Authentifizierungen werden von C++-Plugins auf die gleiche Weise bereitgestellt wie Daten Provider-Plugins von QGIS unterstützt werden. Die Authentifizierungsmethode, die ausgewählt werden kann, ist relativ zu dem Zugriff, der für die Ressource/den Anbieter benötigt wird, z. HTTP(S) oder Datenbank, und ob es Unterstützung sowohl im QGIS-Code als auch in einem Plugin gibt. Daher sind einige Plugins für Authentifizierungsmethoden möglicherweise nicht überall dort, wo ein Auswahlschalter für die Authentifizierungskonfiguration angezeigt wird, anwendbar. Eine Liste von verfügbaren Plugins für Authentifizierungsmethoden und deren kompatible Ressourcen/Anbieter kann aufgerufen werden, indem Sie zu *v* ► *Optionen* navigieren und in der Registerkarte *Authentifizierung* auf die Schaltfläche *Installierte Plugins* klicken.

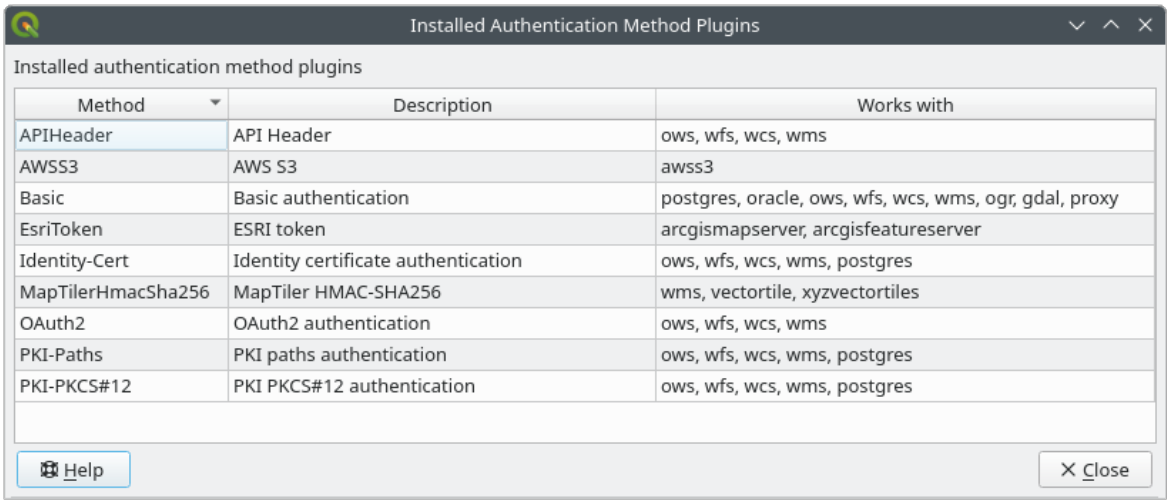


Abb. 21.8: Verfügbare Liste von Methodenerweiterungen

Plugins für neue Authentifizierungsmethoden können erstellt werden, wenn QGIS dadurch nicht neu kompiliert werden muss. Da die Unterstützung für Plugins derzeit nur C++ ist, muss QGIS neu gestartet werden, damit das neue Dropped-In-Plugin dem Benutzer zur Verfügung steht. Stellen Sie sicher, dass Ihr Plugin für dieselbe Ziel-Version von QGIS kompiliert wird, zu der es hinzugefügt werden soll.

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.

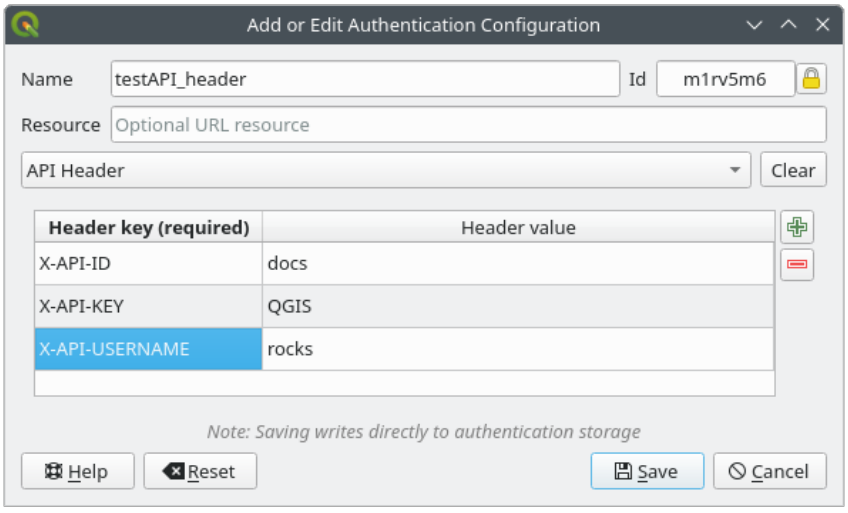


Abb. 21.9: API Header authentication configs

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*.

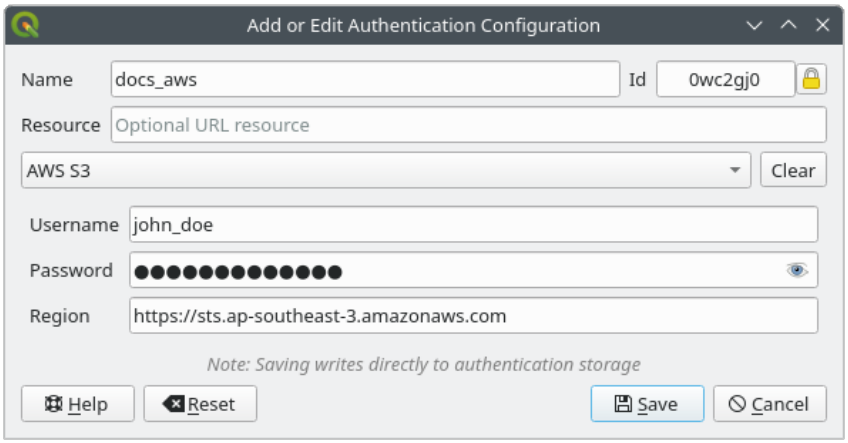


Abb. 21.10: AWS S3 authentication configs

Basic Authentication

The Basic authentication method is used for services that require standard HTTP authentication with a *Username*, *Password* and *Realm*.

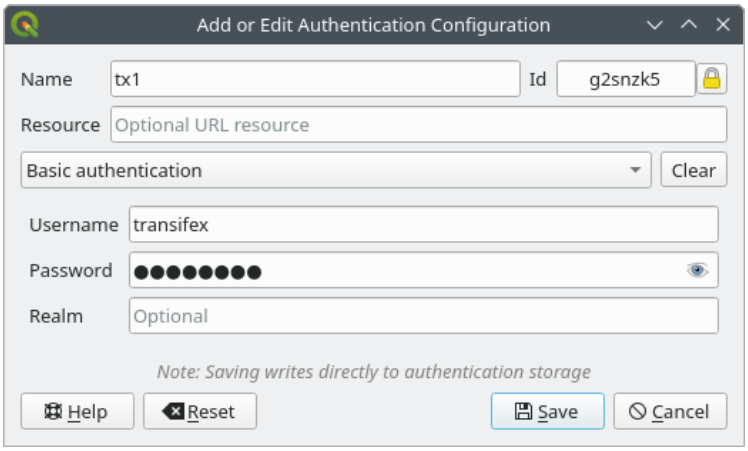


Abb. 21.11: Einfache HTTP Authentifikationskonfiguration

ESRI Token Authentication

The ESRI token authentication method is used for *ArcGIS REST Servers* that require token-based authentication.

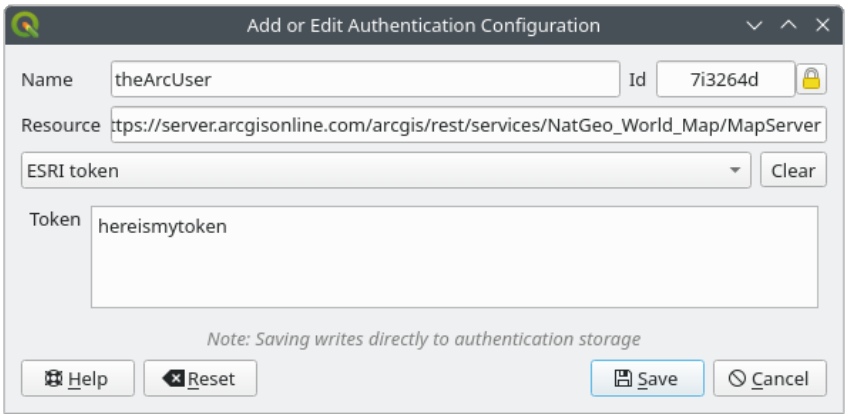


Abb. 21.12: Konfigurationen für die ESRI-Token-Authentifizierung

Identity certificate authentication

The Identity certificate authentication method allows you to connect using a client *identity certificate*.

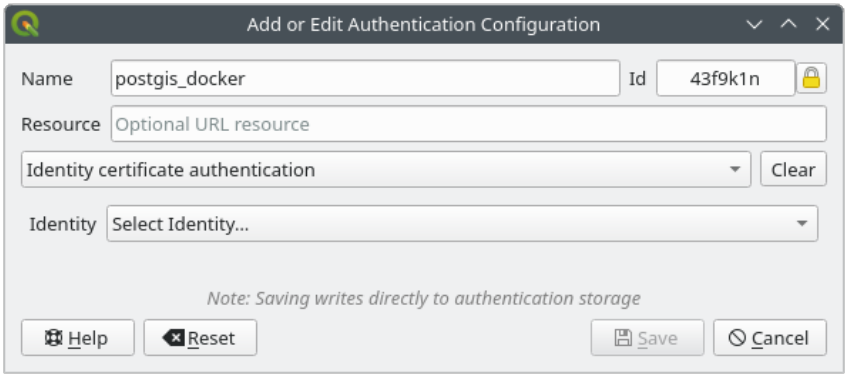


Abb. 21.13: Identity-cert authentication configs

MapTiler HMAC-SHA256 Authentication

The MapTiler HMAC-SHA256 authentication method is used to connect to MapTiler services that require HMAC-SHA256 authorization.

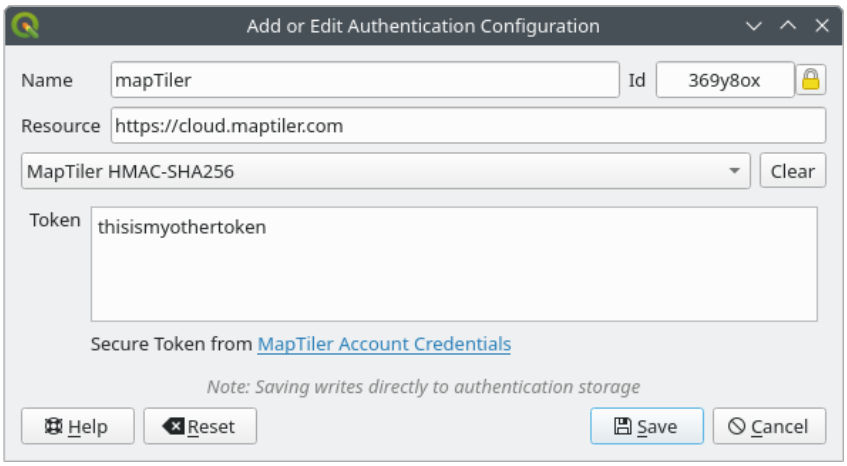


Abb. 21.14: MapTiler HmacSha256 authentication configs

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.

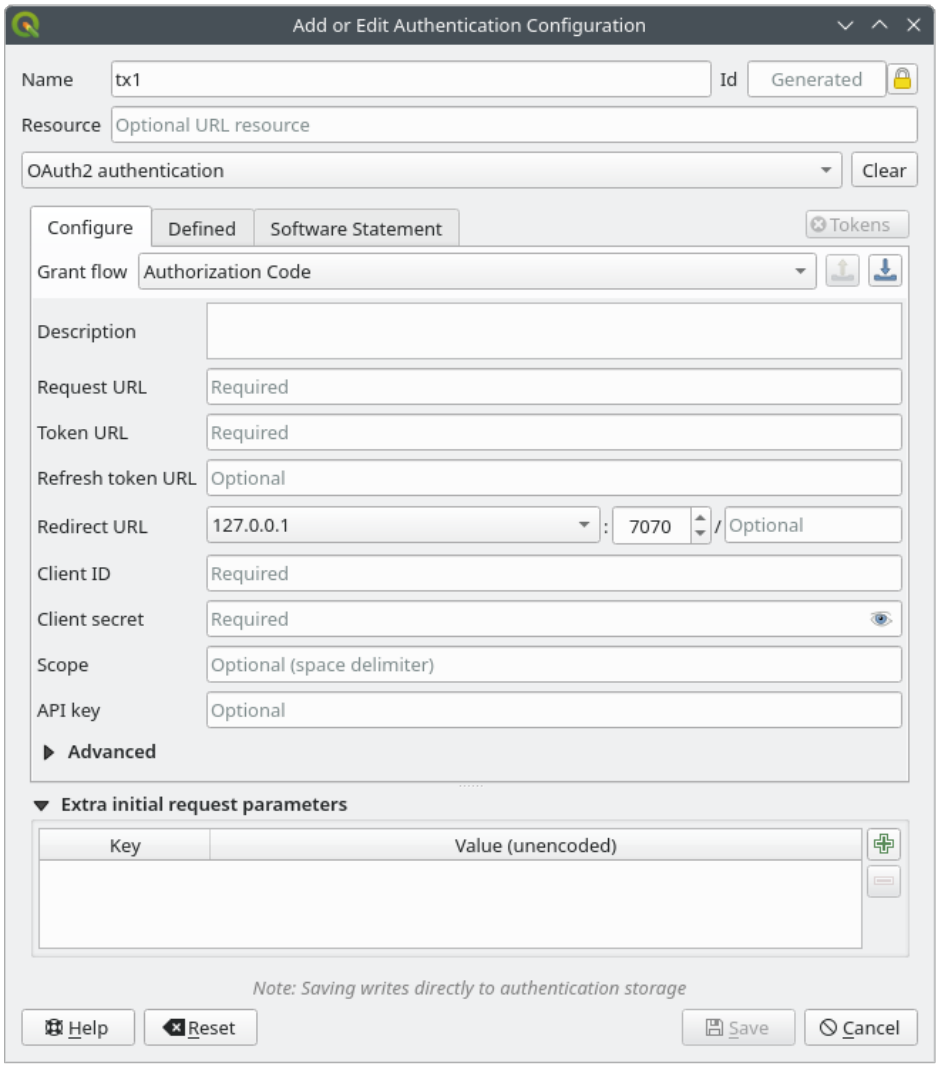


Abb. 21.15: OAuth2-Authentifizierungskonfigurationen

PKI-Paths Authentication

The PKI-Paths authentication method allows you to connect using separate certificate and key files stored on your system.

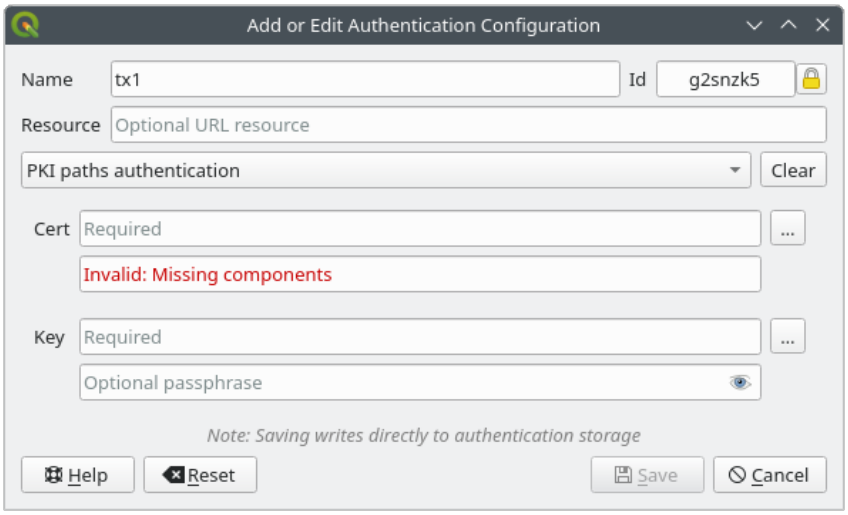


Abb. 21.16: PKI Pfad-Authentifikationskonfiguration

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.

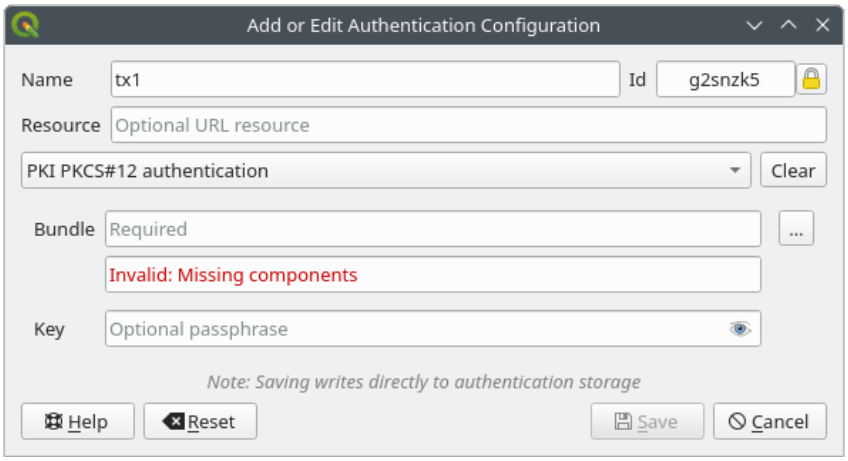


Abb. 21.17: PKI PKCS#12 Dateipfad Authentifikationskonfiguration

21.1.6 Hauptpasswort und Auth Konfig Hilfsmittel

Unter dem Optionen-Menü (*Einstellungen* ► *Optionen*) in der *Authentifikation* Registerkarte gibt es verschiedene Werkzeuge die Authentifikationsdatenbank und -konfigurationen zu verwalten:

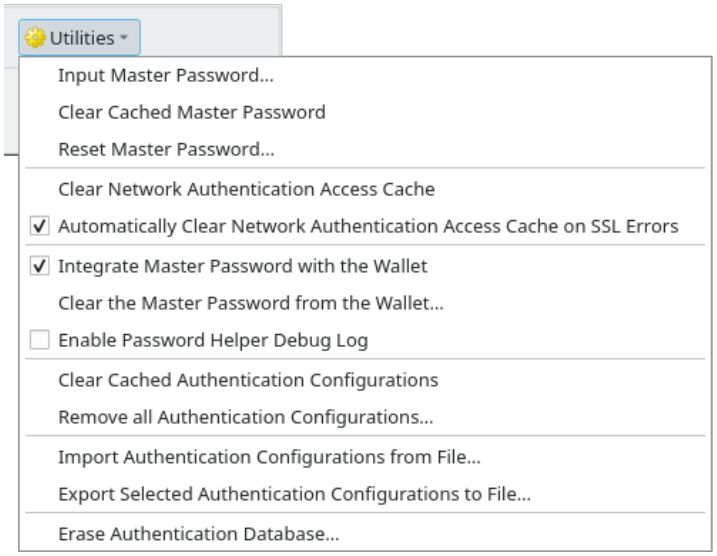


Abb. 21.18: Werkzeug-Menü

- *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 Gebrauche Authentifikationskonfiguration

Typischerweise wird eine Authentifizierungskonfiguration in einem Konfigurationsdialog für ein Netzwerk-Dienste (wie WMS) ausgewählt. Allerdings kann das Auswahl-Widget überall, wo Authentifizierung benötigt wird, eingebettet werden oder in einer Nicht-Kernfunktionalität, wie in Dritt-Beteiligten PyQGIS oder C ++ Plugins.

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 *Authentifizierung*.

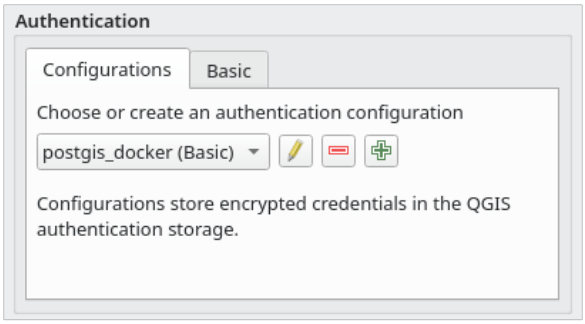


Abb. 21.19: Authentifizierungskonfigurations-Selektor mit ausgewählter Konfiguration

21.1.8 Python-Bindung

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.

Sobald das Master-Passwort eingegeben wurde, ist die API für den Zugriff auf die Authentifizierungs-Konfigurationen in der Authentifizierungsdatenbank geöffnet, ähnlich wie Firefox funktioniert. Es wurde jedoch keine Mauer gegen den PyQGIS-Zugriff definiert. Dies kann zu Problemen führen, bei denen ein Benutzer ein bösartiges PyQGIS-Plugin herunterlädt/installiert oder eine eigenständige App nutzt, die Zugriff auf Authentifizierungsdaten erhält.

21.2 Benutzerauthentifizierung Workflows

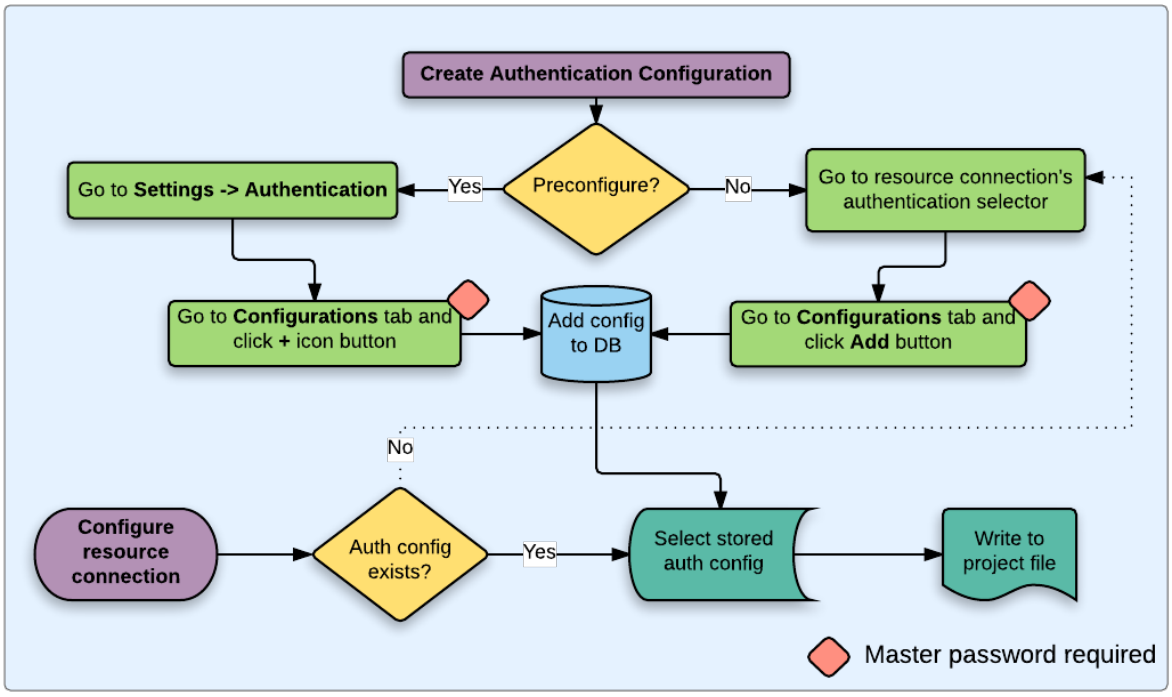


Abb. 21.20: Generischer Benutzer Workflow

21.2.1 HTTP(S) Authentifikation

Eine der häufigsten Verbindungen findet mittels HTTP (S) statt, z.B. bei WebGIS Servern und die Authentifizierungsmethoden werden oft für diese Art von Verbindungen genutzt. Die Methoden haben Zugriff auf das HTTP-Request Objekt und können sowohl den Request als auch seinen Header manipulieren. Dies ermöglicht viele Formen von Internet-basierten Authentifizierungen. Wenn über HTTP(S) mit der Standard Benutzername-/Passwort-Authentifizierungsmethode verwendet wird, wird bei der Verbindung eine HTTP-Basisauthentifizierung versucht.

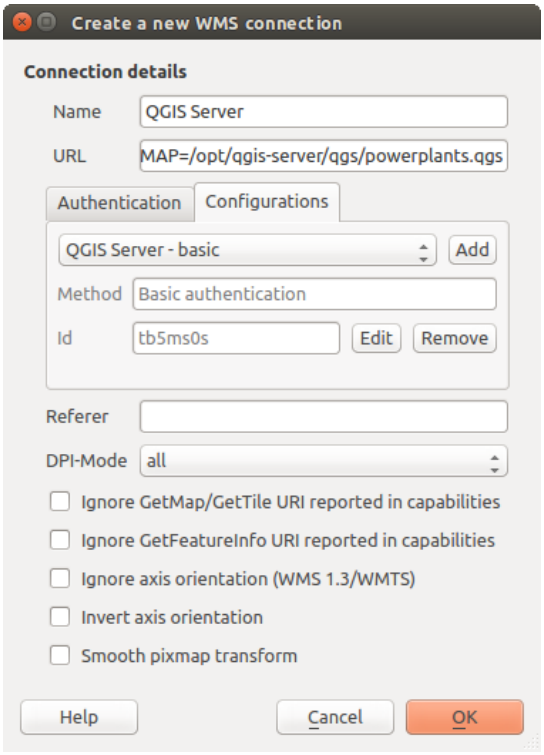


Abb. 21.21: Konfigurieren einer WMS-Verbindung für HTTP BASIC

21.2.2 Datenbank Authentifikation

Verbindungen zu Datenbankressourcen werden in der Regel als `Schlüssel=Wert`-Paare gespeichert, wodurch Benutzernamen und (optional) Passwörter offengelegt werden, wenn *keine* Authentifizierungs-Konfiguration genutzt wird. Bei einer Konfiguration mit dem Authentifizierungssystem wird das `Schlüssel=Wert`-Paar als abstrahierte Darstellung der Anmeldeinformationen angezeigt, z. B. `authfg=81t21b9`.

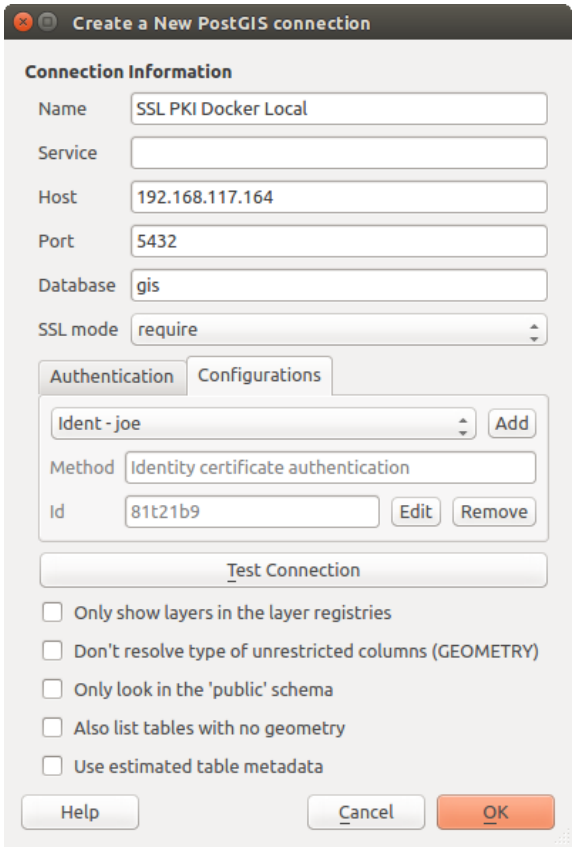


Abb. 21.22: Einstellen einer Postgress-SSL mit PKI-Verbindung

21.2.3 PKI Authentifikation

Wenn PKI-Komponenten innerhalb des Authentifikationssystems konfiguriert werden, haben Sie die Möglichkeit, Komponenten in die Datenbank zu importieren oder Dateien, die auf Ihrem System gespeichert sind, zu referenzieren. Letzteres kann nützlich sein, wenn solche Komponenten sich häufig verändern oder wenn die Dateien durch einen Systemadministrator verschoben werden. In beiden Fällen müssen Sie ein beliebiges Passwort speichern um Zugang zu privaten Schlüsseln innerhalb der Datenbank zu haben.

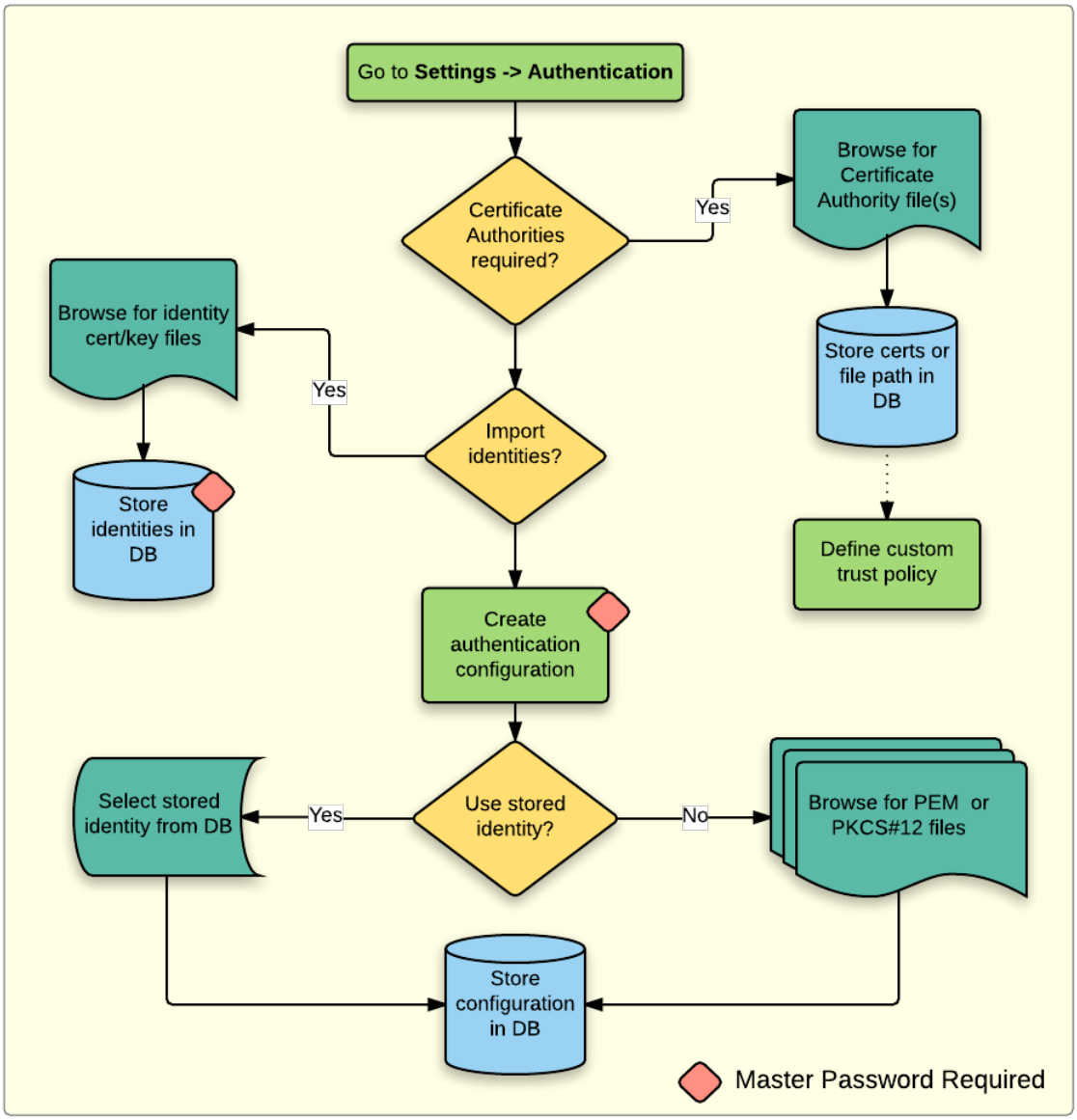


Abb. 21.23: PKI Konfiguration Workflow

Alle PKI-Komponenten können in separaten Editoren innerhalb der **Zertifikatsverwaltung** verwaltet werden. Auf diesen können Sie im QGIS *Optionen*-Dialog (*Einstellungen* ► *Optionen*) über den *Authentifizierung*-Tab durch Klicken auf die Schaltfläche *Zertifikate verwalten* gelangen.

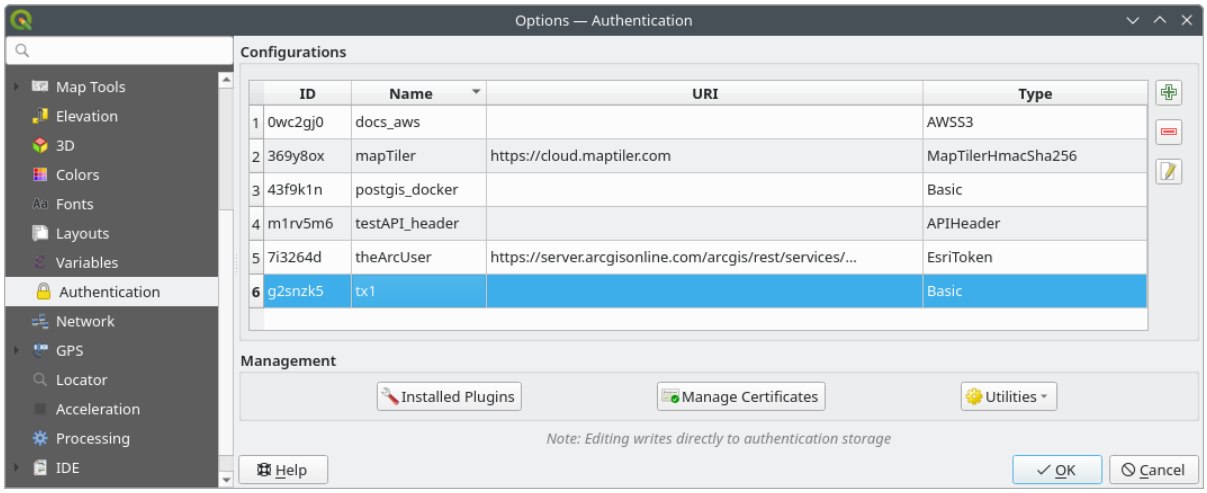


Abb. 21.24: Öffnen der Zertifikatsverwaltung

In der Zertifikatsverwaltung gibt es Editoren für **Identitäten**, **Server** und **Autoritäten**. Jeder hat eine eigene Registerkarte. Sie werden im Folgenden in der Reihenfolge beschrieben, in der sie im Workflow-Diagramm oben auftreten. Die Registerkarten-Reihenfolge bezieht sich auf die Häufigkeit, mit der auf die Editoren zugegriffen wird, sobald Sie an den Workflow gewöhnt sind.

Bemerkung

Da alle Änderungen im Authentifizierungssystem sofort in die Authentifizierungsdatenbank geschrieben werden, ist es nicht erforderlich, auf die Schaltfläche *OK* im Dialogfeld *Optionen* zu klicken, um die Änderungen zu speichern - anders als bei anderen Einstellungen im Dialogfeld Optionen.

Autoritäten

Sie können verfügbare Zertifikat-Autoritäten (ZA) im **Autoritäten**-Tab unter QGIS *Optionen* (*Einstellungen* ► *Optionen*) –> *Authentifizierung*-Tab –> *Zertifikate verwalten* bearbeiten.

Wie oben in dem Workflow-Diagramm Bezug genommen wird, ist der erste Schritt, eine ZA-Datei zu importieren oder eine zuzuweisen. Dieser Schritt ist optional und nicht notwendig, wenn Ihre PKI-Vertrauenskette ihren Ursprung in einer ZA hat, welche bereits auf Ihrem Betriebssystem (OS) installiert ist - wie z.B. ein Zertifikat eines kommerziellen Zertifizierungsanbieters. Falls Ihre Authentifizierungs-Stamm-ZA keine der im OS gespeicherten vertrauenswürdigen ZA ist, müssen Sie sie importieren oder ihren Dateisystempfad referenzieren. (Kontaktieren Sie Ihren Systemadministrator, falls Sie sich unsicher sind.)

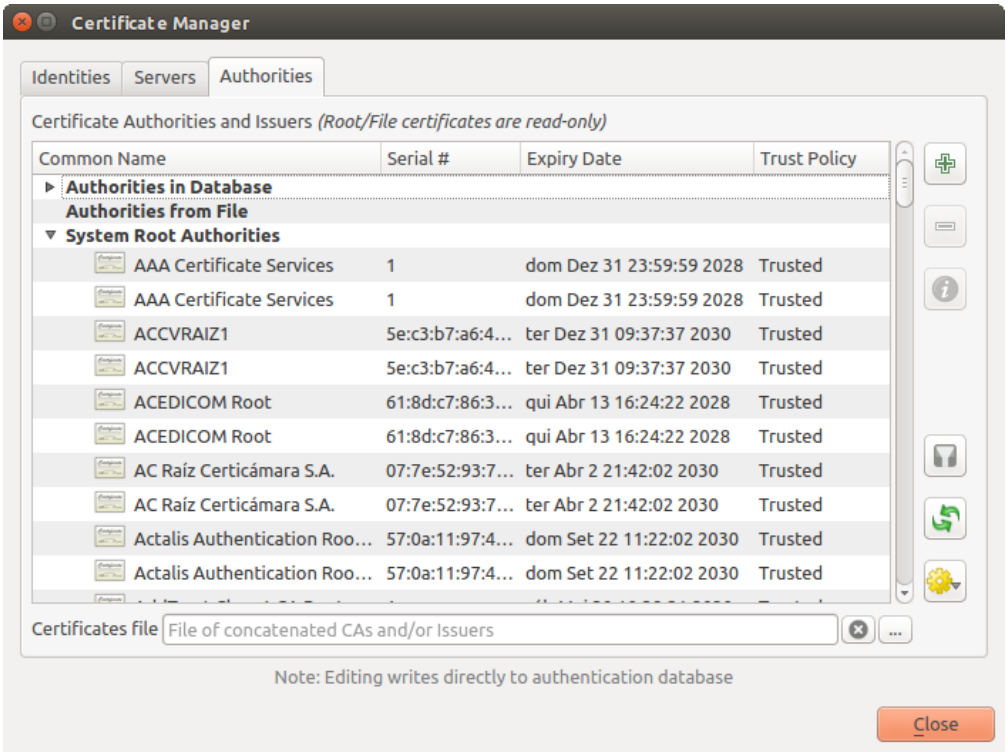


Abb. 21.25: Autoritäteneditor

Standardmäßig sind die Stammzertifizierungs-Autoritäten Ihres Betriebssystems verfügbar. Ihre Vertrauens-Einstellungen werden jedoch nicht vererbt. Sie sollten die Einstellungen für die Zertifikate-Vertrauensregeln überprüfen, insbesondere wenn die Richtlinien der Stammzertifizierungsstellen des Betriebssystems bereits angepasst wurden. Alle abgelaufenen Zertifikate werden auf nicht vertrauenswürdig gesetzt und nicht in sicheren Serververbindungen genutzt - es sei denn, Sie setzen die Vertrauensrichtlinie ausdrücklich außer Kraft. Sehen Sie sich die QGIS-Vertrauenskette für ein beliebiges Zertifikat an, wählen Sie es aus und klicken Sie auf Zertifikatsinformationen anzeigen.

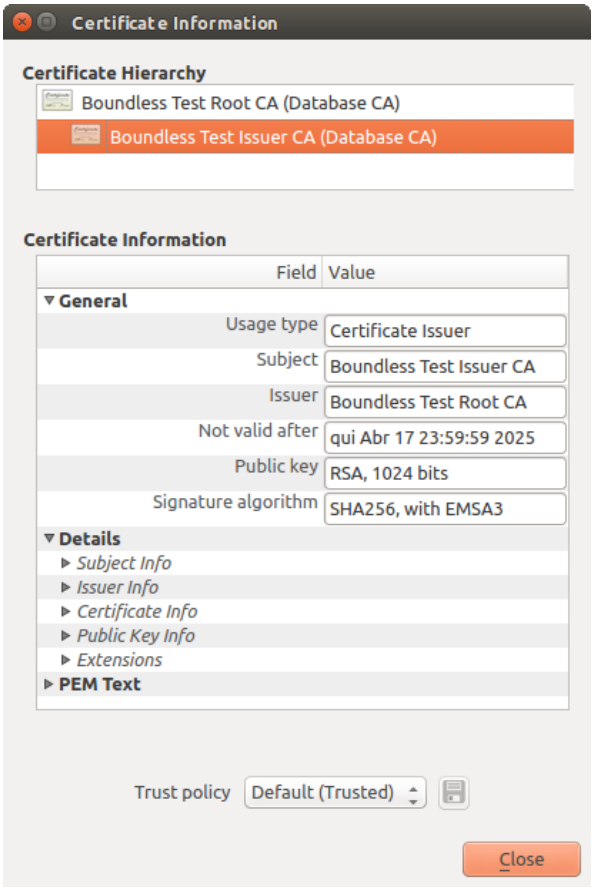


Abb. 21.26: Zertifikatsinformationen-Dialog

Sie können die *Vertrauensrichtlinie* für jedes ausgewählte Zertifikat innerhalb der Kette bearbeiten. Jede Änderung der Vertrauensrichtlinie eines ausgewählten Zertifikats wird nicht in der Datenbank gespeichert, es sei denn, es wird auf Speichern der Änderung der Zertifikatvertrauensrichtlinie *pro* ausgewählter Zertifizierung geklickt. Einfaches Schließen des Dialogfelds speichert **nicht** die Änderung der Richtlinie.

Sie können die gefilterten ZA überprüfen, sowohl die Zwischen- als auch die Stammzertifikate, denen durch Klicken auf sichere Verbindungen getraut wird, oder die Standard-Vertrauensrichtlinie unter **Optionen** ändern.

Warnung

Die Standard-Vertrauensrichtlinie ändern kann zu Problemen mit sicheren Verbindungen führen.

Sie können einzelne ZA importieren oder den Dateisystempfad einer Datei speichern, welche mehrere ZA enthält. Das Standard-PEM-Format für Dateien, welche mehrere ZA-Kettenzertifikate enthalten, hat das Stammzertifikat am Ende der Datei und alle anschließend unterzeichneten Unter-Zertifikate oben, am Anfang der Datei.

Der ZA Zertifikat-Import-Dialog wird alle ZA-Zertifikate innerhalb einer Datei finden, unabhängig von der Reihenfolge. Es gibt auch die Möglichkeit Zertifikate zu importieren, die als ungültig betrachtet werden (falls Sie deren Vertrauensrichtlinien überschreiben möchten). Sie können die Vertrauensrichtlinie nach dem Import überschreiben oder später im **Autoritäten**-Editor.

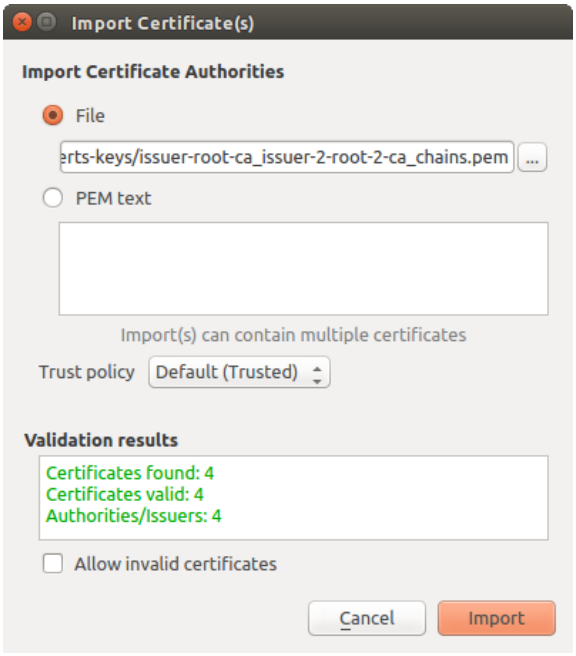


Abb. 21.27: Zertifikat importieren-Dialog

Bemerkung

Wenn Sie Zertifikatinformationen in das *PEM-Text*-Feld einfügen, beachten Sie, dass verschlüsselte Zertifikate nicht unterstützt werden.

Identitäten

Sie können verfügbare Client-Identitäts-Bündel über den Reiter *Identitäten* des *Zertifikat Manager* im Reiter **Authentifizierung** im Dialog QGIS **Optionen**. Eine Identität authentifiziert Sie gegen einen PKI-fähigen Dienst und besteht in der Regel aus einem Client-Zertifikat und einen privaten Schlüssel, entweder als separate Dateien oder kombiniert in einer einzigen „gebündelten“ Datei. Das Bündel oder private Schlüssel sind oft Passphrase geschützt.

Sobald Sie alle Zertifizierungsautoritäten (ZA) importiert haben, können Sie optional alle Identitätsbündel in die Authentifizierungsdatenbank importieren. Wenn Sie die Identitäten nicht speichern möchten, können Sie ihre Komponentendateisystempfade innerhalb einer einzelnen Authentifizierungskonfiguration referenzieren.

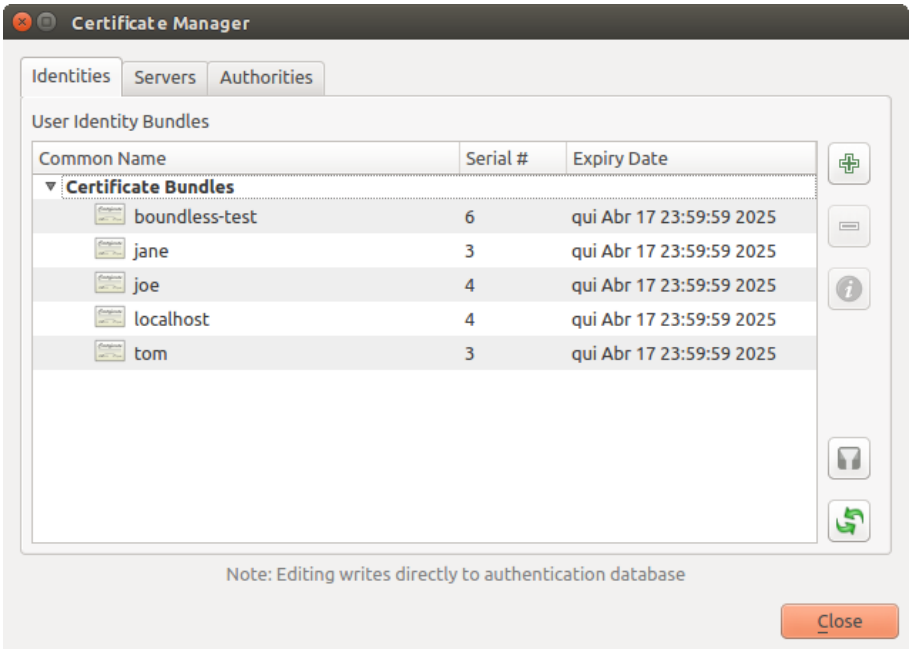


Abb. 21.28: Identitäteneditor

Wenn ein Identitäts-Bündel importieren, können diese Passwort-geschützt oder ungeschützt sein und ZA-Zertifikate enthalten, die eine Vertrauenskette bilden. Vertrauenskette Zertifizierungen werden nicht an dieser Stelle importiert; sie können separat unter dem Reiter guilabel:*Authoritäten* hinzugefügt werden.

Beim Import wird das Zertifikats und der Private Key eines Bündels in der Datenbank gespeichert, wobei der Speicher des Schlüssels über das QGIS-Master-Passwort verschlüsselt ist. Die anschließende Verwendung des gespeicherten Bündels aus der Datenbank erfordert dann nur Eingabe des QGIS-Master-Passworts.

Persönliche Identitäts-Bündel bestehend aus PEM/DER (.pem / .der) und PKCS#12 (.p12/.pfx) Komponenten werden unterstützt. Wenn ein Schlüssel oder ein Bündel passwortgeschützt ist, wird das Passwort benötigt, um die Komponente zu validieren, bevor sie importiert wird. Wenn das Client-Zertifikat in dem Bündel ungültig ist (zum Beispiel ist das Datum des Inkrafttretens noch nicht gestartet oder abgelaufen), kann das Bündel nicht importiert werden.

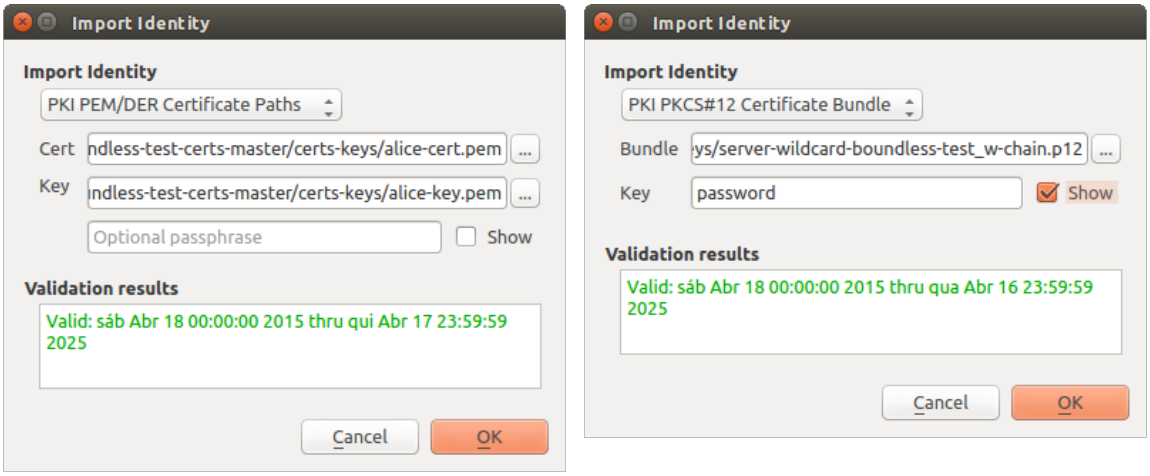


Abb. 21.29: PEM/DER and PKCS#12 identities import

21.2.4 Defekte Layer behandeln

Gelegentlich kommt es vor, dass die ID der Authentifizierungs-Konfiguration, die mit einer Projektdatei gespeichert wird, nicht mehr gültig ist, möglicherweise, weil die aktuelle Authentifizierungsdatenbank sich geändert hat, seitdem das Projekt zuletzt gespeichert wurde, oder aufgrund einer fehlenden Übereinstimmung der Anmeldeinformationen. In solchen Fällen öffnet sich der Dialog *Defekte Layer behandeln* beim Starten von QGIS.

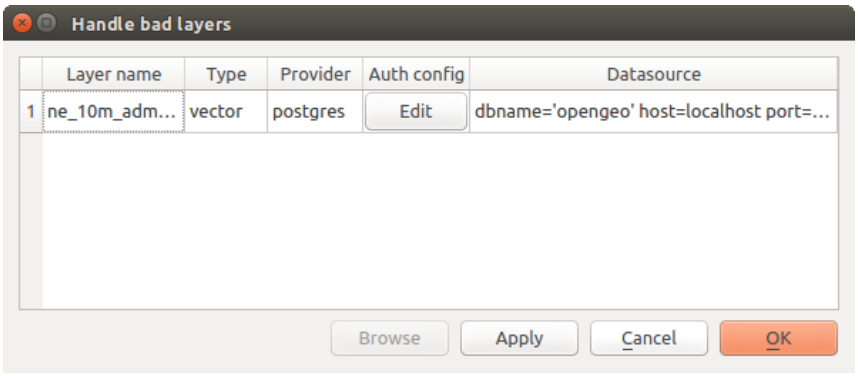


Abb. 21.30: Defekte Layer mit Authentifizierung behandeln

Wenn eine Datenquelle eine ihr zugewiesene Authentifizierungs Konfigurations-ID besitzt, können Sie diese bearbeiten. Dadurch wird der Pfad der Datenquelle automatisch geändert, so wie es beim Öffnen der Projektdatei mit einem Texteditor und dem bearbeiten von Zeichenfolgen passiert.

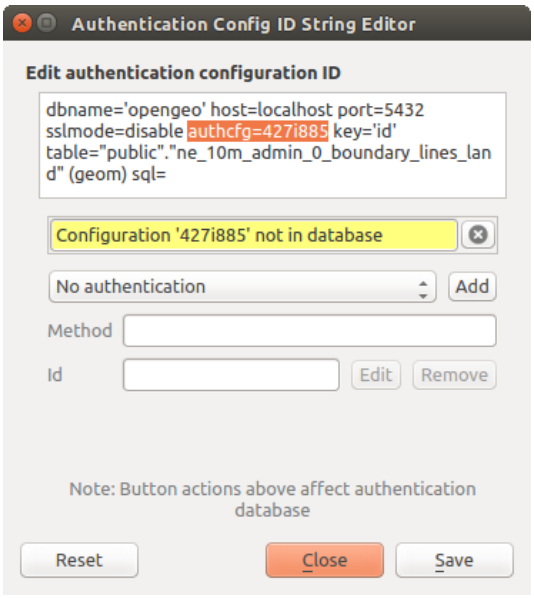


Abb. 21.31: Defekte Layer mit Authentifikationskonfiguration bearbeiten

21.2.5 Kennung der Authentifikationskonfiguration ändern

Gelegentlich müssen Sie die Authentifizierungskonfigurations-ID ändern, die mit dem Zugriff auf eine Ressource verknüpft ist. Es gibt Fälle, in denen dies nützlich ist:

- **Konfigurations-ID der Ressourcenauthentifizierung ist nicht mehr gültig**: This can occur when you have switched auth databases add need to *align* a new configuration to the ID already associated with a resource. Die : Dies kann auftreten, wenn Sie Auth-Datenbanken ausgetauscht haben, sodass eine neue Konfiguration an eine ID *angepasst* werden muss, die bereits mit einer Ressource verknüpft ist.
- **Freigegebene Projektdateien**: Wenn Sie Projekte zwischen Benutzern freigeben wollen, z.B. über einen gemeinsam genutzten Dateiserver, können Sie eine 7-stelliges ID (mit **a-z** und/oder **0-9**) *vordefinieren*, die mit der Ressource verknüpft ist. Einzelne Benutzer ändern dann die ID der Authentifizierungskonfiguration gemäß ihrer Anmeldeinformationen für die Ressource. Beim Öffnen des Projekts wird die ID in der Authentifizierungsdatenbank gefunden, aber die Anmeldeinformationen sind je nach Benutzer unterschiedlich.

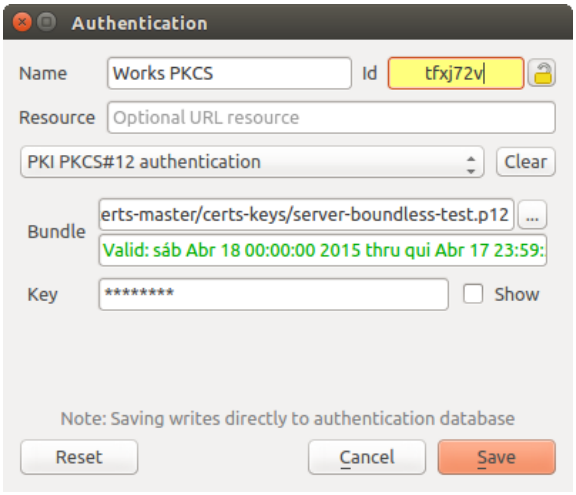


Abb. 21.32: ID einer Layer-Authentifikationskonfiguration ändern (freigeschaltetes gelbes Textfeld)

Warnung

Das Ändern der Auth-Config-ID ist eine fortgeschrittene Operation und sollte nur in Betracht gezogen werden, wenn es völlig bekannt ist, ob dies notwendig ist. Deswegen gibt es eine Sperr-Taste, die geklickt werden muss, um das ID-Text-Feld vor der Bearbeitung zu entsperren.

21.2.6 QGIS-Server-Unterstützung:

Wenn eine Projektdatei mit Layern genutzt wird, die eine Authentifizierung benötigen und die als Hintergrundkarte vom QGIS-Server verwendet werden soll, gibt es in QGIS ein paar zusätzliche Einrichtungs-Schritte zu beachten, um die Ressource zu laden:

- Authentifizierungsdatenbank muss verfügbar sein
- Das Authentifizierungsdatenbank-Master-Passwort muss verfügbar sein

Bei der Instanzierung des Authentifizierungssystems erstellt oder verwendet der Server eine `qgis-auth.db`-Datei im aktiven Benutzerprofil <user_profiles>, oder das Verzeichnis, das durch die ``QGIS_AUTH_DB_DIR_PATH`-Umgebungsvariable definiert ist. Es kann sein, dass der Benutzer des Servers kein HOME-Verzeichnis hat, in diesem Fall verwenden Sie die Umgebungsvariable, um ein Verzeichnis zu definieren, auf das der Benutzer des Servers Lese-/Schreibberechtigungen besitzt und das sich nicht innerhalb der über das Internet zugänglichen Verzeichnisse befindet.

Um das Master-Passwort an den Server zu übergeben, schreiben sie dieses in die erste Zeile einer Datei im Pfad des Dateisystems, die durch den Server-Prozess-Benutzer lesbar ist und durch die `QGIS_AUTH_PASSWORD_FILE` Umgebungsvariable definiert ist. Stellen Sie sicher, dass die Datei als *nur lesbar* für den Server-Prozess-Benutzer gekennzeichnet ist und nicht in einem über das Web zugänglichen Verzeichnis liegt.

Bemerkung

`QGIS_AUTH_PASSWORD_FILE` variable will be removed from the Server environment immediately after accessing.

21.2.7 SSL-Serverausnahmen

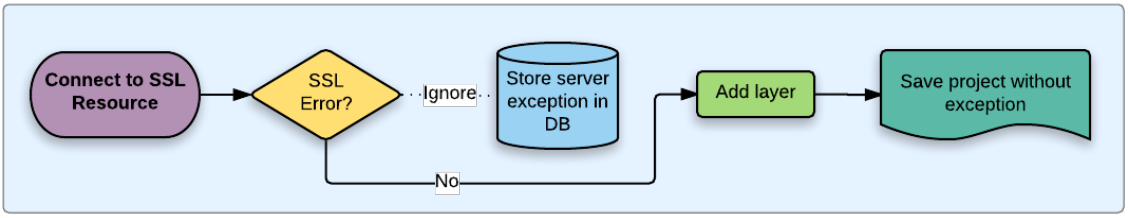


Abb. 21.33: SSL-Serverausnahme

Sie können SSL-Server-Konfigurationen und -Ausnahmen über den **Server**-Tab im Abschnitt **Authentifizierung** unter **Optionen** verwalten.

Manchmal, wenn sie sich mit einem SSL-Server verbinden, gibt es Fehler beim SSL-„Handshake“ oder mit dem Zertifikat des Servers. Sie können diese Fehler ignorieren oder eine SSL-Server-Konfiguration als Ausnahme erstellen. Dies ist ähnlich wie bei Web-Browsern, wo Sie SSL-Fehler außer Kraft setzen können, aber mit einer besseren Kontrolle.

Warnung

Sie sollten keine SSL-Server-Konfiguration erstellen, wenn Sie keine umfassenden Kenntnisse des gesamten SSL-Setups zwischen dem Server und dem Client besitzen. Stattdessen wenden Sie sich besser an den Server-Administrator.

Bemerkung

Einige PKI-Setups verwenden eine völlig andere ZA-Vertrauenskette, um Client-Identitäten zu validieren als die Kette bei der Validierung eines SSL-Server-Zertifikats. Unter solchen Umständen behebt die Erstellung einer neuen Konfiguration für den Verbindungsserver nicht unbedingt Ihr Client-Identitäts-Problem. In diesem Fall kann nur der Identitäts-Verwalter oder Serveradministrator das Problem lösen.

Sie können eine SSL-Serverkonfiguration vorkonfigurieren, indem Sie auf die -Schaltfläche klicken. Alternativ können Sie eine Konfiguration hinzufügen, wenn ein SSL-Fehler während einer Verbindung auftritt und **SSL-Fehler**-Dialogfeld angezeigt wird (wobei der Fehler vorübergehend ignoriert oder in der Datenbank gespeichert und ignoriert werden kann):

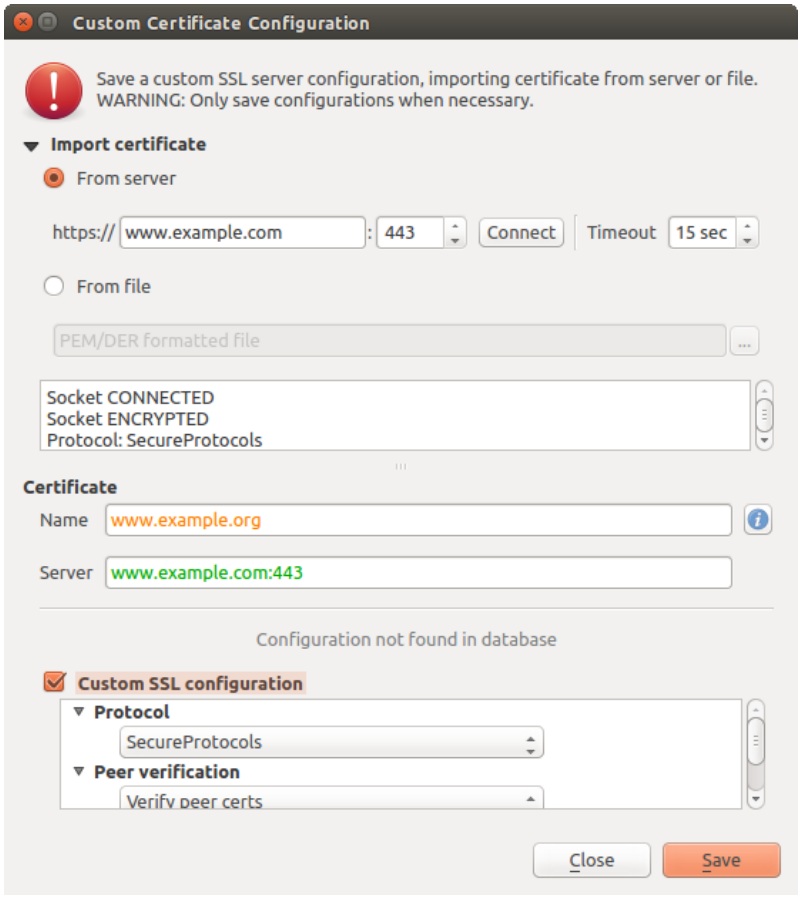


Abb. 21.34: Manuell eine Konfiguration hinzufügen

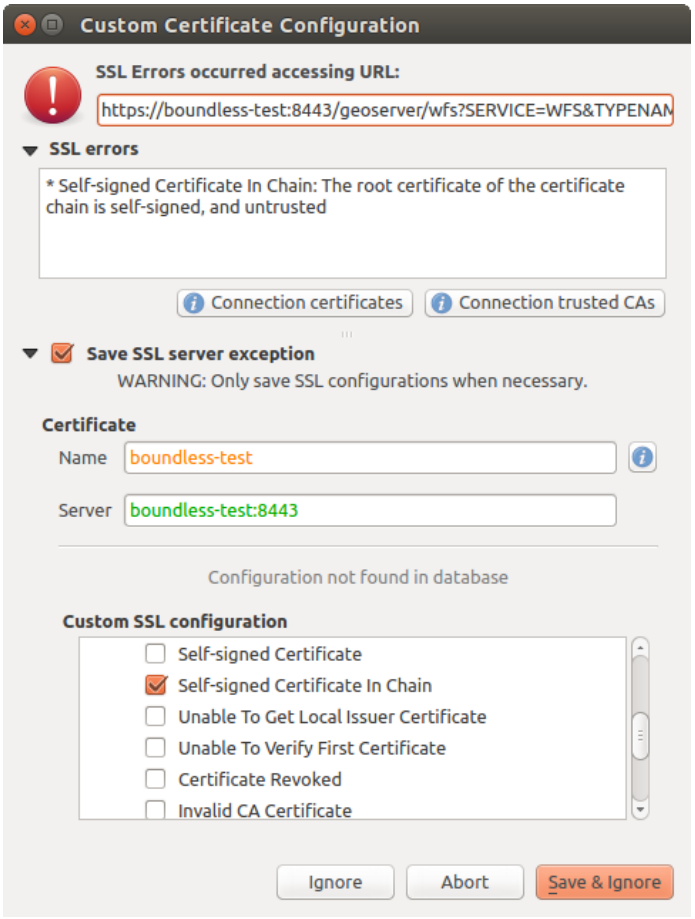


Abb. 21.35: Konfiguration während eines SSL-Fehlers hinzufügen

Sobald eine SSL-Konfiguration in der Datenbank gespeichert wird, kann sie bearbeitet oder gelöscht werden.

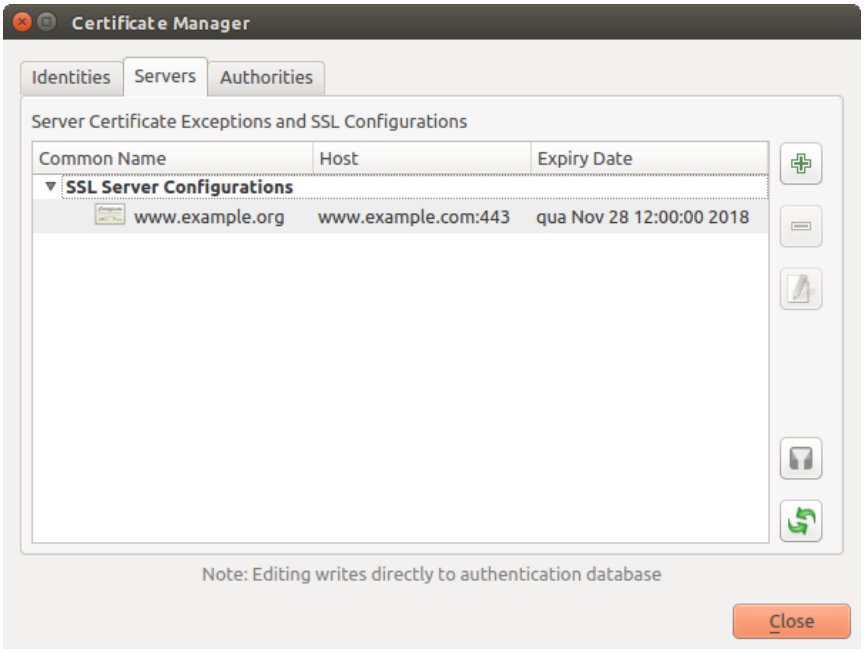


Abb. 21.36: Bestehende SSL-Konfiguration

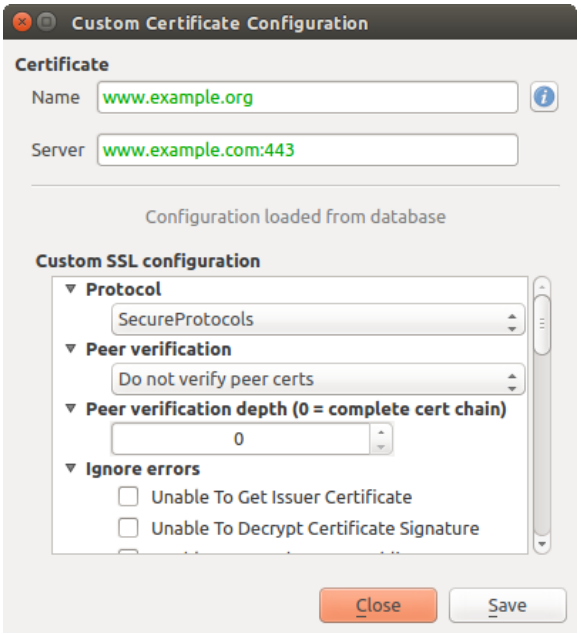


Abb. 21.37: Bestehende SSL-Konfiguration bearbeiten

Wenn Sie eine SSL-Konfiguration vorkonfigurieren möchten und der Importdialog für Ihre Server-Verbindung nicht angezeigt wird, können Sie manuell eine Verbindung über die **Python-Konsole** aufrufen, indem Sie den folgenden Code ausführen (ersetzen Sie ``https://bugreports.qt-project.org` durch die URL Ihres Servers):

```
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)
```

Dies wird ein SSL-Fehler-Dialogfeld öffnen, wenn Fehler auftreten und wo Sie die Konfiguration in die Datenbank speichern können.

KAPITEL 22

GRASS GIS Integration

Die GRASS-Integration stellt den Zugang zu GRASS GIS Datenbanken und Funktionalitäten (siehe GRASS-PROJECT in Literatur und Internetreferenzen) bereit. Die Integration besteht aus zwei Teilen: Anbieter und Plugin. Der Anbieter erlaubt das Browsen, Managen und Visualisieren von GRASS Raster- und Vektorlayern. Das Plugin kann verwendet werden, um neue GRASS Locations und Mapsets zu erstellen, die GRASS Region zu ändern, Vektorlayer zu erstellen oder zu verändern und GRASS 2-D und 3-D Daten mit Hilfe von mehr als 400 GRASS Modulen zu analysieren. In diesem Abschnitt führen wir in die Funktionalitäten des Anbieters und des Plugins ein und geben einige Beispiele zur Handhabung und Arbeit mit GRASS-Daten.

Der Anbieter unterstützt GRASS Version 6 und 7, das Plugin unterstützt GRASS 6 und 7 (ab QGIS 2.12). QGIS Distribution kann entweder Provider/Plugin für GRASS 6 oder GRAS 7 oder für beide Versionen gleichzeitig enthalten (Binärdateien haben unterschiedliche Dateinamen). Nur eine Version des Anbieters/Plugin kann zur Laufzeit jedoch geladen werden.

22.1 Beispieldatensatz

Als Beispiel verwenden wir den QGIS-Alaska-Datensatz (siehe Abschnitt *Beispieldaten herunterladen*). Er enthält eine kleine GRASS-Beispieldatei `LOCATION` mit drei Vektorebenen und einer Raster-Höhenkarte. Erstellen Sie einen neuen Ordner namens `grassdata`, laden Sie den QGIS-Datensatz „Alaska" `qgis_sample_data.zip` von https://qgis.org/downloads-list/#data herunter und entpacken Sie die Datei in `grassdata`.

Weitere Beispiel-GRASS-Dateien (`LOCATION`) finden Sie auf der GRASS Website.

22.2 GRASS Layer visualisieren

Wenn der Anbieter in QGIS geladen ist, wird für jeden Ordner der eine GRASS Location enthält das GRASS Icon im QGIS Browser angezeigt. Gehen Sie im QGIS Browser zum Ordner `grassdata` und erweitern den Ordner `alaska` und das Mapset `demo`.

Sie können GRASS Raster- und Vektorebenen wie jeden anderen Layer aus dem Browser entweder durch Doppelklick auf Layer oder per Drag & Drop auf die Karte oder Legende laden.

Tipp

Probleme beim Laden von GRASS-Layern

Wenn Sie keine GRASS Locationitems sehen, überprüfen Sie *Hilfe* ->**:Menüauswahl:`Über`** *Anbieter*, ob GRASS Vektor-Anbieter geladen sind.

22.3 Daten in eine GRASS LOCATION importieren via drag and drop

In diesem Abschnitt wird Ihnen Beispielhaft gezeigt, wie Sie Raster- oder Vektordateien in eine GRASS Mapset laden.

1. Navigieren Sie im QGIS-Browser zu dem Mapset, aus dem Sie Dateninformationen importieren wollen.
2. Um QGIS Browser einen Layer zu finden, den Sie GRASS importieren möchten finden, beachten Sie, dass Sie eine andere Instanz des Browsers öffnen können (*Browser Panel (2)*), wenn Quelldaten zu weit von der mapset im Baum entfernt sind.
3. Ziehen Sie einen Layer in die Zielkarte. Der Import nimmt unter Umständen für größere Layer etwas Zeit in Anspruch. Man sieht dann ein animiertes Icon vor dem neuen Layer bis der Import abgeschlossen ist.

Wenn Rasterdaten in einem abweichenden KBS vorliegen, können sie mit Hilfe von *Approximate* (schnell) oder *Exact* (genau) transformiert werden. Wenn eine Verknüpfung zur Quelle des Rasters erstellt ist (mit Hilfe von `r.external`), haben die Quelldaten dasselbe KBS und das Format ist GDAL bekannt. Das KBS der Quelldaten wird verwendet. Man kann diese Optionen im *Browser* Reiter unter *GRASS Optionen* einstellen.

Wenn ein Quellraster mehr Kanäle hat, wird eine neue GRASS Karte für jede Schicht mit dem **<Kanalnummerr>** Suffix und eine Gruppe aller Karten mit Symbol erzeugt. Externe Raster haben ein anderes Symbol .

22.4 GRASS Daten im QGIS Browser verwalten

- Karten kopieren: GRASS Karten können zwischen Mapsets innerhalb der gleichen Stelle mit Drag & Drop kopiert werden.
- Löschen von Karten: Rechtsklick auf eine GRASS Karte und wählen Sie **:gui-Label:`Löschen`** aus dem Kontextmenü.
- Umbenennen von Karten: Rechtsklick auf eine GRASS Karte und wählen Sie: gui-Label: *Umbenennen* aus dem Kontextmenü.

22.5 GRASS Optionen

GRASS Optionen werden im *GRASS Optionen* Dialog eingestellt, welchen Sie auch durch Rechtsklick auf die Location oder das Mapset Element im Browser öffnen können, wenn Sie *GRASS Optionen* wählen.

22.6 GRASS Plugin starten

Um die GRASS Funktionalitäten in QGIS zu nutzen, muss das GRASS Plugin in den Erweiterungen ausgewählt und geladen werden. Man geht dazu im Menü zu *Erweiterungen* ► *Erweiterungen verwalten und installieren…*, wählt dort *GRASS* und klickt auf *OK*.

Die folgenden Haupteigenschaften werden beim Starten des GRASS Plugins im GRASS Menü bereitgestellt (*Erweiterungen* ► *GRASS*):

- Mapset öffnen
- Neues Mapset
- Schließe Mapset
- GRASS-Werkzeugkiste öffnen
- Aktuelle GRASS-Region darstelllen
- GRASS Optionen

22.7 GRASS Mapset öffnen

Ein GRASS Mapset muss geöffnet werden, um Zugang zu den GRASS Tools im Plugin zu bekommen (die Werkzeuge sind deaktiviert, wenn kein Mapset geöffnet ist). Sie können ein Mapset aus dem Browser öffnen: Rechtsklick auf das Mapset Element und wählen Sie dann *Mapset öffnen* aus dem Kontextmenü.

22.8 Information zur GRASS-Datenbank

GRASS Daten werden in einem Ordner gespeichert, der als GISDBASE bezeichnet wird. Standardmäßig wird der Ordner `grassdata` genannt und er muss erstellt worden sein, bevor man beginnt, mit dem GRASS Plugin in QGIS zu arbeiten. Innerhalb dieses Ordners sind die GRASS Daten als Projekte in Unterordnern genannt `LOCATIONs` organisiert. Jede `LOCATION` ist durch ein Koordinatenbezugssystem, Kartenprojektion und eine geographische Grenze definiert und kann darüberhinaus weitere Unterordner `MAPSETs` (Unterordner des Ordners `LOCATION`) besitzen, um die Layer der `LOCATION` weiter z.B. thematisch oder räumlich zu unterteilen (siehe Neteler & Mitasova 2008 in *Literatur und Internetreferenzen*). Um Raster- und Vektorlayer mit den GRASS Modulen zu analysieren, müssen diese zuerst in eine passende GRASS `LOCATION` importiert werden. (Dies ist nicht ganz korrekt. Mit den GRASS Modulen `r.external` und `v.external` können Sie eine ‚read-only' Verknüpfung zu externen durch GDAL/OGR-unterstützte Layer erstellen, ohne die Daten importieren zu müssen. Da dies aber nicht der normale Weg für GRASS Anfänger ist, wird auf diese Möglichkeit nicht näher eingegangen.).

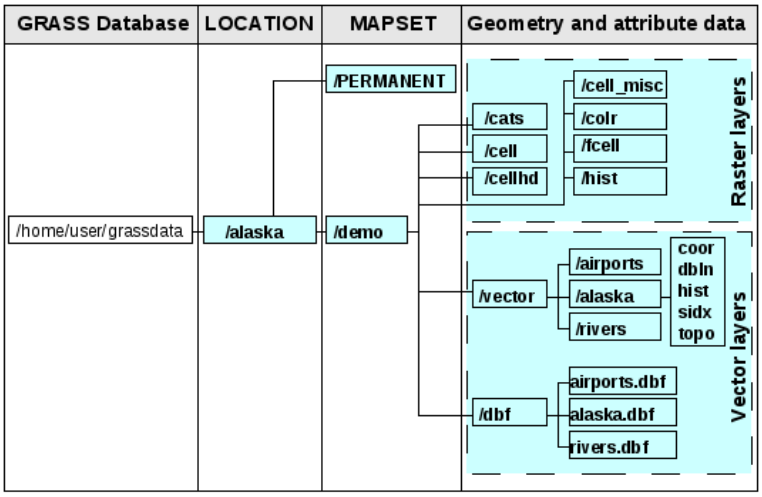


Abb. 22.1: GRASS Daten in der Alaska LOCATION

22.9 Daten in eine GRASS LOCATION importieren

Siehe Abschnitt import_data dnd um herauszufinden, wie die Daten einfach importiert per Drag & Drop in den Browser importert werden können.

Dieser Abschnitt zeigt ein Beispiel wie man Raster- und Vektordaten in die ‚alaska' GRASS `LOCATION` aus dem QGIS ‚Alaska' Datensatz in einem üblichen Weg, mit Hilfe von Standard GRASS Modulen, importiert. Daher verwenden wir die Landcover Rasterkarte `landcover.img` und die GML Vektorkarte `lakes.gml` aus dem QGIS ‚Alaska' Datensatz (siehe *Beispieldaten herunterladen*).

1. Starten Sie QGIS und laden Sie das GRASS Plugin, falls dies noch nicht geschehen ist.
2. Klicken Sie in der GRASS Werkzeugleiste das Mapset öffnen Icon um den *Wählen Sie ein GRASS Mapset* Assistenten zu starten.
3. Wählen Sie als GRASS Datenbank den Ordner `grassdata` im QGIS Alaska Datensatz, als `LOCATION` ‚alaska', als `MAPSET` ‚demo' und klicken auf *OK*.
4. Nun klicken Sie auf das Icon GRASS-Werkzeugkiste öffnen, damit die GRASS Werkzeuge (siehe Abschnitt *Die GRASS Werkzeugkiste*) zur Verfügung stehen.
5. Um die Rasterkarte `landcover.img` zu importieren, drücken Sie auf das Modul `r.in.gdal` im *Modulbaum* Reiter. Diese GRASS Module ermöglicht es, GDAL-unterstützte Rasterdateien in eine GRASS `LOCATION` zu importieren.
6. Browsen Sie zum Ordner `raster` im QGIS ‚Alaska' Datensatz und wählen Sie die Datei `landcover.img`.
7. Geben Sie als Ausgabe-Rasterkarte `landcover_grass` ein und klicken auf *Starten*. Im Reiter *Ergebnis* sehen Sie das aktuell ausgeführte GRASS Kommando `r.in.gdal -o input=/path/to/landcover.img output=landcover_grass`.
8. Klicken Sie nach der Ausgabe **Erfolgreich beendet** auf *Ergebnis visualisieren*. Der Rasterlayer `landcover_grass` ist nun in GRASS importiert und wird im QGIS Arbeitsbereich angezeigt.
9. Um die GML Vektordatei `lakes.gml` zu importieren klicken Sie das Modul `v.in.ogr` im *Modulbaum* Reiter. Dieses GRASS Modul ermöglicht den Import con OGR-unterstützen Vektordateien in eine GRASS `LOCATION`. Der Moduldialog für `v.in.ogr` erscheint.
10. Browsen Sie zum Ordner `gml im QGIS 'Alaska' Datensatz und wählen Sie die Datei :file:`lakes.gml` als OGR Datei aus.

11. Vergeben Sie `lakes_grass` als Ausgabe-Vektorkarte und klicken auf *Starten*. Sie brauchen sich um die anderen Optionen in diesem Beispiel nicht kümmern. Im Reiter *Ergebnis* sehen Sie das aktuell ausgeführte GRASS Kommando `v.in.ogr -o dsn=/path/to/lakes.gml output=lakes\_grass`.
12. Klicken Sie nach Erscheinen der Ausgabe **Erfolgreich beendet** auf *Ergebnis visualisieren*. Der Vektorlayer `lakes_grass` ist nun in GRASS importiert und wird im QGIS Arbeitsbereich dargestellt.

22.9.1 Eine neue GRASS LOCATION erstellen

Als ein Beispiel möchten wir Ihnen zeigen, wie die GRASS `LOCATION` des Alaska Beispieldatensatzes erstellt wurde. Das Koordinatenbezugssystem ist Albers Equal Area mit der Einheit ‚feet'. Diese GRASS :file:` LOCATION alaska` wird für alle GRASS GIS Beispiele verwendet. Es ist also sinnvoll, sich diesen Datensatz zu installieren (siehe Abschnitt *Beispieldaten herunterladen*).

1. Starten Sie QGIS und laden Sie das GRASS Plugin, falls dies noch nicht geschehen ist.
2. Laden Sie das Shapefile `alaska.shp` aus dem QGIS Alaska Datensatz (see *Beispieldaten herunterladen*) in QGIS ein (see section *Laden eines Layers aus einer Datei*).
3. Klicken Sie in der GRASS Werkzeugleiste auf das Neues Mapset Icon um den *Neues Mapset* Assistenten zu öffnen.
4. Wählen Sie einen vorhandenen GRASS Datenbankordner (GISDBASE) `grassdata` oder legen Sie für die neue `LOCATION` mit Hilfe des Dateimanagers einen neuen Ordner an. Klicken Sie dann auf *Weiter*.
5. Mit diesem Assistenten können wir einen neuen `MAPSET` innerhalb einer bestehenden `LOCATION` erstellen (siehe Abschnitt *Eine neue GRASS MAPSET erstellen*) oder eine komplett neue `LOCATION` erstellen. Wählen Sie *Create new location* (siehe Abb. 22.2).
6. Gebe einen Namen für die `LOCATION` an – wir verwendeten ‚alaska' – und klicke auf *Weiter*.
7. Definieren Sie die Projektion indem Sie den Radiobutton *Projektion* zum Aktivieren der Projektionsliste klicken.
8. Wir benutzen Albers Equal Area Alaska (Fuß) Projektion. Da wir zufällig wissen dass es durch den EPSG ID 2964 repräsentiert wird geben wir dies in das Suchfenster ein. (Bemerkung: Wenn Sie diesen Ablauf für eine andere `LOCATION` und Projektion wiederholen wollen und Sie die EPSG ID nicht gespeichert haben, klicken Sie auf das KBS-Status Icon in der unteren rechten Ecke der Statusleiste (siehe Abschnitt *Arbeiten mit Projektionen*)).
9. Geben Sie bei *Filter* 2964 ein um die Projektion auszuwählen.
10. Klicken Sie auf *Weiter*.
11. Um die voreingestellte Region zu bestimmen, muss man die Grenzen der `LOCATION` im Norden, Süden, Osten und Westen angeben. Wir klicken einfach auf den Knopf *Setze aktuelle QGIS-Ausdehnung*, um die Ausdehnung des in QGIS geladenen Layers `alaska.shp` als voreingestellte GRASS-Region zu verwenden.
12. Klicken Sie auf *Weiter*.
13. Wir müssen außerdem ein `MAPSET` innerhalb unserer neuen `LOCATION` vorgeben (das ist beim Erstellen einer neuen `LOCATION` erforderlich). Sie können hier einen beliebigen Namen vergeben, wir verwendeten ‚demo'. GRASS erstellt automatisch ein spezielles `MAPSET` mit Namen `PERMANENT`, das die Hauptdaten des Projekts, die vorgegebene Ausdehnung und die Vorgaben zum Koordinatensystem speichert (siehe auch Neteler & Mitasova 2008 in *Literatur und Internetreferenzen*).
14. Sehen Sie sich die Zusammenfassung an und vergewissern sich, dass alles korrekt ist und klicken anschließend auf *Abschließen*.
15. Die neue `LOCATION`, ‚alaska', und zwei `MAPSETs`, ‚demo' und ‚PERMANENT' werden erstellt. Die gerade geöffnete Arbeitsumgebung ist ‚demo', wie Sie es definiert haben.
16. Beachten Sie, dass einige Werkzeuge des GRASS Plugins grau hinterlegt waren und nun auch zur Verfügung stehen.

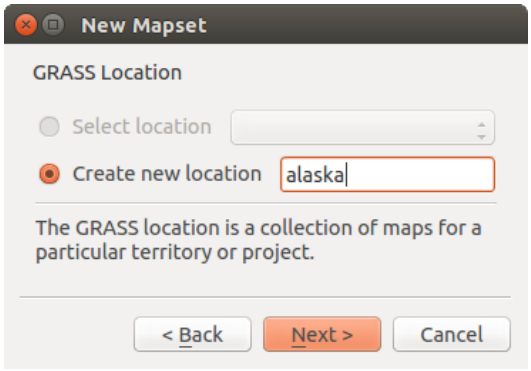


Abb. 22.2: Erstellen einer neuen GRASS LOCATION oder einer neuen MAPSET in QGIS

Wenn das wie eine Menge von Schritten schien ist es doch nicht so schlimm und ein schneller Weg eine LOCATION zu erstellen. Die LOCATION ‚alaska' is jetzt bereit für den Datenimport (siehe Abschnitt *Daten in eine GRASS LOCATION importieren*). Sie können auch die bereits bestehenden Vektor- und Rasterdaten in der Beispiel GRASS LOCATION ‚alaska' verwenden, die im |qgi| ‚Alaska' Datensatz *Beispieldaten herunterladen* enthalten ist und weiter zu Abschnitt *Das GRASS Vektormodell* gehen.

22.9.2 Eine neue GRASS MAPSET erstellen

Ein Benutzer hat nur Schreibzugriff auf eine MAPSET die er oder sie erstellt hat. Das heißt dass Sie neben dem Zugriff zu Ihrer eigenen MAPSET Sie Karten in MAPSETs anderer Benutzer lesen können (und Sie können Ihre lesen) aber Sie nur die Karten in Ihrer eigenen MAPSET verändern und löschen können.

Alle MAPSETs beinhalten eine WIND Datei die die aktuellen Grenzkoordinatenwerte und die aktuell ausgewählte Rasterauflösung speichert (siehe Neteler & Mitasova 2008 in *Literatur und Internetreferenzen*, und Abschnitt *Einstellung der GRASS Region*).

1. Starten Sie QGIS und laden Sie das GRASS Plugin, falls dies noch nicht geschehen ist.
2. Klicken Sie in der GRASS Werkzeugleiste auf das Neues Mapset Icon um den *Neues Mapset* Assistenten zu öffnen.
3. Wählen Sie den GRASS Datenbankordner (GISDBASE) grassdata mit der LOCATION ‚alaska' aus wo Sie eine weitere MAPSET genannt ‚test' hinzufügen wollen.
4. Klicken Sie auf *Weiter*.
5. Mit diesem Assistenten können wir ein neues MAPSET innerhalb eines bestehenden LOCATION erstellen oder einen komplett neuen LOCATION erstellen. Klicken Sie auf das Optionsfeld *Select location* (siehe Abb. 22.2) und klicken Sie auf *Next*.
6. Geben Sie den Namen test ein für ein neues MAPSET. Unten im Wizard sehen Sie eine Liste aller existierenden MAPSETs und ihrer Eigentümer.
7. Klicken Sie auf *Weiter*, vergewissern Sie sich, dass alle Angaben richtig sind und klicken auf *Abschließen*.

22.10 Das GRASS Vektormodell

Es ist wichtig, das :index:`GRASS Vektordaten Modell ‚vor der Digitalisierung zu verstehen. Im Allgemeinen verwendet GRAS ein topologisches Vektormodell. Das bedeutet, dass Flächen nicht als geschlossene Polygone vorhanden sind, sondern als ein oder mehrere Umrandungen (Boundaries). Eine Umrandung (Boundary) zwischen zwei aneinander grenzenden Flächen ist nur einmal digitalisiert worden; beide Flächen teilen sich diese Umrandung. Umrandungen dürfen keine Lücken haben. Eine Fläche besteht also aus einer Umrandung und einem Zentroid, der diese Fläche als ein sog. **Labelpunkt** mit einer Attributtabelle verknüpft.

Neben den Umrandungen und Zentroiden kann eine Vektorkarte selbstverständlich auch Punkte und Linien enthalten. Alle diese Geometrieelemente können innerhalb ein und dem selben Datensatz enthalten sein. Sie werden in unterschiedlichen ‚Ebenen' innerhalb von QGIS dargestellt. Auch wenn es möglich ist, Geometrieelemente zu mischen, so ist es eigentlich unüblich und wird normalerweise auch in GRASS GIS nur selten verwendet. Etwa bei Netzwerkanalysen. Im Normalfall sollten Sie versuchen, unterschiedliche Geometrietypen in unterschiedlichen Datensätzen (Layern) zu speichern.

Es ist auch möglich, unterschiedliche Inhalte des gleichen Geometrietyps in verschiendenen Ebenen eines Vektorlayers zu speichern. Beispielsweise können Felder, Wälder und Seen in einem Vektordatensatz gespeichert werden. Angrenzende Seen, Felder und Wälder teilen sich dann die gleiche Umrandung, jedoch haben sie separate Attributtabellen, die über ihre Ebene angesprochen wird. Darüber hinaus können Sie auch Attribute für die Umrandungen vergeben, falls eine Umrandung gleichzeitg einen Weg darstellt. In diesem Fall könnte auch die Umrandung eine separate Attributtabelle haben.

Die ‚Ebene' eines jeden Objektes wird in GRASS intern als ‚layer' bezeichnet. ‚Layer' ist die Nummer die definiert ob es mehr als einen Layer innerhalb des Datensatzes gibt (z.B. ob die Geometrie Wald oder See ist). Vorerst kann dies nur eine Nummer sein. In Zukunft wird GRASS auch Namen als Felder in der Benutzeroberfläche unterstützen.

Attribute können innerhalb der GRASS `LOCATION` als dBase oder SQLite3 oder in externen Datenbanktabellen abgelegt werden, z.B. PostgreSQL, MySQL, SQLITE3, etc.

Die Attribute in den Tabellen werden über ein sog. ‚Kategoriefeld' an die Geometrien des Datensatzes gehängt.

Die ‚Kategorie' (oder key, ID, etc) ist eine Ganzzahl, über die eine Verknüpfung zwischen den Geometrien und den Spalten in der Datenbanktabelle hergestellt wird.

> **Tipp**
>
> **Das GRASS Vektormodell verstehen**
>
> Der beste Weg, um das GRASS-Vektormodell und seine Funktionen kennenzulernen, ist das Herunterladen eines der zahlreichen GRASS-Tutorials, in denen das Vektormodell ausführlich beschrieben wird. Weitere Informationen, Bücher und Tutorials in mehreren Sprachen finden Sie unter https://grass.osgeo.org/learn/manuals/.

22.11 Einen neuen GRASS Vektorlayer erstellen

Um einen neuen GRASS Vektorlayer zu erzeugen, wählen Sie eines der folgenden Elemente aus dem Mapset Kontextmenü im Browser:

- Neuer Punktlayer
- Neuer Linienlayer
- Neuer Polygonlayer

und einen Namen in das Dialogfeld eingeben. Eine neue Vektorkarte wird erstellt und der Layer dem Kartenfenster hinzugefügt und die Bearbeitung gestartet. Einen Typ des Layers zu wählen beschränkt nicht den Geometrietypen, der in der Vektorkarte digitalisiert werden kann. In GRASS ist es möglich, alle Arten von Geometrietypen (Punkt, Linie und Polygon) in einer Vektorkarte zu organisieren. Der Typ wird nur verwendet, um die Layer dem Kartenfenster hinzuzufügen, da QGIS von dem Layer fordert eine bestimmte Art zu haben.

Es ist auch möglich, Layer bestehenden Vektorkarten hinzuzufügen, wählen Sie ein Element oben aus dem beschriebenen Kontextmenü der bestehenden Vektorkarte.

GRASS GIS erlaubt es aufgrund des topologischen Datenmodells, die verschiedenen Geometrietypen (Punkt, Linie und Fläche) in einem Vektorlayer abzuspeichern. Aus diesem Grund ist es nicht notwendig im Vorfeld einen Geometrietyp festzulegen. Dies unterscheidet sich von der Erstellung eines neuen Shapefile in QGIS, denn Shapefiles verwenden das Simple Feature Vektormodell (siehe Abschnitt *Neue Vektorlayer erstellen*).

22.12 Digitalisieren und Editieren eines GRASS Vektorlayers

GRASS Vektorebenen können mit den Standard QGIS Digitalisierfunktionen werden digitalisiert. Es gibt jedoch einige Besonderheiten, die Sie kennen sollten, aufgrund

- GRASS topologischen Modell im Vergleich zu einfachen QGIS Funktion
- Komplexität von GRASS-Modellen
  - mehrere Layer in einer Karte
  - mehrere Geometrietypen in einer Karte
  - Geometrieteilung von mehreren Objekte durch mehrere Layer

Die Besonderheiten sind in den folgenden Abschnitten besprochen.

Speichern, Änderungen verwerfen, rückgängig machen, wiederholen

> **Warnung**
>
> Alle während der Bearbeitung vorgenommenen Änderungen werden sofort auf Vektorkarten und die dazugehörigen Attributtabellen geschrieben.

Die Änderungen werden nach jeder Operation geschrieben, es ist jedoch möglich, Rückgängig/Wiederholen zu wählen oder alle Änderungen zu verwerfen, wenn die Bearbeitung geschlossen wird. Wenn Änderungen rückgängig gemacht werden oder Änderungen verworfen werden, wird der ursprüngliche Zustand in der Vektorkarte neu geschrieben und die der Attributtabellen.

Es gibt zwei Hauptgründe für dieses Verhalten:

- Es ist die Natur von GRASS-Vektoren aus Überzeugung zu kommen, dass der Benutzer tun will, was er tut und es ist besser, dass die Daten gespeichert werden, wenn die Arbeit plötzlich unterbrochen wird (z. B. Stromausfall)
- Die Notwendigkeit für eine effektive Bearbeitung von topologischen Daten visualisiert Informationen über die topologische Korrektheit, solche Informationen können nur von GRASS Vektorkarte erworben werden, wenn Änderungen an der Karte vorgenommen werden.

Werkzeugleiste

Die ‚Digitalisierungsleiste' hat einige spezielle Werkzeuge, wenn Sie einen GRASS Layer bearbeiten:

Tab. 22.1: GRASS Digitizing Tools

Icon	Werkzeug	Zweck
	Neuer Punkt	Digitalisiert neuen Punkt
	Neue Linie	Digitalisiert neue Linie
	Neue Grenze	Digitalisiert neue Grenze
	Neuer Zentroid	Digitalisiert neuen Zentroiden (Labelpunkt für eine existierende Fläche)
	Neue geschlossene Grenze	Digitalisiert neue geschlossene Grenze

Tipp

Polygone in GRASS digitalisieren

Wenn Sie ein Polygon innerhalb eines GRASS Vektorlayers erstellen wollen, digitalisieren Sie zuerst die Boundary der Fläche. Danach fügen Sie einen Zentroid (Labelpunkt) hinzu. Der Grund ist, dass in topologischen Layern die Attributinformationen einer Fläche immer mit dem Zentroiden und nicht mit der Grenze verknüpft werden.

Kategorie

Kategorie, die oft als cat bezeichnet wird, ist eine Art von ID. Der Name stammt aus Zeiten, in denen GRASS-Vektoren nur einzelne Attribut „Kategorien“ hatten. Kategorien werden als Bindeglied zwischen Geometrie und Attribute verwendet. Eine einzelne Geometrie kann mehrere Kategorien haben und somit mehrere Funktionen in verschiedenen Layern darstellen. Derzeit ist es möglich, nur eine Kategorie pro Layer mit QGIS Werkzeugen zu bearbeiten. Neue Funktionen haben automatische neue einzigartige Kategorien, ohne Grenzen. Grenzen bilden in der Regel nur Bereiche und stellen keine lineare Funktionen dar, es ist jedoch möglich, Attribute für eine Grenze später zu definieren, beispielsweise in einer anderen Ebene.

Neue Kategorien werden immer nur in derzeit bearbeiteten Layern erstellt.

Es ist nicht möglich mehrere Kategorien zuweisen, um die Geometrie mit QGIS zu bearbeiten, solchen Daten werden wie mehrere Funktionen dargestellt und einzelne Merkmale, auch aus verschiedenen Schichten, können gelöscht werden.

Attribute

Die Attribute des aktuell bearbeiteten Layer können nur geändert werden, wenn die Vektorkarte mehrere Layer enthält, Merkmale anderer Layer werden alle ‚<nicht editierbar (layer#)>‘ eingestellt, so werden Sie gewarnt, dass solche Attribut nicht editierbar sind. Der Grund dafür ist, dass andere Layer in der Regel andere Gruppe von Feldern haben und haben können, während QGIS nur einen festen Satz von Feldern pro Layer unterstützt.

Wenn einer Geometrie primitive nicht eine Kategorie zugeordnet ist, wird eine neue einzigartige Kategorie automatisch zugeordnet und eine neue Eintragung in der Attributtabelle erstellt, wenn ein Attribut dieser Geometrie verändert wird.

Tipp

Wenn Sie Bulk-Aktualisierung von Attributen in der Tabelle machen wollen, zum Beispiel mit dem ‚Feldrechner‘ (*Verwendung des Feldrechners*), und es einige Funktionen ohne Kategorie gibt, die Sie nicht aktualisieren wollen (in der Regel Grenzen) , können Sie diese in dem „Erweiterten Filter‘ auf `cat is not null` einstellen.

Stil editieren

Die topologische Symbologie ist für eine effektive Bearbeitung von topologischen Daten wesentlich. Bei Beginn der Bearbeitung, eine spezialisierter Renderer ‚GRASS Bearbeiten‘ wird automatisch auf die Ebene gesetzt und der ursprüngliche Renderer wiederhergestellt, wenn die Bearbeitung geschlossen ist. Der Stil kann in den Layereigenschaften im ‚Stil‘ Reiter angepasst werden. Der Stil kann auch in der Projektdatei oder in einer separaten Datei gespeichert werden. Wenn Sie den Stil anpassen, ändern sich nicht seinen Namen, weil er zurückgesetzt wird, wenn das Bearbeiten erneut gestartet wird.

Tipp

Speichern Sie die Projektdatei nicht, wenn der Layer bearbeitet wird, dies würde den Layer mit ‚Edit Style‘ speichern, was keine Bedeutung hat, wenn der Layer nicht bearbeitet wird.

Der Stil basiert auf topologischen Informationen, die vorübergehend als Feld ‚topo_symbol‘ der Tabelle hinzugefügt wird. Das Feld wird automatisch entfernt, wenn die Bearbeitung abgeschlossen ist.

> **Tipp**
>
> Entfernen Sie nicht das ‚topo_symbol' Feld aus der Attributtabelle, das würde das Objekt unsichtbar machen, weil die Darstellung basiert auf dieser Spalte.

Einrasten

Um einen Bereich zu bilden, müssen Ecken verbundener Grenzen **genau** die gleichen Koordinaten haben. Dies kann mit dem Schnappen Werkzeug erreicht werden, wenn Kartenfenster und Vektorkarte das gleiche KBS haben. Ansonsten, koordiniert durch Umwandlung von Karte auf Kartenfenster und wieder zurück, können die Koordinaten etwas unterschiedlich wegen Darstellungsfehler und KBS Transformationen werden.

> **Tipp**
>
> Layer KBS verwenden, auch für das Fenster während dem editieren.

Einschränkungen

Die gleichzeitige Bearbeitung von mehreren Layern innerhalb des gleichen Vektors wird nicht unterstützt. Dies ist vor allem deswegen unmöglich, durch das nicht händeln mehrere Handhabungs Stacks für eine einzelne Datenquelle.

Unter Linux und macOS kann jeweils nur ein GRASS layer bearbeitet werden. Das liegt an einem Fehler in GRASS der es nicht erlaubt, Datenbanktreiber in zufälliger Reihenfolge zu schließen. Der Fehler wird von den GRASS Entwicklern behoben.

> **Tipp**
>
> **GRASS Schreibberechtigung**
>
> Sie müssen der Besitzer der GRASS `MAPSET`, die Sie bearbeiten wollen, sein. Es ist unmöglich Datenlayer in einer `MAPSET` die Ihnen nicht gehört zu bearbeiten, auch wenn Sie Schreibrechte darauf haben.

22.13 Einstellung der GRASS Region

Die Region Einstellung (Einstellung eines räumlichen Arbeitsfensters) in GRASS ist wichtig für das Arbeiten mit Rasterlayern. Vektoranalysen sind standardmäßig nicht auf eine definierte Region Definition begrenzt. Aber alle neu erstellten Raster werden die räumliche Ausdehnung und Auflösung der gerade definierten GRASS Region haben, unabhängig von Ihrer ursprünglichen Ausdehnung und Auflösung. Die aktuelle GRASS Region ist in der `$LOCATION/$MAPSET/WIND` Datei gespeichert und Sie definiert die Nord-, Süd-. Ost- und Westgrenze, die Anzahl von Spalten und Reihen sowie die horizontale und vertikale räumliche Auflösung.

Es ist auch möglich, die Region mit dem Knopf Aktuelle GRASS-Region darstellen aus- bzw. einzuschalten.

Die Region kann im Reiter ‚Region' im ‚GRASS-Werkzeuge' Widget angepasst werden. Man kann die neuen Grenzen und die neue Auflösung hier eingeben und auf *Anwenden* drücken. Durch Drücken auf den Knopf *Ausdehnung durch Ziehen auf der Karte wählen* kann die Region interaktiv durch Ziehen eines Rechteckes mit der Maus im QGIS Arbeitsbereich festgelegt werden.

Das GRASS Modul `g.region` bietet viele weitere Optionen zur Einstellung der passenden Ausdehnung und Auflösung der Region für die Rasteranalyse. Das Modul `g.region` können Sie über die GRASS Werkzeugkiste nutzen, wie in Kapitel *Die GRASS Werkzeugkiste* beschrieben.

22.14 Die GRASS Werkzeugkiste

Die GRASS-Werkzeugkiste ermöglicht es, GRASS Module auf Daten innerhalb einer ausgewählten GRASS `LOCATION` und `MAPSET` anzuwenden. Dazu muss im Vorfeld eine GRASS `LOCATION` und `MAPSET` geöffnet werden, in der Sie Schreibrechte besitzen. Dies ist normalerweise garantiert, wenn Sie die `MAPSET` selbst erstellt haben und notwendig, damit die Ergebniskarten der Raster- und Vektoranalysen in der ausgewählten `LOCATION` und :file:`MAPSET` gespeichert werden können.

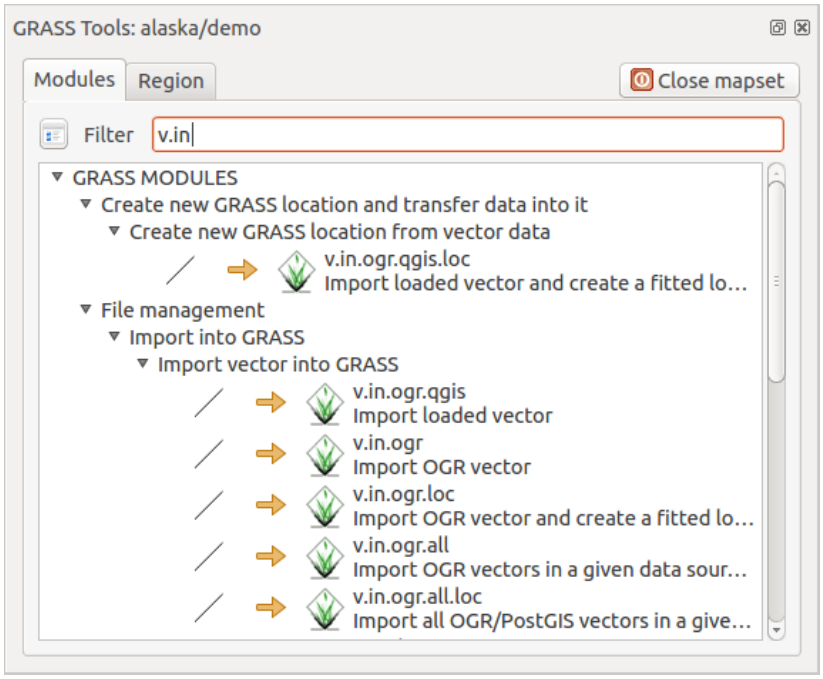


Abb. 22.3: GRASS Werkzeuge und Modulbaum

22.14.1 Arbeiten mit GRASS Modulen

The GRASS plugin provides a user-friendly toolbox with about 200 of the available GRASS modules and functionalities.

Eine vollständige Liste der GRASS-Module, die in der grafischen Toolbox in der QGIS-Version | CURRENT | verfügbar sind ist im GRASS-Wiki unter https://grasswiki.osgeo.org/wiki/GRASS-QGIS_relevant_module_list verfügbar.

Es ist außerdem möglich, die GRASS Werkzeugkiste anzupassen und weitere Module zu integrieren. Die Herangehensweise ist in Abschnitt *Anpassen der Module* beschrieben.

Wie in Abb. 22.3 gezeigt, können Sie das passende GRASS-Modul über die thematisch gruppierte *Modules Tree* oder die durchsuchbare Registerkarte *Modules List* suchen.

Wenn Sie auf das grafische Icon eines Modules klicken, öffnet sich ein neuer Moduldialog mit drei Reitern *Optionen*, *Ergebnis* und *Handbuch*.

Optionen

Der Reiter *Optionen* stellt Ihnen in vereinfachter Form die unbedingt notwendigen Eingabeparameter zur Verfügung, die das Modul zum Laufen benötigt.

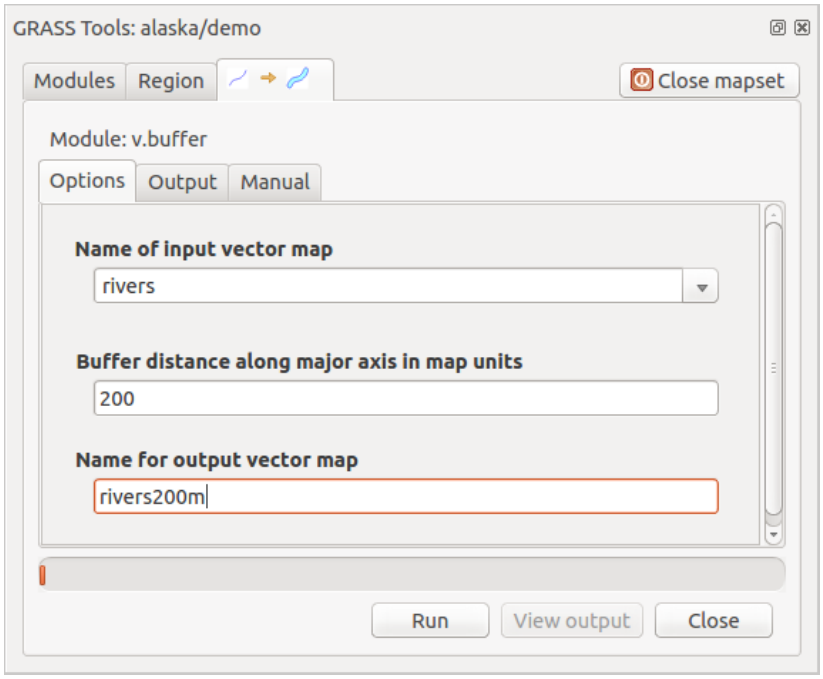


Abb. 22.4: GRASS Werkzeuge Modul Optionen

Die zur Verfügung gestellten Modulparameter sind oftmals nicht vollständig um den Dialog einfach zu halten. Wenn Sie weitergehende Modulparamter und -flags verwenden wollen müssen Sie die GRASS Shell starten und das Programm in der Kommandozeile benutzen.

Eine neue Funktion seit der QGIS Version 1.8.0 ist der Knopf *Fortgeschrittene Optionen einblenden* unterhalb der vereinfachten Moduldialoge im Reiter *Optionen*. Momentan ist diese Funktion nur für das Modul v.in.ascii umgesetzt. Es ist aber zukünftig für weitere Module vorgesehen und wird so die kompletten Funktionen der GRASS Module grafisch bereitstellen, ohne die GRASS Shell und somit die Kommandozeile benutzen zu müssen.

Ergebnis

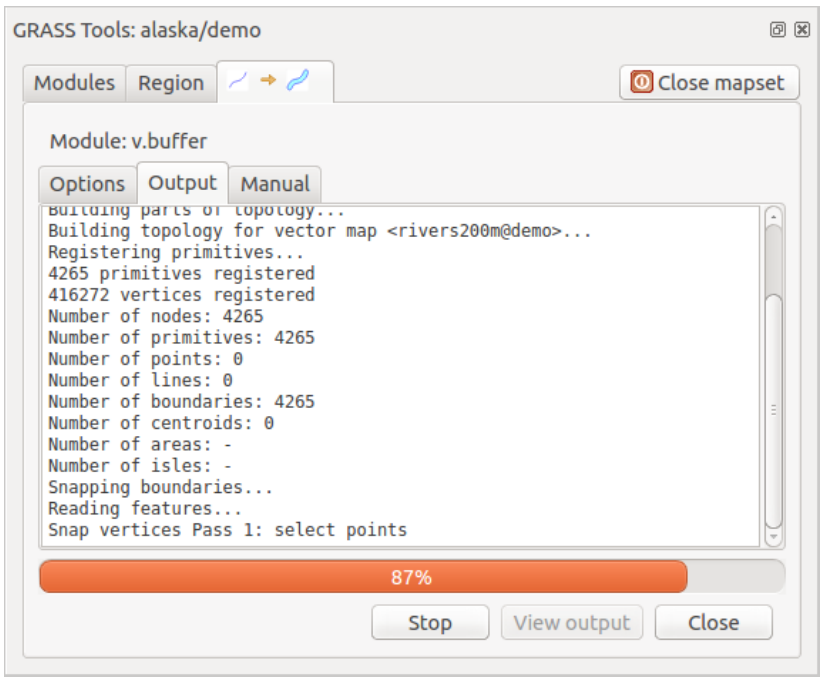


Abb. 22.5: GRASS Werkzeuge Modul Ergebnis

Der Reiter *Ergebnis* stellt Informationen über den Ausgabestatus des Moduls bereit. Nach Klick auf den *Starten* Knopf wechselt das Modul zum Reiter *Ergebnis* und zeigt Informationen über die Abarbeitung des Prozesses. Wenn alles funktioniert, erscheint schlussendlich die Meldung `Erfolgreich beendet`.

Handbuch

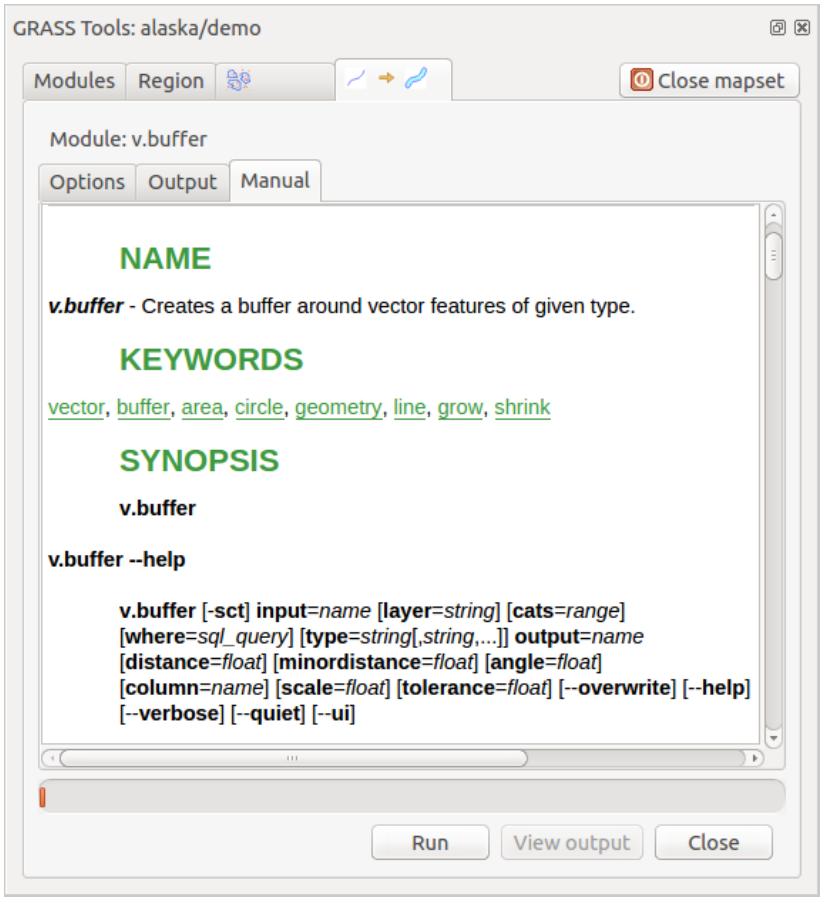


Abb. 22.6: GRASS Werkzeuge Modul Handbuch

Der Reiter *Handbuch* zeigt die HTML Hilfeseite der GRASS Module. Sie können es zum Überprüfen weitergehender Modulparameter oder -flags oder um vertieftere Kenntnisse über den Anwendungszweck der Module zu erwerben benutzen. Am Ende jeder Modul Handbuchseite sehen Sie weiterführende Links zum `Main Help index`, dem `Thematic index` und dem `Full index`. Diese Links stellen die gleiche Information wie das Modul `g.manual` zur Verfügung.

Tipp

Ergebnisse direkt anzeigen

Wollen Sie Ihre Ergenisse direkt in der Kartenansicht ansehen, nutzen Sie den Knopf ‚Ergebnis visualisieren' im unteren Bereich des jeweiligen Modulreiters

22.14.2 GRASS Beispielanwendung

Die folgenden Beispiele sollen die Anwendung verschiedener GRASS Module demonstrieren.

Höhenlinien aus einem DGM erstellen

Im ersten Beispiel wird eine Höhenlinienkarte aus einem Höhenraster (DEM) erstellt. Hier wird angenommen dass Sie die Alaska `LOCATION` wie in Abschnitt *Daten in eine GRASS LOCATION importieren* eingerichtet haben.

- Als erstes öffnen Sie die Location, indem Sie auf das Mapset öffnen Icon klicken und dann die Alaska Location auswählen.
- Als nächstes drücken Sie auf das Icon GRASS-Werkzeugkiste öffnen.
- In der Liste von Werkzeugkategorien doppelklicken Sie auf *Raster* ► *Oberflächenverwaltung* ► *Vektorkonturlinien erzeugen*.
- Jetzt wird ein einfacher Klick auf das Werkzeug **r.contour** einen Werkzeugdialog wie oben beschrieben öffnen (siehe *Arbeiten mit GRASS Modulen*).
- Geben Sie `gtopo30` in *Name des EIngaberasters* ein.
- Geben Sie im Feld *Abstand zwischen den Kontourintervallen* den Wert 100 an. (Dadurch werden Höhenlinien in einem Abstand von 100m erstellt.)
- In das Feld *Name der Vektorausgabekarte* geben Sie den Namen `hoehen_100` an.
- Klicken Sie auf *Starten* um die Verarbeitung zu starten. Warten Sie bis die Meldung `Erfolgreich beendet` im Ergebnisfenster erscheint. Drücken Sie dann auf *Ergebnis visualisieren* und *Schließen*.

Da die aktuelle GRASS Region ziemlich groß ist, kann es eine Weile dauern, bis der Layer vollständig dargestellt wird. Danach können Sie noch die Layereigenschaften verändern und eine Farbe für die Linien auswählen, die sich deutlich vom Höhenmodell unterscheidet, siehe *Der Eigenschaften-Dialog von Vektorlayern*.

Als nächstes zoomen Sie in einen bergigen Bereich im Zentrum Alaskas. Wenn Sie weit genug in die Karte hineingezoomt sind, werden Sie erkennen, dass die Höhenlinien teilweise sehr eckig erscheinen. Um das Erscheinungsbild zu optimieren, bietet GRASS ein Modul mit dem Namen **v.generalize**. Dabei wird mit Hilfe des Douglas Peuker Algorithmus und der einer Reduktion der Stützpunkte eine Glättung der Linien erreicht, ohne die Geometrien zu zerstören. Da der Ergebnislayer weniger Stützpunkte hat, ist er auch kleiner und kann schneller geladen werden. Die Analyse wird z.B. angewendet, wenn man sehr detailierte Daten nur in einem kleinen Maßstab anzeigen möchte.

> **Tipp**
>
> **Geometrien in QGIS vereinfachen**
>
> Beachten Sie, dass es in QGIS das Werkzeug *Vektor* ► *Geometrie-Werkzeuge* ► *Vereinfachen* gibt, dass ähnlich dem GRASS **v.generalize** Douglas-Peuker-Algorithmus arbeitet.

In diesem Beispiel wollen wir nun aber etwas anderes erreichen. Die Höhenlinien, die wir mit r.contour erstellt haben, zeigen teilweise sehr scharfe Winkel, die wir glätten möchten. Unter den Algorithmen des Moduls **v.generalize** befindet sich auch der Chaikens-Algorithmus, der exakt das macht, was wir möchten (auch Hermite Splines). Achten Sie aber darauf, dass es passieren kann, dass Stützpunkte nicht nur eliminiert sondern auch **hinzugefügt** werden können. Dadurch kann der Layer wieder langsamer geladen werden.

- Öffnen Sie die GRASS Werkzeugkiste und doppelklicken Sie die Kategorien *Vektor* ► *Karte entwickeln* ► *Generalisierung* und klicken Sie dann auf das **v.generalize** Modul um sein Optionsfenster zu öffnen.
- Stellen Sie sicher, dass ‚hoehen_100' als Vektorlayer in Feld *Name der Vektoreingabekarte* erscheint.

- Wählen Sie Chaiken's Algorithm aus der Liste der Generalisierungs Algorithmen. Lassen Sie alle anderen Optionen auf der Voreinstellung. Geben Sie in der untersten Zeile im Feld *Name der Ausgabe-Vektorkarte* ‚hoehen_100_smooth' ein und klicken auf *Starten*.
- Die Ausführung dauert einen Moment. Wenn `Erfolgreich beendet` im Ergebnisfenster erscheint, klicken Sie auf *View Output* und dann auf *Schließen*.
- Ändern Sie nun auch die Farbe des neuen Layers, damit er sich deutlich von dem Höhenmodell und den zuvor berechneten Höhenlinien abhebt. Sie werden erkennen, dass die Kanten der neuen Höhenlinien wesentlich weicher gezeichnet sind.

Abb. 22.7: GRASS Moduk v.generalize zum Glätten einer Vektorkarte

Tipp

Vektorlinien glätten mit dem GRASS Modul v.generalize

Die oben beschriebene Anwendung kann auch in anderen Situationen verwendet werden. Wenn Sie z.B. eine Rasterkarte mit Niederschlagswerten haben, können Sie mit r.contour einen Isohyetallayer (konstanter Niederschlag) erstellen.

Erstellen eines 3D Schummerungseffekts

Es gibt verschiedene Methoden, um Höhenlayer anzuzeigen und ihnen einen 3D Schummerungseffekt zu verleihen. Der Gebrauch von Höhenlinien ist eine populäre Methode, die häufig angewendet wird, um topographische Karten zu erstellen. Eine andere Möglichkeit, um einen 3D Effekt zu erzeugen ist, bietet das Hillshading. Der Effekt basiert dabei auf einem Höhenmodell. Dabei wird zuerst die Hangneigung und -richtung der Zellen bestimmt und dann durch die Simulation des Sonnenstandes eine Reflexionswert erzeugt. Dadurch werden der Sonne zugewandte Bereiche aufgehellt und der Sonne abgewandte Bereiche (im Schatten) dunkler dargestellt.

- Fangen Sie mit diesem Beispiel an indem Sie zuerst den Rasterlayer `gtopo30` laden. Öffnen Sie die GRASS-Werkzeugkiste und unter der Raster Kategorie wechseln Sie in den Bereich *Räumliche Analysen* ► *Geländeanalyse*.
- Nun klicken Sie auf **r.shaded.relief**, um den Modulreiter zu öffnen.
- Ändern Sie den Wert im Feld *Winkel der Sonne in Grad östlich von der Nordrichtung* 1,00 270 auf 315.
- Geben Sie `gtopo30_shade` für das neu zu erstellende Raster mit Schummerung ein und klicken auf *Starten*.
- Nachdem die Karte berechnet wurde, visualisieren Sie sie und setzen Sie die Farbe des Rasterlayers auf Graustufen.

- Um die Schummerung und das Höhenmodell `gtopo30` zusammen zu sehen, ziehen Sie die Schummerungskarte in der Legende unter das Höhenmodell. Öffnen Sie dann den Dialog Layereigenschaften der Karte `gtopo30`, und ändern Sie die Transparenz im Reiter *Transparenz* auf den Wert 25%.

Die Höhenkarte `gtopo30` wird nun als Farbkarte leicht transparent über der Schummerungskarte angezeigt. Dadurch entsteht ein visueller 3D Effekt. Um den Unterschied besser zu erkennen, wechseln Sie über das Kontrollkästchen den Anzeigemodus der Schummerungskarte in der Legende und wieder zurück.

Rasterstatistik auf Basis eines Vektorlayer berechnen

Das folgende Beispiel zeigt, wie man univariate Statistik für Rasterwerte innerhalb von Vektorpolygonen berechnen kann und die Werte in neue Attributspalten des Vektorlayers hinzufügt.

- Es werden wieder die Alaska Daten verwendet. Unter *Daten in eine GRASS LOCATION importieren* findet man Hinweise zum Import der Datei `shapefiles/trees.shp` nach GRASS.
- Jetzt ist ein Zwischenschritt nötig: es müssen Zentroiden zu der importierten Baumkarte hinzugefügt werden um es zu einem kompletten GRASS Flächenvektor (der sowohl Umrandungen als auch Zentroide enthält) zu machen.
- Wählen Sie aus der Werkzeugkiste *Vektor* ► *Karte entwickeln* ► *Objekte verwalten* und öffnen Sie das Modul **v.centroids**.
- Geben Sie als *Name für die Ausgabe-Vektorkarte* ‚forest_areas' an und starten Sie das Modul.
- Nun laden Sie den neuen Layer `forest_areas` und visualisieren Sie die verschiedenen Waldtypen in verschiedenen Farben - deciduous, evergreen und mixed. Dazu öffnen Sie den Dialog *Eigenschaften* des Layers, wechseln zum Reiter *Darstellung*, wählen ‚Eindeutiger Wert' und setzen das *Klassifikationsfeld* auf ‚VEGDESC' (siehe auch die Beschreibung des Reiters Darstellung in Abschnitt *Symbolisierung* des Vektordatenabschnitts).
- Als nächstes öffnen Sie wieder die GRASS-Werkzeugkiste und wechseln nach *Vektor* ► *Vektor mit anderen Karten aktualisieren*.
- Klicken Sie auf das **v.rast.stats** Modul. Geben Sie `gtopo30` und `forest_areas` ein.
- Ein weiterer Parameter wird benötigt: Geben Sie bei *column prefix* `elev` ein und klicken auf *Starten*. Dies ist eine rechenintensive Operation, die längere Zeit in Anspruch nimmt (wahrscheinlich bis zu zwei Stunden).
- Schließlich öffnen Sie den Layer `forest_areas` erneut und lassen Sie sich die Attributtabelle anzeigen. Dort gibt es nun weitere Spalten einschließlich `elev_min`, `elev_max`, `elev_mean` usw. für jedes Waldpolygon.

22.14.3 Anpassen der Module

Nahezu alle GRASS-Module können in die GRASS-Werkzeugkiste integriert werden. Eine XML-Schnittstelle wertet die sehr einfachen XML-Dateien, die die Module beschreiben, aus und übernimmt die Oberflächendarstellung.

Beispielhaft ist hier die XML-Datei zum Modul `v.buffer` (v.buffer.qgm) dargestellt:

```
<?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>
```

Der Parser liest diese Definition und erstellt eine neue Registerkarte in der Toolbox, wenn Sie das Modul auswählen.

KAPITEL 23

Die QGIS Verarbeitungsumgebung

23.1 Einführung

In diesem Kapitel wird die QGIS-Verarbeitungsumgebung vorgestellt, ein Geoprocessing-Framework, mit der QGIS-eigene und fremde Algorithmen aus QGIS heraus aufgerufen werden können, um räumliche Analyseaufgaben produktiver und einfacher zu gestalten.

Als *Kernplugin* ist die Verarbeitungsumgebung standardmäßig installiert, aber Sie müssen sie aktivieren:

1. Gehen Sie zu *Erweiterungen* ► *Erweiterungen verwalten und Installieren…*
2. Klicken Sie auf die Registerkarte *Installiert* auf der linken Seite
3. Markieren Sie das Kästchen neben dem Eintrag *Processing*
4. Schließen Sie den Dialog.

   In der oberen Menüleiste finden Sie jetzt den neuen Eintrag *Verarbeitung*. Von dort aus können Sie die Hauptkomponenten dieses Frameworks erreichen.

In den folgenden Abschnitten werden wir betrachten, wie die grafischen Elemente dieser Umgebung benutzt werden und wie man das Beste aus jedem einzelnen von ihnen herausholen kann.

Die grafische Benutzeroberfläche des Frameworks besteht aus vier Grundelementen, die zur Ausführung von Algorithmen für unterschiedliche Zwecke verwendet werden. Die Wahl des einen oder anderen Tools hängt von der Art der durchzuführenden Analyse und den besonderen Merkmalen des jeweiligen Benutzers und Projekts ab. Alle diese Elemente (mit Ausnahme der Schnittstelle für die Stapelverarbeitung, die über die Toolbox oder den Dialog zur Ausführung von Algorithmen aufgerufen wird, wie wir noch sehen werden) können über den Menüpunkt *Verarbeitung* aufgerufen werden (Sie werden weitere Einträge sehen; die übrigen werden nicht zur Ausführung von Algorithmen verwendet und werden später in diesem Kapitel erläutert).

- Die *Werkzeugkiste*: Das Hauptelement der grafischen Benutzeroberfläche, mit dem ein einzelner Algorithmus oder ein auf diesem Algorithmus basierender Stapelprozess ausgeführt werden kann.

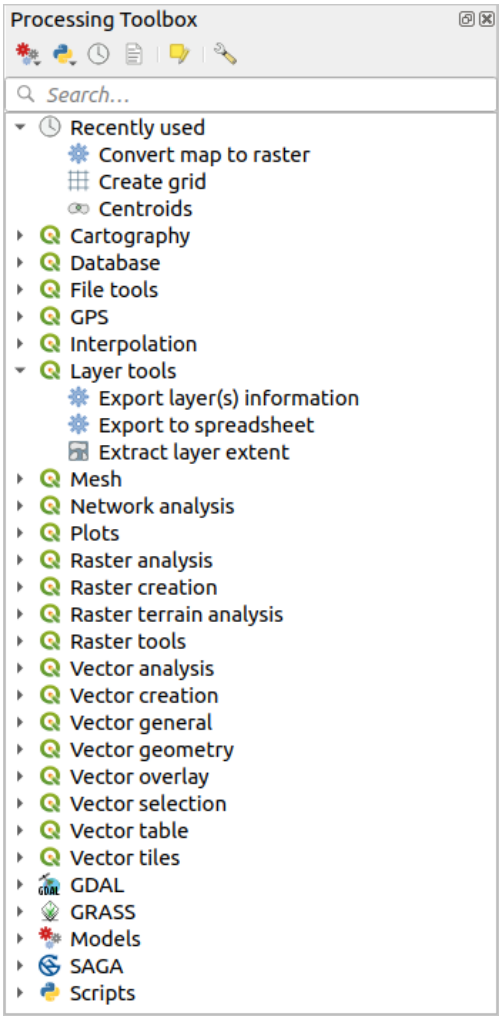


Abb. 23.1: Verarbeitungswerkzeuge

- *Modellentwurf*: Mehrere Algorithmen können mit Hilfe des Modellierers grafisch kombiniert werden, um einen Arbeitsablauf zu definieren, so dass ein einziger Prozess entsteht, der mehrere Unterprozesse umfasst.

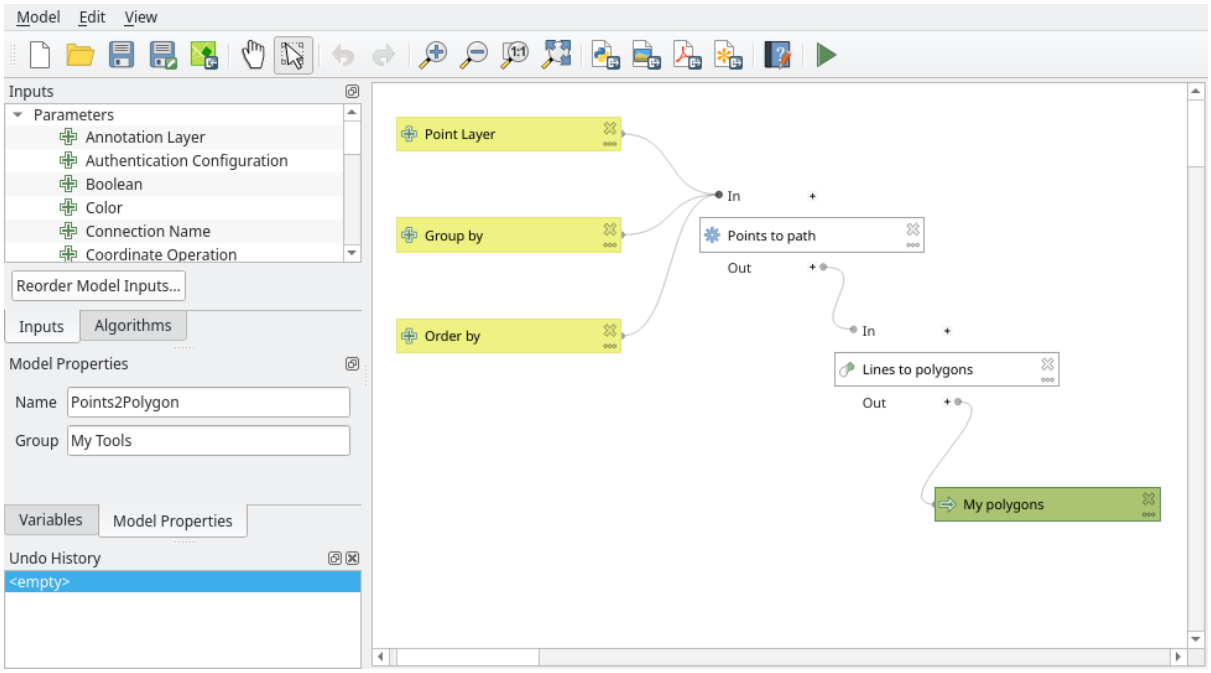


Abb. 23.2: Verarbeitungsmodellierung

- *Protokoll*: Alle Aktionen, die mit einem der oben genannten Elemente durchgeführt wurden, werden in einer Verlaufsdatei gespeichert und können später mit dem Protokoll-Manager leicht reproduziert werden.

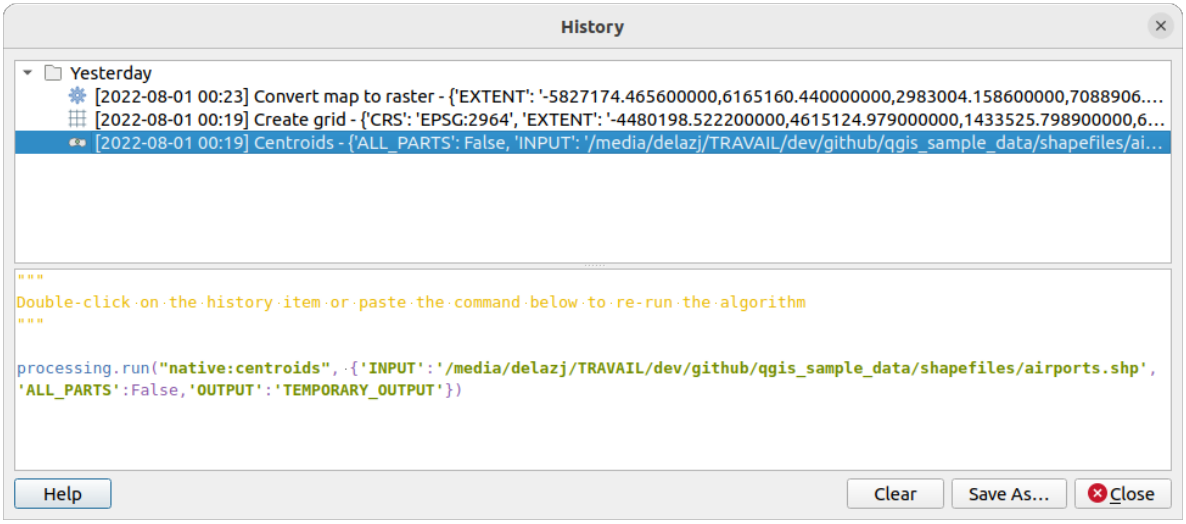


Abb. 23.3: Verarbeitungsprotokoll

- *Stapelverarbeitung*: Mit dieser Schnittstelle können Sie Batch-Prozesse ausführen und die Ausführung eines einzelnen Algorithmus auf mehreren Datensätzen automatisieren.

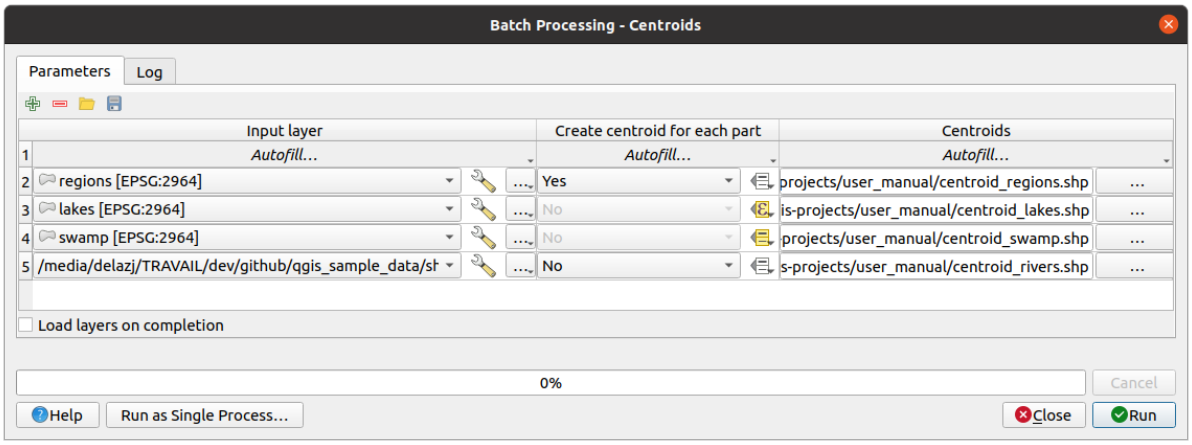


Abb. 23.4: Stabelverarbeitung

In den folgenden Abschnitten werden wir auf jedes dieser Elemente im Detail eingehen.

23.2 Konfigurieren der Verarbeitungsumgebung

Auf dem Reiter *Verarbeitung*, den Sie unter *Einstellungen*► *Optionen* ►) finden, können Sie konfigurieren, wie die Algorithmen funktionieren. Die Konfigurationsparameter sind in separate Blöcke unterteilt, die Sie auf der linken Seite des Dialogs auswählen können.

23.2.1 Allgemein

Der Block *Allgemein* enthält die Standardeinstellungen, um zu steuern, wie sich der Algorithmus-Dialog sowie die Eingabe- und Ausgabeparameter verhalten sollen. Einige der Einstellungen können jedoch auf der Ebene des Algorithmus überschrieben werden, für den *Algorithmusablauf* oder für *einzelne Parameter*.

- *Erweiterungsvoreinstellung für Rasterlayerausgaben* ist standardmäßig `tif`
- *Erweiterungsvoreinstellung für Vektorlayerausgaben* ist standardmäßig `gpkg`
- *Filterung ungültiger Objekte* bei der Ausführung von Algorithmen:
  - *Nicht filtern (schneller)*: alle Objekte (mit gültigen und ungültigen Geometrien) werden verarbeitet, aber das Ergebnis kann fehlerhaft sein, je nachdem, wie die Ungültigkeit der Geometrie die Operationen beeinflusst
  - guilabel:*Objekte mit ungültigen Geometrien ignorieren*, was bedeutet, dass nur eine Teilmenge Ihres Datensatzes (mit gültigen Geometriemerkmale) verarbeitet wird
  - *Ausführung bei ungültiger Geometrie anhalten*: Sie müssen die ungültigen Geometrien aufspüren und korrigieren, wenn Sie möchten, dass der Algorithmus den gesamten Layer verarbeitet. Algorithmen wie *Check validity* oder *Fix geometries* können Ihnen dabei helfen, dies zu erreichen.

  Die Einstellungen unter *Filterung ungültiger Objekte* kann zur Laufzeit des Algorithmus für jede einzelne Eingabe außer Kraft gesetzt werden.
- *Behalte Dialog nach Abschluss des Algorithmus offen*. Sobald die Ausführung eines Algorithmus beendet ist und die ausgegebenen Layer in das QGIS Projekt geladen sind, wird der Algorithmus-Dialog geschlossen. Wenn Sie ihn geöffnet lassen möchten (um den Algorithmus mit anderen Parametern erneut auszuführen oder um die Ausgabe, die in den Log-Tab geschrieben wird, besser zu überprüfen), aktivieren Sie diese Option.
- *Maximale Threads*

- *Ausgabeverzeichnis* für nicht temporäre Ausgaben: Wenn kein Ordnerpfad für die Ausgaben der Verarbeitung angegeben ist, werden diese in diesem Ordner gespeichert. Der Standardordner ist `processing` im Benutzerverzeichnis (z. B. `C:\Users\1\processing\` unter Windows).
- *Temporäres Ausgabeverzeichnis übersteuern*: Temporäre Ausgaben werden standardmäßig im Ordner `tmp` auf dem Rechner gespeichert. Mit dieser Option können Sie einen anderen Ort für die Speicherung festlegen.
- *Skript zur Ausführung vor dem Run* und *Skript zur Ausführung nach Lauf*. Diese Parameter verweisen auf Dateien, die Skripte enthalten, die mithilfe der Verarbeitungsskriptfunktionalität geschrieben wurden, die im Abschnitt über Skripting und die Konsole erläutert wird.
- *Ausgabedateinamen für Layernamen bevorzugen*. Der Name jedes resultierenden Layers, der von einem Algorithmus erzeugt wird, wird vom Algorithmus selbst festgelegt. In einigen Fällen kann ein fester Name verwendet werden, was bedeutet, dass derselbe Name für die Ausgabe verwendet wird, unabhängig davon, welcher Input Layer verwendet wird. In anderen Fällen kann der Name vom Namen des Input-Layers oder von einigen der Parameter abhängen, die zur Ausführung des Algorithmus verwendet werden. Wenn dieses Kontrollkästchen aktiviert ist, wird der Name stattdessen aus dem Dateinamen der Ausgabe übernommen. Wenn die Ausgabe in einer temporären Datei gespeichert wird, ist der Dateiname dieser temporären Datei in der Regel lang und zufällig, um Kollisionen mit anderen bereits vorhandenen Dateinamen zu vermeiden.
- *Ergebnisgruppenname*. Wenn Sie alle Layer der Verarbeitungsergebnisse in einer Gruppe im Bedienfeld *Layer* abrufen möchten, geben Sie einen Gruppennamen für diesen Parameter an. Die Gruppe kann bereits existieren oder nicht. QGIS fügt alle ausgegebenen Layer zu einer solchen Gruppe hinzu. Standardmäßig ist dieser Parameter leer, so dass alle Ausgabe-Layer an verschiedenen Stellen im *Layer* Bedienfeld hinzugefügt werden, abhängig von dem Element, das beim Ausführen eines Algorithmus aktiv ist. Beachten Sie, dass die Layer nur dann in das Bedienfeld *Layer* geladen werden, wenn *Ausgabe-Datei nach Ausführung des Algorithmus öffnen* im Algorithmus-Dialog aktiviert ist.
- *Algorithmen mit Problemen anzeigen*: Standardmäßig vermeidet QGIS die Anzeige von fehlerhaften Algorithmen (in der Regel von Drittanbietern). Wenn diese Option aktiviert ist, werden sie in der Processing Toolbox mit einem Warnsymbol und einem Tooltip angezeigt, der erklärt, dass sie Probleme haben. Die Verwendung erfolgt auf eigene Gefahr.
- guilabel:*Objektanzahl für Ausgabevektorlayer anzeigen*. Da die Berechnung der Anzahl der Objekte bei bestimmten Datenformaten einige Zeit in Anspruch nehmen kann, ist diese Option standardmäßig deaktiviert.
- *Zeige Layer Koordinatensystem in Auswahlkästen*
- *Tooltipp anzeigen, wenn es abgeschaltete Anbieter gibt*
- *Stil für Linien-Layer*, *Stil für Punkt-Layer*, *Stil für Polygon-Layer* und *Stil für Raster-Layer* werden verwendet, um den Standard-Rendering-Stil für Ausgabe-Layer (d.h. Layer, die von Verarbeitungsalgorithmen erzeugt werden) festzulegen. Erstellen Sie einfach den gewünschten Stil mit QGIS, speichern Sie ihn in einer Datei und geben Sie dann den Pfad zu dieser Datei in den Einstellungen an, damit die Algorithmen ihn verwenden können. Immer wenn ein Layer von Processing geladen und zur Kartenansicht von QGIS hinzugefügt wird, wird er mit diesem Stil gerendert.

  Rendering-Stile können für jeden Algorithmus und jede seiner Ausgaben individuell konfiguriert werden. Klicken Sie einfach mit der rechten Maustaste auf den Namen des Algorithmus in der Toolbox und wählen Sie *Renderingstile für Ausgaben bearbeiten*. Daraufhin wird ein Dialogfeld wie das folgende angezeigt.

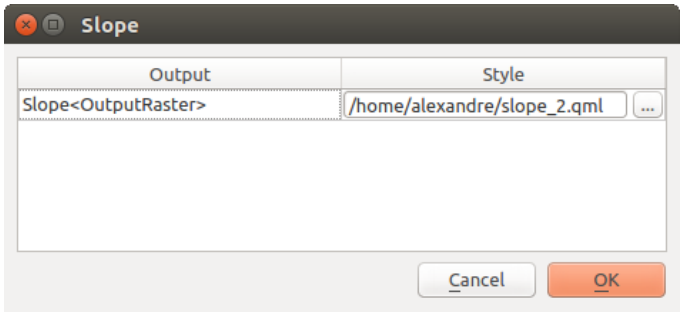


Abb. 23.5: Darstellungsstile

Wählen Sie die Stildatei (`.qml`), die Sie für jede Ausgabe wünschen, und drücken Sie *OK*.

- *Vor der Ausführung warnen, wenn Parameter unterschiedlicher KBSe verwendet werden*: Standardmäßig projizieren die nativen Algorithmen von QGIS (d.h. diejenigen, die unter der Gruppe *Menus* ► *QGIS (native C++)* aufgelistet sind) die Input Layer vor der Ausführung transparent auf das erste KBS. Aktivieren Sie diese Option, um eine Benachrichtigung von anderen Tools zu erhalten, die keine Reprojektion unterstützen, wenn die KBS der Eingaben nicht identisch sind. Anbieter von Drittanbietern sind davon nicht betroffen.

23.2.2 Menüs

Der Block *Menüs* steuert, ob ein Algorithmus, ein Skript oder ein Modell (eingebaut oder von Plugins bereitgestellt) über ein eigenes Menü oder eine Symbolleiste (zusammen mit der Processing Toolbox) verfügbar gemacht werden soll. Für jedes Element eines jeden Anbieters können Sie:

- *Schaltfläche in der Symbolleiste hinzufügen*, sodass sie in der Werkzeugleiste *Verarbeitungsalgorithmen* verfügbar ist
- dem Algorithmus ein *Icon* zuweisen
- Setzen Sie *Menüpfad*: Der Algorithmus ist dann über ein vorhandenes oder benutzerdefiniertes Menü verfügbar, z. B. fügt `Vektor/MyTopAlgorithms` eine Verknüpfung im Untermenü *MyTopAlgorithms* des Standardmenüs *Vektor* hinzu.

Starten Sie QGIS neu, um die Einstellungen zu übernehmen. Sie können Ihre Änderungen jederzeit mit *Rücksetzen auf Voreinstellungen* rückgängig machen.

23.2.3 Modelle und Skripte

In den Blöcken *Models* und *Scripts* wird ein Standardpfad bereitgestellt, der unter dem aktiven Ordner *Benutzerprofil* festgelegt ist und zum Speichern von Modellen bzw. Skripten dient. Wie bei den anderen Optionen können Sie den Pfad ändern, z. B. um zu einem Remote- oder freigegebenen Ordner für alle Ihre Benutzer zu führen.

23.2.4 Anbieter

Sie finden auch einen Block für den Algorithmus *Providers*. Hier zeigen die installierten Anbieter ihre Einstellungen an.

Standardmäßig wird QGIS mit zwei Algorithmus-Anbietern installiert:

- *GDAL*, dessen Algorithmen Sie in diesem Dialogfeld *aktivieren* (oder nicht) können. Weitere Informationen finden Sie unter *GDAL-Algorithmen*.
- *GRASS*, dessen Algorithmen die Installation des *GRASS GIS Processing Provider* aus dem *Plugins Manager* erfordern. Anschließend können Sie einige spezielle Einstellungen vornehmen, wie zum Beispiel:
  - *Für Raster-Layer verwenden Sie r.external (schneller) anstelle von r.in.gdal*
  - *Für Vector-Layer verwenden Sie v.external (schneller) anstelle von v.in.ogr*
  - *Speicherort der GRASS-Dokumentation*: Standardmäßig und abhängig von Ihrem Betriebssystem verwendet QGIS eine fest codierte Liste von Pfaden, um die lokale GRASS-Dokumentation zu finden und zu öffnen, wenn Sie auf die Schaltfläche *Hilfe* in einem GRASS-Algorithmus-Dialogfeld klicken. Durch Ausfüllen dieser Option wird der Standardpfad überschrieben, sodass Sie einen benutzerdefinierten Speicherort oder eine Remote-Dokumentation verwenden können.
  - *Konsolenausgabe protokollieren*
  - *Befehle zur Protokollierung der Ausführung*

Installierte Plugins, die Verarbeitungsalgorithmen bereitstellen, können ebenfalls in dieser Gruppe mit benutzerdefinierten Einstellungen aufgeführt sein.

23.3 Die Werkzeugkiste

Die *Verarbeitungswerkzeuge* (*Verarbeitung* ► *Werkzeugkiste*) ist das Hauptelement der Verarbeitungs-GUI und dasjenige, das Sie bei Ihrer täglichen Arbeit am ehesten verwenden werden. Sie zeigt eine Liste aller verfügbaren **Algorithmen**, gruppiert in verschiedenen Blöcken, die *Anbieter* genannt werden, sowie benutzerdefinierte **Modelle** und **Skripte**, die Sie hinzufügen können, um die Palette der Werkzeuge zu erweitern. Die Toolbox ist somit der Zugangspunkt, um sie auszuführen, sei es als einzelner Prozess oder als Batch-Prozess, der mehrere Ausführungen desselben Algorithmus auf verschiedenen Eingabesätzen umfasst.

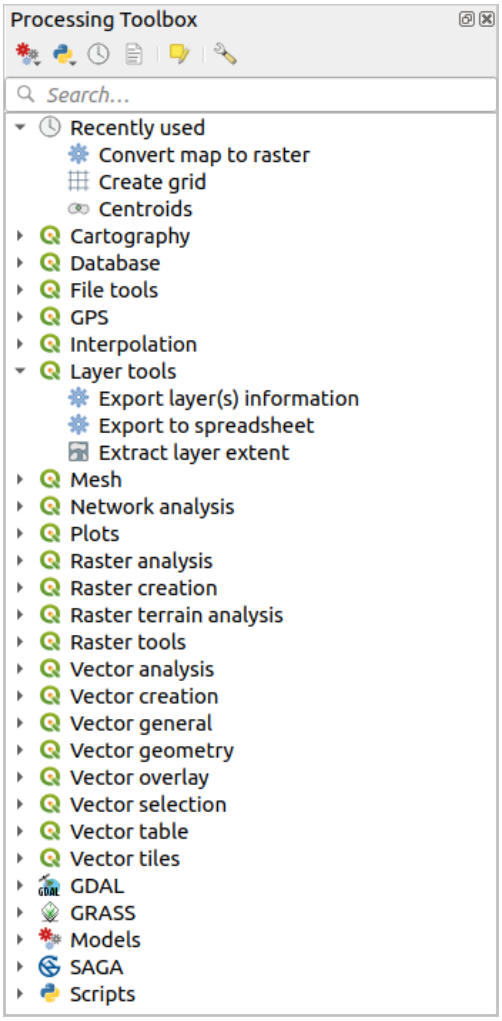


Abb. 23.6: Verarbeitungswerkzeuge

Anbieter können im Dialog *Verarbeitungseinstellungen* (de)aktiviert werden. Standardmäßig sind nur Provider aktiv, die nicht auf Anwendungen von Drittanbietern angewiesen sind (d.h. solche, die nur QGIS-Elemente zur Ausführung benötigen). Algorithmen, die externe Anwendungen benötigen, müssen möglicherweise zusätzlich konfiguriert werden. Die Konfiguration von Providern wird in einem *späteren Kapitel* in diesem Handbuch erklärt.

Im oberen Teil des Werkzeugkastens finden Sie eine Reihe von Werkzeugen:

- mit Modelle arbeiten: *Erzeuge neues Modell…*, *Vorhandenes Modell öffnen…* und *Modell zur Werkzeugkiste hinzufügen…*;
- mit Skripte arbeiten: *Erzeuge neues Skript…*, *Neues Skript aus Vorlage erzeugen…*, *Vorhandenes Skript öffnen…* und *Skript zur Werkzeugkiste hinzufügen…*;
- öffnen des Protokoll-Fensters;
- öffnen der Ergebnisanzeige;
- schalten Sie die Werkzeugkiste mit der Schaltfläche Objekte insitu ändern auf die *gefilterte Ansicht* um: Es werden nur die Algorithmen angezeigt, die für die Ausführung auf dem aktiven Layer geeignet sind, ohne dass ein neuer Layer ausgegeben wird;
- öffnen des Optionen - Verarbeitung-Dialogs.

Unterhalb dieser Symbolleiste befindet sich ein Feld für die Suche, mit dem Sie die benötigten Werkzeuge leicht finden können. Sie können jedes Wort oder jeden Satz in das Textfeld eingeben. Beachten Sie, dass sich die Anzahl der Algorithmen, Modelle oder Skripte in der Toolbox während der Eingabe auf diejenigen reduziert, die den von Ihnen eingegebenen Text in ihrem Namen oder Schlüsselwort enthalten.

Oben in der Algorithmenliste werden die am häufigsten verwendeten Tools Zuletzt verwendet und die Favoriten angezeigt, sodass Sie diese bequem erneut ausführen können. Die Liste Zuletzt verwendet‘ ist standardmäßig vorhanden und zeigt die Algorithmen an, die Sie zuletzt ausgeführt haben. Die Liste |favorites| :sup:`Favoriten wird erstellt, wenn Sie einen Algorithmus manuell hinzufügen

Sie können mit der rechten Maustaste auf einen beliebigen Algorithmus in der Toolbox klicken und aus dem Kontextmenü folgende Optionen auswählen:

- *Ausführen…*, um den Algorithmus auszuführen.
- *Als Batch-Prozess ausführen…*, um den Algorithmus im Batch-Modus auszuführen.
- *Rendering-Stile für Ausgaben bearbeiten…*, um die Rendering-Stile für die Ausgaben des Algorithmus anzupassen.
- *Zu Favoriten hinzufügen*, um den Algorithmus zur Favoritenliste hinzuzufügen.
- *Aus Favoriten entfernen*, um den Algorithmus aus der Favoritenliste zu entfernen.

Im Gegensatz zur Liste „Zuletzt verwendet“ bleibt die Favoritenliste unverändert und wird nicht davon beeinflusst, welche Algorithmen ausgeführt wurden.

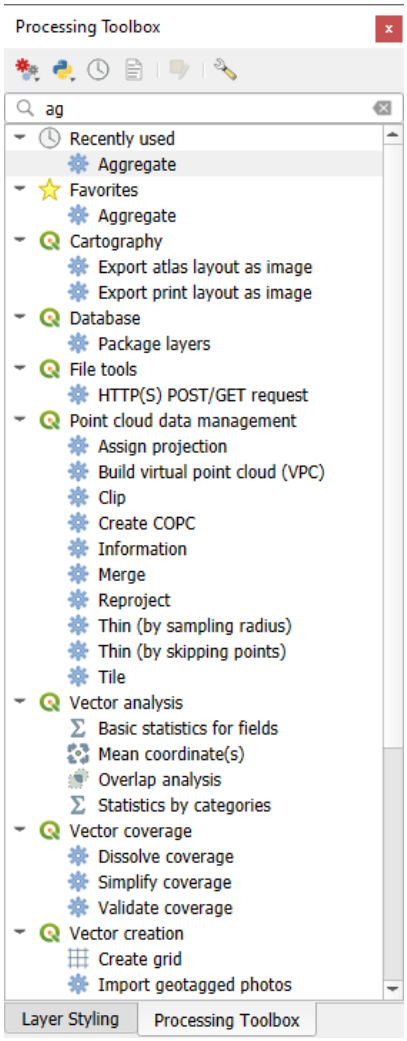


Abb. 23.7: Im Fenster Verarbeitungswerkzeuge werden nur passende Suchergebnisse angezeigt

Um ein Werkzeug auszuführen, doppelklicken Sie einfach auf den Namen des Werkzeugs in der Toolbox.

23.3.1 Der Algorithmus Dialog

Nach einem Doppelklick auf den Namen des Algorithmus, den Sie ausführen möchten, wird ein Dialogfeld ähnlich dem in Abb. 23.8 unten angezeigt (in diesem Fall entspricht das Dialogfeld dem Algorithmus `Zentroide`).

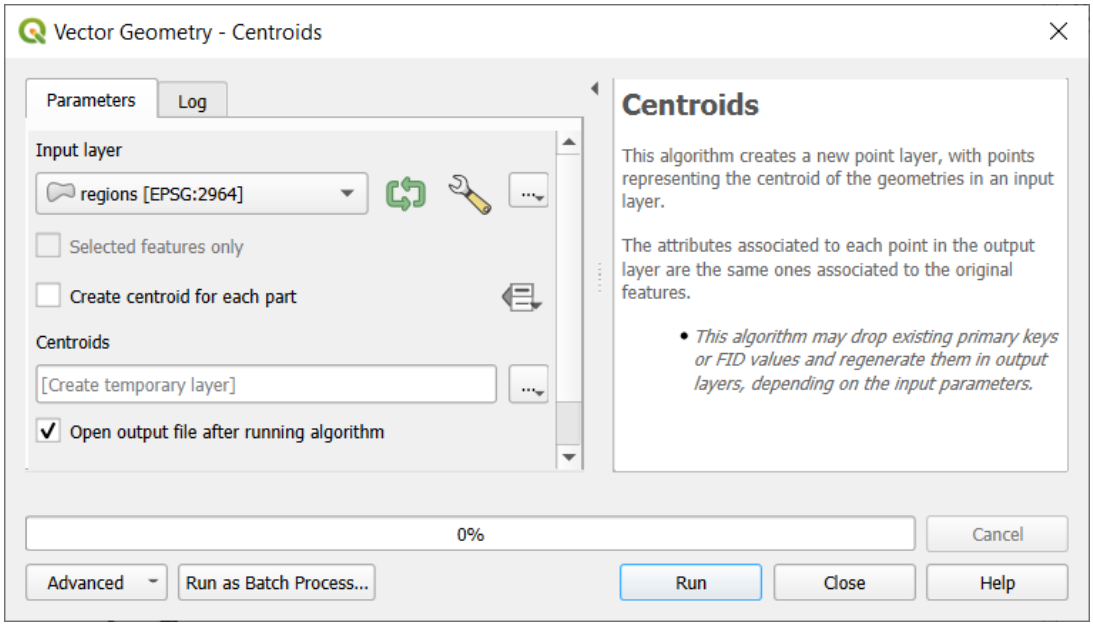


Abb. 23.8: Algorithmus-Dialog - Parameter-Reiter

Der Dialog zeigt auf der linken Seite zwei Reiter (*Parameter* und *Protokoll*), auf der rechten Seite die Algorithmusbeschreibung und am unteren Rand eine Reihe von Schaltflächen.

Parameter-Typen

Die Registerkarte *Parameter* dient zum Einstellen der Eingabewerte, die der Algorithmus zur Ausführung benötigt. Sie zeigt eine Liste von Eingabewerten und Konfigurationsparametern, die eingestellt werden müssen. Je nach den Anforderungen des auszuführenden Algorithmus hat sie natürlich einen anderen Inhalt und wird automatisch auf der Grundlage dieser Anforderungen erstellt.

Tipp

Eigene Standardwerte für Algorithmusparameter festlegen

Algorithmus-Dialoge öffnen sich mit einigen Parametern, die mit Werten aus der QGIS-Installation vorbelegt sind. Es ist jedoch möglich, *Ihre eigenen Standardwerte* für bestimmte Algorithmus-Parameter zu setzen, sodass sie beim Start des Algorithmus verwendet werden.

Obwohl die Anzahl und Art der Parameter von den Eigenschaften des Algorithmus abhängt, ist die Struktur für alle ähnlich. Die Parameter, die Sie in der Tabelle finden, können folgende Typen sein:

- Ein **Vektorlayer**, um aus einer Liste aller in QGIS verfügbaren (aktuell geöffneten) Vektorlayer auszuwählen. Sie können auch ungeladene Layer verwenden: Drücken Sie die Schaltfläche … auf der rechten Seite des Widgets und wählen Sie:
  - *Datei auswählen…*: wählt eine Datei auf der Festplatte mit dem Datei-Explorer des Betriebssystems aus
  - *Layer suchen…*: öffnet das *Browser Bedienfeld*, das es erlaubt, die Layer direkt aus Datenbankquellen (PostgreSQL, SQL Server, Oracle, …), Webservices (WFS, AFS, …) oder Dateien auf der Festplatte zu übernehmen.

Abb. 23.9: Vektorlayer-Auswahlfeld

Tipp

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.

Bemerkung

Standardmäßig zeigt das Layer-Widget das KBS des Layers zusammen mit seinem Namen an. Wenn Sie diese zusätzlichen Informationen nicht sehen möchten, können Sie diese Funktion im Dialogfeld Verarbeitungseinstellungen deaktivieren, indem Sie die Option *Allgemein ► Zeige Layer Koordinatensystem in Auswahlkästchen* deaktivieren.

Das Vektor-Eingabe-Widget hat außerdem folgende Elemente:

- eine Schaltfläche Iteration : Wenn diese Schaltfläche aktiviert ist, wird der Algorithmus wiederholt für jedes seiner Objekte ausgeführt, anstatt nur einmal für den gesamten Layer, und erzeugt so viele Ausgaben wie der Algorithmus ausgeführt wird. Auf diese Weise kann der Prozess automatisiert werden, wenn alle Objekte in einem Layer separat verarbeitet werden müssen. Enthält der Algorithmus mehrere Eingabevektoren, über die iteriert werden kann, wird die Iteration nur für den ersten aktivierten Parameter durchgeführt, und zwar in der Reihenfolge, in der die Parameter im Algorithmus angegeben sind.
- **|Optionen|** Fortgeschrittene Optionen, um Einstellungen für diesen spezifischen Parameter vorzunehmen. Diese Einstellungen betreffen:
  - *Filterung ungültiger Objekte*: ermöglicht es, die *Standardmethode* für die Behandlung von Objekten mit ungültigen Geometrien zu überschreiben
  - guilabel:*Zu verarbeitende Objekte beschränken*: optionale Begrenzung der Anzahl der verarbeiteten Objekte aus der Quelle
  - *Objektfilter*: ermöglicht die Eingabe eines Ausdrucks zur dynamischen Unterteilung des Layers beim Ausführen des Tools, sodass keine separaten Schritte zur Festlegung von Layer-Filtern oder zur Erstellung von Layer-Untergruppen erforderlich sind.

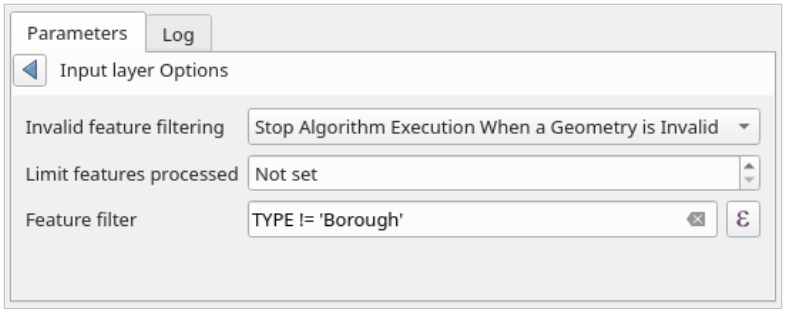


Abb. 23.10: Erweiterte Optionen für das Vektor-Eingabe-Widget

- Es ist auch möglich, die Ausführung des Algorithmus auf *Nur gewählte Objekte* zu beschränken.

- Eine **Tabelle**, um aus einer Liste aller in QGIS verfügbaren auszuwählen. Nicht-räumliche Tabellen werden in QGIS wie Vektor-Layer geladen und verwenden das *gleiche Werkzeug*.
- Ein **Rasterlayer**, zur Auswahl aus einer Liste aller in QGIS verfügbaren Layer. Der Selektor enthält auch einen … Button auf der rechten Seite, mit dem Sie Dateinamen auswählen können, die Layer repräsentieren, die derzeit nicht in QGIS geladen sind.

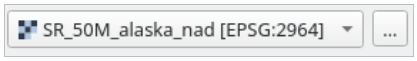


Abb. 23.11: Raster-Eingabe-Widget

- Eine **Option**, um aus einer Auswahlliste von möglichen Optionen auszuwählen.
- Ein **numerischer Wert**, der in ein Eingabefeld eingegeben wird. In einigen Kontexten (wenn der Parameter auf der Ebene des Objekts und nicht des Layers gilt), finden Sie eine Schaltfläche Datendefinierte Übersteuerung an seiner Seite, die es Ihnen ermöglicht, den *Ausdruckseditor* zu öffnen und einen mathematischen Ausdruck einzugeben, um variable Werte für den Parameter zu erzeugen. Einige nützliche Variablen, die sich auf die in QGIS geladenen Daten beziehen, können zu Ihrem Ausdruck hinzugefügt werden, sodass Sie einen von einer dieser Variablen abgeleiteten Wert auswählen können, wie z.B. die Zellengröße eines Layers oder die nördlichste Koordinate eines anderen Layers.

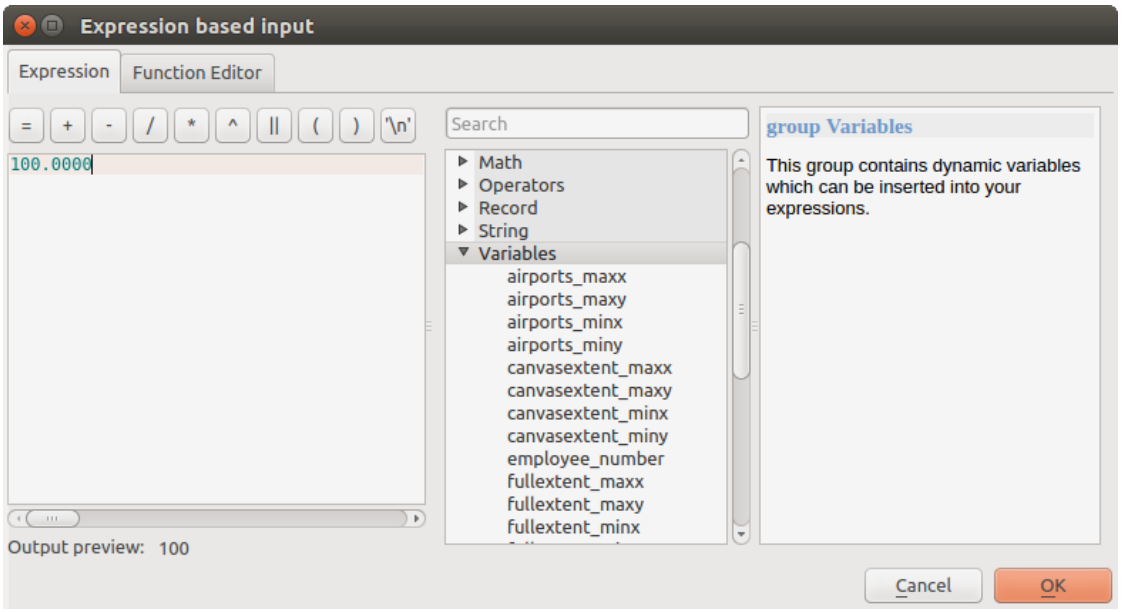


Abb. 23.12: Ausdrucksbasierte Eingabe

- Ein **Bereich** mit Minimal- und Maximalwerten, die in zwei Feldern eingegeben werden.
- Eine **Textzeichenfolge**, die in ein Textfeld eingefügt werden kann.
- Ein **Feld**, um aus der Attributtabelle eines Layer oder einer einzelnen Tabelle zu wählen, die in einem anderen Parameter ausgewählt wurde.
- Ein **Koordinatenbezugssystem**. Sie können es unter den zuletzt verwendeten aus der Dropdown-Liste oder aus dem Dialogfeld *KBS-Auswahl* auswählen, das erscheint, wenn Sie auf die Schaltfläche auf der rechten Seite klicken.

> **Tipp**
>
> **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.

- Ein **Ausdehnung**, ein Eingabefeld, das ein Rechteck durch seine Eckkoordinaten im Format `xmin, xmax, ymin, ymax` definiert. Drücken Sie die Schaltfläche Auf aktuelle Kartenansicht setzen, um die Kartenansicht zu verwenden. Wenn Sie auf den Pfeil auf der rechten Seite des Werteselektors klicken, erscheint ein Pop-up-Menü, das Ihnen folgende Optionen bietet:
  - menuselection:*Aus Layer berechnen* –>: füllt das Textfeld mit den Koordinaten des Begrenzungsrahmens eines Layers, der aus den geladenen Layern ausgewählt werden kann
  - menuselection:*Aus Layoutkarte berechnen* –>: füllt das Textfeld mit den Koordinaten eines aus einem Layout im aktuellen Projekt ausgewählten Kartenelements
  - *Berechnen aus Lesezeichen* ►: füllt das Textfeld mit den Koordinaten eines gespeicherten Lesezeichens
  - *Aktuellen Umfang der Kartenansicht verwenden*
  - *Zeichnen auf der Kartenansicht*: Das Parameterfenster wird ausgeblendet, so dass Sie auf die Kartenansicht klicken und ziehen können. Sobald Sie das Ausdehnungsrechteck definiert haben, erscheint das Dialogfeld wieder und enthält die Werte im Textfeld für die Ausdehnung.

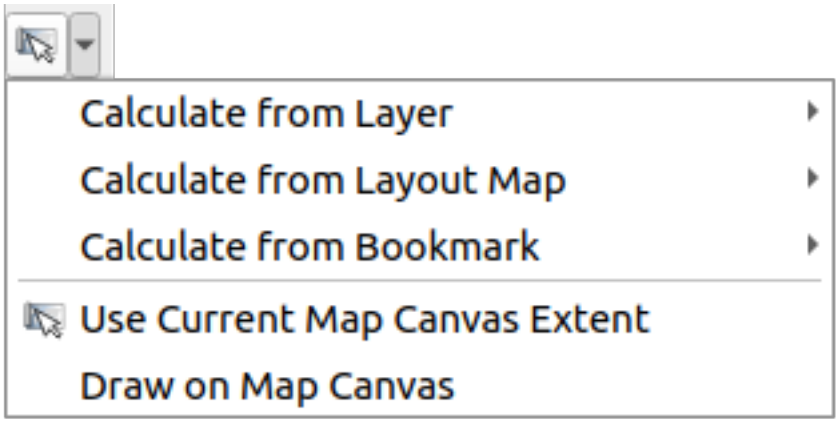


Abb. 23.13: Ausdehnung Auswahl

- Eine **Liste von Elementen** (Raster- oder Vektor-Layer, Tabellen, Felder), aus der Sie auswählen können. Klicken Sie auf die Schaltfläche … links neben der Option, um einen Dialog wie den folgenden anzuzeigen. Die Mehrfachauswahl ist erlaubt, und wenn der Dialog geschlossen wird, wird die Anzahl der ausgewählten Elemente im Parametertextfeld-Widget angezeigt.

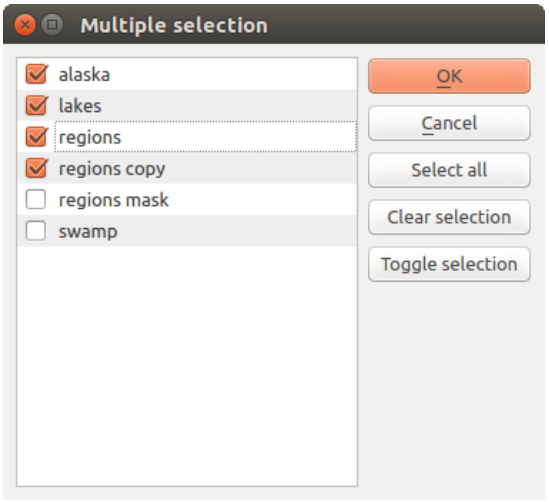


Abb. 23.14: Mehrfachauswahl

- Eine **kleine Tabelle**, die vom Benutzer bearbeitet werden kann. Diese werden u.a. zur Definition von Parametern wie Lookup-Tabellen verwendet.

  Klicken Sie auf den Knopf auf der rechten Seite, um die Tabelle zu sehen und zu editieren.

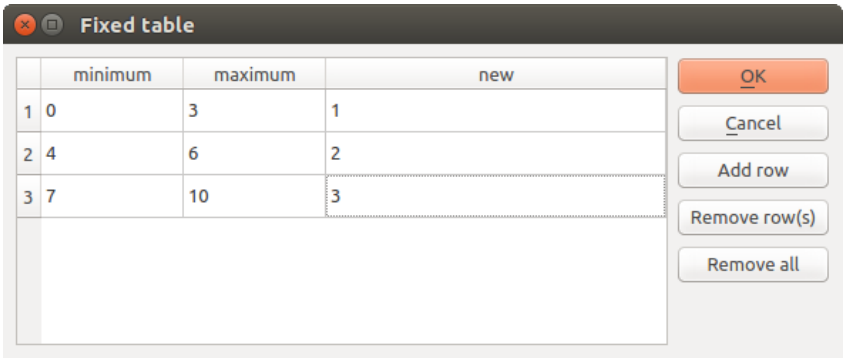


Abb. 23.15: Tabelle

In Abhängigkeit vom Algorithmus kann die Anzahl der Zeilen verändert werden oder auch nicht, indem Sie auf den Knopf rechts neben dem Fenster klicken.

Bemerkung

Einige Algorithmen benötigen viele Parameter, z.B. müssen Sie bei *Raster calculator* manuell die Zellengröße, den Umfang und das KBS angeben. Sie können die manuelle Auswahl aller Parameter vermeiden, wenn der Algorithmus über den Parameter `Referenz Layer` verfügt. Mit diesem Parameter können Sie einen Layer auswählen, dessen Eigenschaften (Zellgröße, Ausdehnung, KBS) verwendet werden.

Protokollierung der Ausführung

Neben der Registerkarte *Parameters* gibt es eine weitere Registerkarte namens *Log* (siehe Abb. 23.16 unten). In dieser Registerkarte werden Informationen angezeigt, die der Algorithmus während seiner Ausführung bereitstellt. So können Sie die Ausführung verfolgen und erhalten während der Ausführung weitere Details zum Algorithmus. Sie können direkt auf die Namen der im Protokoll aufgeführten Ausgabedateien, Ordner oder HTML-Dateien klicken. Dadurch wird der Ordner mit der generierten Datei geöffnet und diese automatisch ausgewählt. Informationen zur Ausführung des Algorithmus werden auch unter *Ansicht* ► *Bedienfelder* ► *Protokollmeldungen* ausgegeben

Beachten Sie, dass nicht alle Algorithmen Informationen in die Registerkarte *Protokoll* schreiben, und viele von ihnen laufen möglicherweise im Stillen, ohne dass sie eine andere Ausgabe als die endgültigen Dateien erzeugen. In diesem Fall sollten Sie das Bedienfeld *Protokoll* überprüfen.

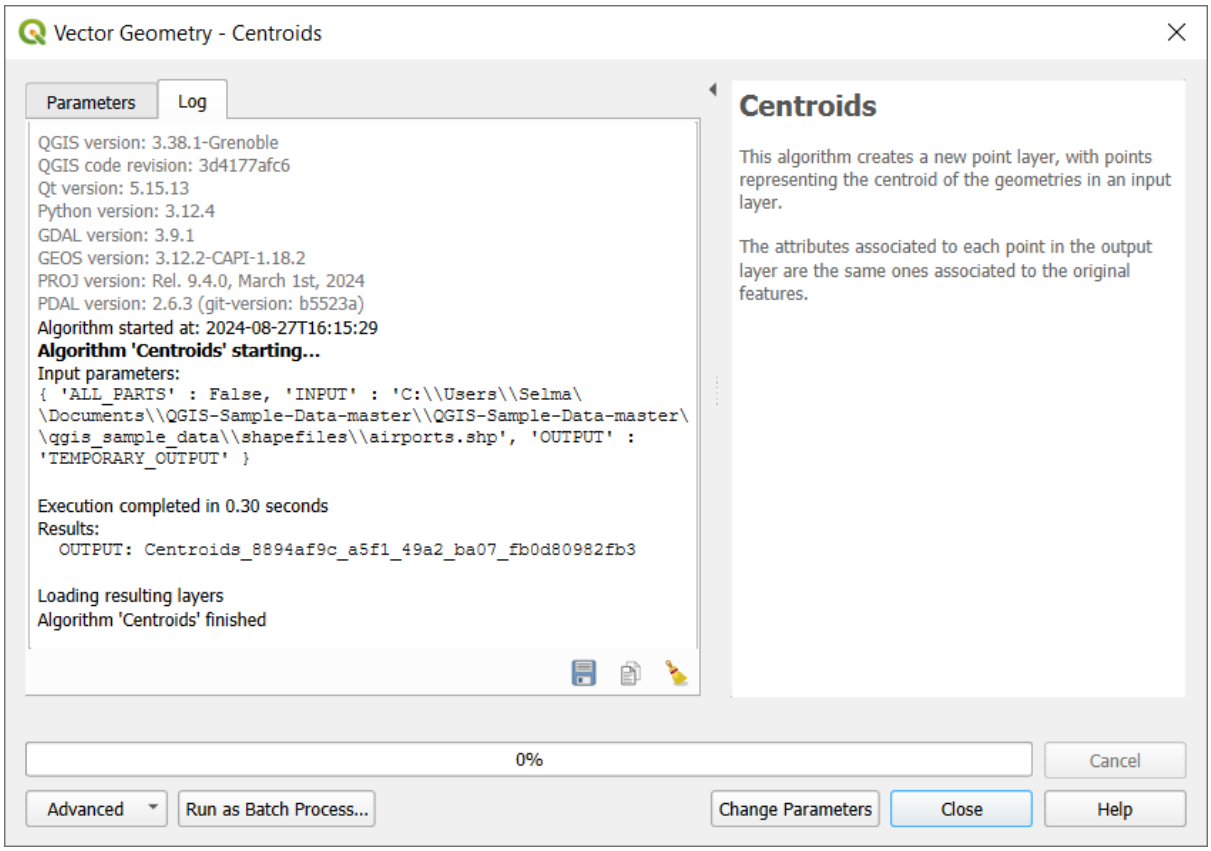


Abb. 23.16: Algorithmus-Dialog - Protokoll-Reiter

Am unteren Rand der Registerkarte *Protokoll* finden Sie die Schaltflächen *Protokoll in Datei speichern*, *Protokoll in Zwischenablage kopieren* und *Protokoll löschen*. Diese sind besonders praktisch, wenn Sie die Option *Dialog nach Ausführung des Algorithmus offen lassen* im *Allgemein* Teil der Verarbeitungsoptionen aktiviert haben.

Andere Werkzeuge

Auf der rechten Seite des Dialogs finden Sie eine kurze Beschreibung des Algorithmus, die Ihnen hilft, seinen Einsatzzweck und seine Grundideen zu verstehen. Wenn eine solche Beschreibung nicht verfügbar ist, wird das Bedienfeld nicht angezeigt.

Für eine detailliertere Hilfedatei, die z. B. eine Beschreibung aller verwendeten Parameter oder Beispiele enthält, finden Sie am unteren Rand des Dialogs eine Schaltfläche *Hilfe*, die Sie zur *Dokumentation der Verarbeitungsalgorithmen* oder zur Dokumentation des Anbieters (für einige Drittanbieter) führt.

Das Ausklappmenü *Erweitert* unten links auf dem Fenster bietet Funktionen zur Wiederverwendung der im Dialog definierten Konfiguration, ohne den Algorithmus auszuführen:

- *Algorithmeneinstellungen…*: erlaubt das Überschreiben von Verarbeitungseinstellungen für die aktuelle Algorithmusausführung. Weitere Einzelheiten unter *Algorithmus-Einstellungen überschreiben*.
- *Als Python-Befehl kopieren*: ermöglicht das einfache Kopieren des entsprechenden *PyQGIS-Befehls*, um das Werkzeug mit den im Dialog definierten Parametern auszuführen
- *Als qgis_process-Befehl kopieren*: erlaubt die einfache Generierung von *qgis_process Kommando*, inklusive der Umgebungseinstellungen wie Entfernungseinheiten, Flächeneinheiten, Ellipsoid, und komplexer Parameterwerte wie GeoPackage Ausgaben mit bestimmten Layern
- *Als JSON kopieren*: alle Einstellungen des Befehls werden in ein `JSON` Format kopiert, bereit, von qgis_process verwendet zu werden. Dies ist ein bequemer Weg, um das erwartete Format der Befehle zu sehen, sogar für komplexe Parameter (wie TIN Interpolationsparameter). Sie können diese einfach speichern und später wiederherstellen, indem Sie die Werte einfügen.
- *Einstellungen einfügen* in einem `JSON` Format

Die Schaltfläche *Als Batch-Prozess ausführen…* löst den *Batch-Verarbeitungsmodus* aus, in dem mehrere Instanzen des Algorithmus mit einer Vielzahl von Parametern konfiguriert und ausgeführt werden können. Mit der Schaltfläche *Als Einzelprozess ausführen…* können Sie vom Batch-Modus zurückwechseln.

Wenn die Ausführung eines Algorithmus beendet ist (entweder erfolgreich oder nicht), wird eine neue Schaltfläche *Parameter ändern* angezeigt, solange die Registerkarte *Protokoll* aktiv ist.

Algorithmus-Einstellungen überschreiben

Ausgelöst durch das *Erweitert*-Dropdown-Menü am unteren Rand eines Algorithmus-Dialogs, zeigt die Option *Algorithmuseinstellungen…* ein Bedienfeld, das dem Benutzer erlaubt, allgemeine Verarbeitungseinstellungen zu kontrollieren, die nur für die Ausführung dieses Algorithmus gelten. Es ist als Ort gedacht, an dem ein Benutzer seine *Globalen Verarbeitungseinstellungen* ad-hoc überschreiben kann, ohne seine üblichen Standardeinstellungen ändern zu müssen.

Die folgenden Einstellungen können überschrieben werden:

- *Filterung ungültiger Objekte*: Im Gegensatz zu den bestehenden parametrischen Einstellungen gilt die hier festgelegte Behandlungsmethode für **alle Eingaben** des Algorithmus.
- *Berechnungseinstellungen*, wie z.B. *Abstandseinheiten* und *Flächeneinheiten*, die für Entfernungs-/Flächenmessungen verwendet werden
- *Umgebungseinstellungen*, wie *Temporärerverzeichnis* und *Zu verwendende Threadanzahl*

Anmerkung zum Thema Projektionen

Die Ausführung der Verarbeitungsalgorithmen erfolgt immer im Koordinatenbezugssystem (KBS) des Input-Layers. Aufgrund der on-the-fly Reprojektion von QGIS kann es vorkommen, dass sich zwei Layer scheinbar überlappen und übereinstimmen, was aber nicht der Fall ist, wenn die Originalkoordinaten verwendet werden, ohne sie auf ein gemeinsames Koordinatensystem zu projizieren. Wann immer Sie mehr als einen Layer als Eingabe für einen *nativen QGIS Algorithmus* verwenden, egal ob Vektor oder Raster, werden die Layer alle neu projiziert, um mit dem Koordinatenreferenzsystem des ersten Eingabe-Layers übereinzustimmen.

Dies gilt jedoch weniger für die meisten Algorithmen anderer Entwickler, die in der Verarbeitungs-Umgebung verfügbar sind. Diese gehen größtenteils davon aus, dass sich alle Layer bereits in einem gemeinsamen Koordinatensystem befinden und zur Analyse bereit sind.

Standardmäßig wird im Parameterdialog eine Beschreibung des KBS jedes Layers zusammen mit seinem Namen angezeigt, was die Auswahl von Layern, die das gleiche KBS haben, für die Verwendung als Eingabe-Layer erleichtert. Wenn Sie diese zusätzlichen Informationen nicht sehen möchten, können Sie diese Funktion im Dialogfeld Verarbeitungseinstellungen deaktivieren, indem Sie die Option *Layer-KBS-Definition in Auswahlfeldern anzeigen* abwählen.

Wenn Sie versuchen, einen Algorithmus auszuführen, der als Eingabe zwei oder mehr Layer mit nicht übereinstimmenden KBS verwendet, wird ein Warndialog angezeigt. Dies geschieht dank der Option *Warnung vor der Ausführung, wenn die KBS der Layer nicht übereinstimmen*.

Sie können den Algorithmus immer noch ausführen, seien Sie sich aber dessen bewusst, dass dies in den meisten Fällen zu falschen Ergebnissen führt, so z.B. leere Layer, da sich die Eingabe-Layer nicht überlappen.

> **Tipp**
>
> **Verwenden Sie Verarbeitungsalgorithmen um die Projektion anzugleichen**
>
> Wenn ein Algorithmus aufgrund von nicht übereinstimmenden KBS nicht erfolgreich auf mehreren Eingabe-Layern ausgeführt werden kann, verwenden Sie den QGIS-internen Algorithmus *Reproject layer*, um die Layer auf das gleiche KBS zu projizieren, bevor Sie den Algorithmus mit diesen Ausgaben ausführen.

23.3.2 Von Algorithmen erstellte Datenobjekte

Von einem Algorithmus generierte Datenobjekte können unterschiedlicher Art sein. Es kann sich um eine Layer (Vektor mit oder ohne Geometrie, Raster, Netz, Punktwolke usw.) handeln, die als einfache Datei auf der Festplatte oder in einer Datenbank gespeichert ist, um textbasierte Dateien wie `qml`-Stildateien oder Grafikdateien `.html`, um einen Ordner usw.

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.

Der Parameter **output** zum Festlegen des Ergebnisses der Algorithmusausführung bietet ein Textfeld zum Eingeben des Pfads zum Ausgabenergebnis und eine Dropdown-Schaltfläche … für zusätzliche Optionen (die Verfügbarkeit hängt vom ausgeführten Algorithmus ab):

- *Skip Output*: if you are not interested in a given output of the algorithm
- *Temporären Layer erstellen* (`TEMPORARY_OUTPUT`): Die Ausgabe wird in einem Vektor gespeichert *temporärer Layer*.
- *In Datei speichern…*: Es wird ein Dialogfeld zum Speichern der Datei angezeigt, in dem Sie den gewünschten Dateipfad auswählen können. Die unterstützten Dateierweiterungen werden je nach Art der Ausgabe und des Algorithmus in der Dateiformatauswahl des Dialogfelds angezeigt.

Das Format der Ausgabe wird durch die Dateinamenerweiterung definiert. Die unterstützten Formate hängen davon ab, was vom Algorithmus selbst unterstützt wird. Um ein Format auszuwählen, wählen Sie einfach die entsprechende Dateierweiterung aus (oder fügen Sie sie hinzu, wenn Sie stattdessen direkt den Dateipfad eingeben). Wenn die Erweiterung des eingegebenen Dateipfads keinem der unterstützten Formate entspricht, wird eine Standarderweiterung an den Dateipfad angehängt, und das dieser Erweiterung entsprechende Dateiformat wird zum Speichern des Layers oder der Tabelle verwendet. Die Standard-Erweiterungen sind `.dbf` für Tabellen, `.tif` für Raster-Layer und `.gpkg` für Vektor-Layer. Diese können im Einstellungsdialog geändert werden, indem man ein anderes von QGIS unterstütztes Format auswählt.

Sie können einen Standardordner für Ausgabedatenobjekte festlegen. Gehen Sie in den Einstellungsdialog (Sie können ihn über das Menü *Einstellungen* ► *Optionen* ► *Verarbeitung* öffnen), und in der Gruppe *Allgemein* finden Sie einen Parameter namens *Ausgabeordner*. Dieser Ausgabeordner wird als Standardpfad verwendet, wenn Sie bei der Ausführung eines Algorithmus nur einen Dateinamen ohne Pfad eingeben (z. B. `myfile.shp`).

- *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*.
- *In Geopackage speichern…*: Der Ausgabelayer wird als Tabelle einer neuen oder bestehenden GeoPackage-Datenbank gespeichert.
- *In Datenbanktabelle speichern…*: Die Ausgabeschicht wird als neue Tabelle in einer Datenbank wie GeoPackage, SpatiaLite, PostgreSQL oder MS SQL Server gespeichert. Eine Verbindung sollte bereits bestehen. Sie müssen den Namen der neuen Tabelle angeben und je nach Art der Zieldatenbank ein Schema auswählen.
- *An Layer anhängen…*: Fügt die vom Algorithmus ausgegebenen Objekte zu einem bestehenden Layer hinzu. Nach Auswahl des Ziel-Layers öffnet sich ein Feldzuordnungsfenster, in dem manuell festgelegt werden kann, wie die Felder aus dem Ausgab-Layer den Feldern des Ziel-Layers zugeordnet werden sollen.

- *In Verzeichnis speichern*: Verwenden Sie ein neues oder vorhandenes Verzeichnis zum Speichern der Algorithmusausgaben.
- *In einem temporären Verzeichnis speichern*: Erstellt ein Verzeichnis innerhalb des festgelegten Ordners `TEMP` zum Speichern der Algorithmusausgaben. Die Dateien werden gelöscht, sobald Sie QGIS beenden.

Wenn Sie einen Algorithmus ausführen, der einen Vektorlayer im iterativen Modus verwendet, wird der eingegebene Dateipfad als Basispfad für alle erstellten Dateien, die mithilfe des Basisnamens benannt werden und an die eine Nummer, die den Index der Iteration darstellt, angehängt wird, verwendet. Die Dateierweiterung (und das Format) wird für alle so erstellten Dateien benutzt.

Algorithmen generieren auch Grafiken und Text als HTML-Dateien. Diese Ergebnisse werden am Ende der Algorithmusausführung im Dialogfeld *Ergebnisanzeige* angezeigt. Dieses Dialogfeld speichert die von allen Algorithmen während der aktuellen Sitzung erzeugten Ergebnisse und kann jederzeit durch Auswahl von *Verarbeitung* ► *Ergebnisanzeige* im QGIS-Hauptmenü angezeigt werden.

Bei layerbasierten Ausgaben ist es möglich, *Ausgabedatei nach Ausführung des Algorithmus öffnen* direkt in QGIS als neuen Eintrag im *Layers*-Panel zu aktivieren. Standardmäßig werden die Dateien geöffnet.

23.4 Das Protokoll

23.4.1 Das Verarbeitung Protokoll

Jedes Mal, wenn Sie einen Algorithmus ausführen, werden Informationen über den Prozess im Verlaufsmanager gespeichert. Datum und Uhrzeit der Ausführung werden zusammen mit den verwendeten Parametern gespeichert, was es einfach macht, alle Arbeiten, die mit dem Verarbeitungsframework entwickelt wurden, zu verfolgen, zu kontrollieren und zu reproduzieren.

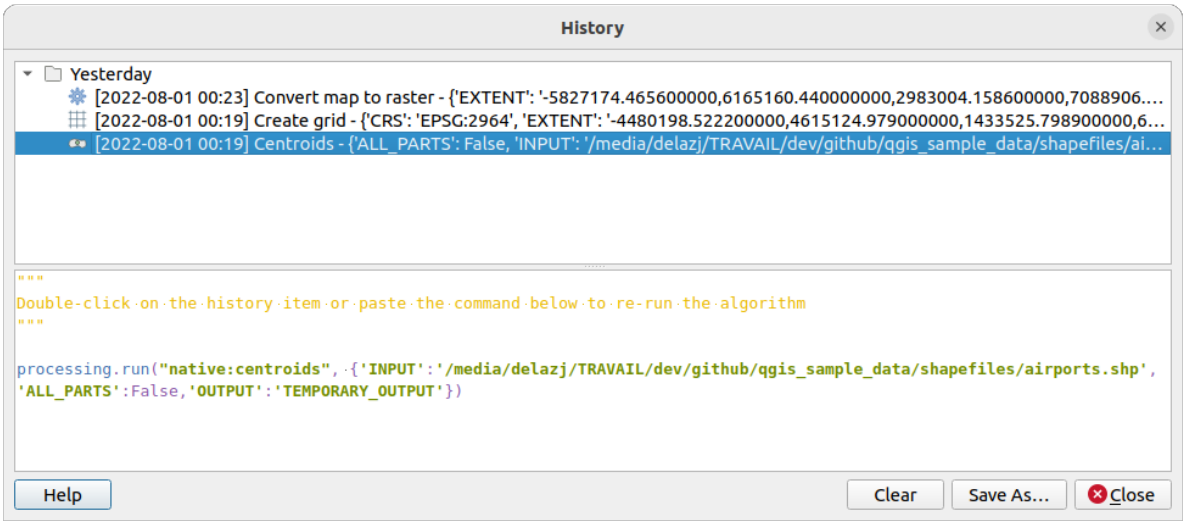


Abb. 23.17: Protokoll

Prozessinformationen werden als Befehlszeilenausdruck gespeichert, selbst wenn der Algorithmus aus der Toolbox gestartet wurde. Dies macht es nützlich für diejenigen, die lernen möchten, wie man die Befehlszeilenschnittstelle verwendet, da sie einen Algorithmus mit der Toolbox aufrufen und dann das Protokoll nutzen können, um zu sehen, wie er von der Befehlszeile aus aufgerufen werden könnte.

Mit einem Rechts-Klick auf einen Eintrag können Sie:

- *Als Python-Befehl kopieren*: ermöglicht das einfache Kopieren des entsprechenden *PyQGIS-Befehls* aus dem Dialog. Wie der Code, der unterhalb der Befehlsliste angezeigt wird.
- *Als qgis_process-Befehl kopieren*: ermöglicht die einfache Generierung von *qgis_process Befehlen*, einschließlich der Umgebungseinstellungen wie Entfernungseinheiten, Flächeneinheiten, Ellipsoid, und alle komplexen Parameterwerte wie GeoPackage Ausgaben mit bestimmten Layern
- *Als JSON kopieren*: alle Einstellungen des Befehls werden in ein `JSON` Format kopiert, bereit von qgis_process verarbeitet zu werden. Dies ist ein bequemer Weg, um das erwartete Format der Befehle zu sehen, sogar für komplexe Parameter (wie TIN Interpolationsparameter). Sie können diese einfach speichern und später wiederherstellen, indem Sie die Werte über einen Algorithmus-Dialog einfügen.
- *Test erzeugen…* unter Verwendung des betreffenden Algorithmus und der Parameter, gemäß den Anweisungen in der Processing README-Datei.

Neben dem Durchsuchen der Einträge in der Registrierung können Sie Prozesse auch erneut ausführen, indem Sie einfach auf den Eintrag doppelklicken. Das Algorithmus-Dialogfeld wird dann mit bereits festgelegten Parametern geöffnet, und Sie können jeden davon an Ihre Bedürfnisse anpassen und den Algorithmus erneut ausführen.

Der *Protokoll*-Dialog bietet auch eine bequeme Möglichkeit, zur Konsolidierung der Testinfrastruktur von QGIS Processing Algorithmen und Skripten beizutragen.

23.4.2 Das Verarbeitung Log

Der Verlaufsdialog enthält nur die Ausführungsaufrufe, aber nicht die Informationen, die der Algorithmus bei der Ausführung erzeugt. Diese Informationen werden in das QGIS-Log geschrieben (*Ansicht* ► *Bedienfelder* ► *Protokoll*).

Algorithmen von Drittanbietern werden normalerweise mithilfe ihrer Befehlszeilenschnittstellen ausgeführt, die über die Konsole mit dem Benutzer kommunizieren. Obwohl diese Konsole nicht angezeigt wird, wird normalerweise jedes Mal, wenn Sie einen dieser Algorithmen ausführen, ein vollständiger Verlauf davon in das Protokoll geschrieben. Um das Protokoll nicht mit diesen Informationen zu überladen, können Sie es für jeden Anbieter im Einstellungsdialog deaktivieren.

Einige Algorithmen geben, auch wenn sie mit den gegebenen Eingabedaten ein Ergebnis liefern können, Kommentare oder zusätzliche Informationen zum Protokollieren aus, wenn sie potenzielle Probleme mit den Daten erkannt haben, um Sie zu warnen. Stellen Sie sicher, dass Sie diese Meldungen im Protokoll überprüfen, wenn Sie unerwartete Ergebnisse erhalten.

23.5 Ein Modell erstellen

Mit dem *Modellentwurf* können Sie komplexe Modelle über eine einfache und leicht zu bedienende Schnittstelle erstellen. Bei der Arbeit mit einem GIS sind die meisten Analysevorgänge nicht isoliert, sondern Teil einer Kette von Vorgängen. Mit dem Modellentwurf kann diese Kette von Operationen in einen einzigen Prozess verpackt werden, sodass sie zu einem späteren Zeitpunkt mit einem anderen Satz von Eingaben bequem ausgeführt werden kann. Unabhängig von der Anzahl der Schritte und der verschiedenen Algorithmen wird ein Modell als ein einziger Algorithmus ausgeführt, was Zeit und Mühe spart.

Der Modellentwurf kann aus der Menüleiste über (*Verarbeitung* ► *Modellentwurf*) geöffnet werden.

23.5.1 Die Benutzeroberfläche des Modellentwurfs

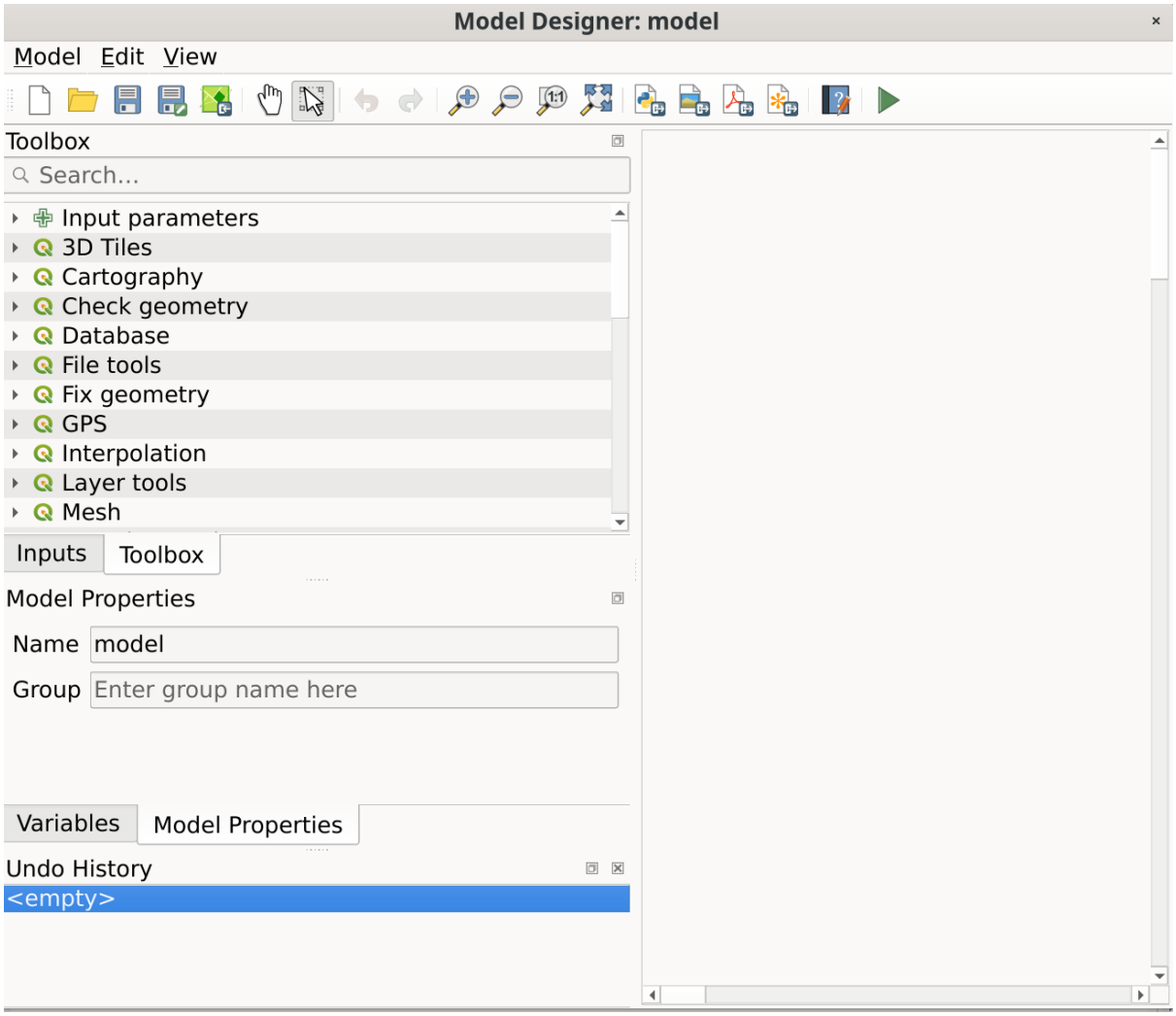


Abb. 23.18: Modellentwurf

Auf der rechten Seite der Benutzeroberfläche befindet sich ein Bereich, auf dem die Struktur des Modells und der Arbeitsablauf, den es darstellt, aufgebaut werden können.

Im oberen Bereich der Benutzeroberfläche bieten verschiedene Menüs und die *Navigation*-Werkzeugleiste Zugang zu einer Vielzahl von Werkzeugen.

Menü Modell

Bezeichnung	Tastenkürz	Navigation Toolbar	Beschreibung
Modell überprüfen			Prüft, ob die im Modell verwendeten Algorithmen und Eingaben existieren. Praktisch vor der Freigabe eines Modells.
Modell ausführen…	F5	☑	Führt das Modell aus
Ausgewählte Schritte ausführen…	Umschalt		Führt nur die ausgewählten Schritte in einem Modell aus, sodass ein Teil des Modells ausgeführt werden kann.
Modelleingaben neu anordnen…			Legt die Reihenfolge fest, in der die Eingaben dem Benutzer im Algorithmus-Dialog präsentiert werden.
Ausgabelayer neu anordnen…			Legt eine bestimmte Reihenfolge fest, die die Ausgaben des Modells beim Laden der Ergebnisse in ein Projekt verwenden müssen.
Modell öffnen…	Strg+O	☑	Öffnet eine `.model3`-Datei zur Bearbeitung oder Ausführung
Modell speichern	Strg+S	☑	Speichert das Modell auf der Festplatte als `.model3`-Datei
Modell speichern als…	Strg+Shi	☑	Speichert das Modell auf der Festplatte als neue `.model3`-Datei
Modell in Projekt speichern		☑	Bettet die Modelldatei in die Projektdatei ein und macht sie bei der Weitergabe der Projektdatei verfügbar.
Modelhilfe bearbeiten…		☑	Eine Schnittstelle zur Dokumentation des Modells, der Algorithmen, der Parameter und Ausgaben sowie des Autors und der Versionierung
Exportieren ►			
► *Als Bild exportieren…*		☑	Speichert den grafischen Entwurf des Modells in einem Bilddateiformat (zu Illustrationszwecken)
► *Als PDF exportieren…*			Speichert den grafischen Entwurf des Modells in einem `PDF`-Dateiformat (zu Illustrationszwecken)
► *Als SVG exportieren…*			Speichert den grafischen Entwurf des Modells in einem `SVG`-Dateiformat (zu Illustrationszwecken)
► *Als Python-Skript exportieren…*		☑	Erzeugt eine Python-Skriptdatei mit den Anweisungen für das Modell

Bearbeiten-Menü

Bezeichnung	Tastenkürz	Navigation Toolbar	Beschreibung
Alles wählen	Strg+A		Wählt alle Modellkomponenten im Designer aus
Gewählte Komponenten auf Gitter einrasten			rastet die Elemente ein und richtet sie in einem Raster aus
Wiederholen	Strg+Y	☑	Rückgängig machen der zuletzt abgebrochenen Aktion. Siehe auch das *Undo/Redo* Bedienfeld.
Rückgängig	Strg+Z	☑	Macht die vorherige Änderung rückgängig. Siehe auch das *Undo/Redo* Bedienfeld.
Ausschneiden	Strg+X		Schneidet eine Auswahl von Komponenten aus dem Modell aus.
Kopieren	Strg+C		Kopiert eine Auswahl von Komponenten aus dem Modell.
Einfügen	Strg+V		Fügt eine ausgeschnittene oder kopierte Auswahl von Komponenten aus einem Modell in ein anderes oder innerhalb desselben Modells ein. Die ausgewählten Komponenten behalten ihre ursprünglichen Eigenschaften und Kommentare bei.
Gewählte Komponenten löschen	Entf		Entfernt eine Komponente aus dem Modell.
Gruppenrahmen hinzufügen			Fügt einen Rahmen im Hintergrund zusammengehöriger Komponenten hinzu, um diese visuell zu gruppieren. Besonders nützlich bei großen Modellen, um den Arbeitsablauf übersichtlich zu halten.

Menü Ansicht

Bezeichnung	Tastenkürz	Navigation Toolbar	Beschreibung
Zoomen auf ►			Zoomt auf den Umfang des ausgewählten Gruppenrahmens
Hineinzoomen	Strg++	☑	
Hinaus zoomen	Strg+-	☑	
Zoomen auf 100%	Strg+1	☑	
Volle Ausdehnung	Strg+0	☑	Zeigt alle Komponenten im Darstellungsbereich des Designers an
Kommentare anzeigen			Zeigt Kommentare zu jedem Algorithmus oder jeder Eingabe im Modelldesigner an
Einrasten aktivieren			
Bedienfeldsichtbarkeit umschalten	Strg+Tab		Schaltet die *Bedienfelder* im Designer ein oder aus

Bedienfelder

Der linke Teil des Fensters ist ein Bereich mit fünf Bedienfeldern, mit denen Sie dem Modell neue Elemente hinzufügen können:

1. *Eingaben*: alle *Eingabeparameter*, die Ihr Modell beeinflussen könnten
2. *Toolbox*: here you can find available *Processing algorithms*. It also includes input parameters. Use *Search…* box to find algorithms or inputs.
3. *Modell-Eigenschaften*: geben Sie den Namen (erforderlich) des Modells und die Gruppe an, in der es in der *Verarbeitungs-Werkzeugkiste* angezeigt wird
4. *Variablen*: Modelle können spezielle *Variablen* enthalten, die eindeutig und nur für sie verfügbar sind. Auf diese Variablen kann mit jedem im Modell verwendeten Ausdruck zugegriffen werden. Sie sind nützlich, um Algorithmen innerhalb eines Modells zu steuern und mehrere Aspekte des Modells durch Änderung einer einzigen Variable zu kontrollieren. Die Variablen können im Bedienfeld *Variablen* eingesehen und geändert werden.
5. *Rücknahmeprotokoll*: Dieses Bedienfeld registriert alles, was im Modellierer passiert, und macht es einfach, Dinge rückgängig zu machen, die man falsch gemacht hat.

Zu den verfügbaren Algorithmen

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 Ein Modell erstellen

Das Erstellen eines Modells beinhaltet zwei grundlegende Schritte:

1. *Definition von erforderlichen Eingaben*. Diese Eingaben werden dem Paramterfenster hinzugefügt, sodass der Anwender Ihre Werte einstellen kann, wenn er das Modell ausführt. Das Modell selber ist ein Algorithmus, also wird das Parameterfenster automatisch erstellt, so wie es mit allen Algorithmen, die in der Verarbeitungsumgebung zur Verfügung stehen, passiert.
2. *Definition des Arbeitsablaufs*. Der Arbeitsablauf wird definiert, indem Algorithmen hinzugefügt werden. In diesen wird festgelegt, wie sie die definierten Eingaben oder die von anderen Algorithmen im Modell erzeugten Ausgaben verwenden.

Definition von Eingaben

Der erste Schritt besteht darin, die Eingaben für das Modell zu definieren. Sie befinden sich im Bedienfeld *Eingaben* auf der linken Seite des Modellentwurfsfensters. Wenn Sie mit dem Mauszeiger über die Eingaben fahren, wird ein Tooltip mit zusätzlichen Informationen angezeigt. Eine vollständige Liste der im Modellentwurf verfügbaren Parameter und ihrer Entsprechung für die Skripterstellung finden Sie unter *Eingabe- und Ausgabetypen für Verarbeitungsalgorithmen*.

Wenn Sie auf ein Element doppelklicken, wird ein Dialog angezeigt, in dem Sie seine Eigenschaften definieren können. Je nach Parameter enthält der Dialog mindestens ein Grundelement (die Beschreibung, die der Benutzer bei der Ausführung des Modells sieht). Die Angabe einer Beschreibung ist Pflicht. Wenn Sie einen numerischen Wert hinzufügen, wie in der nächsten Abbildung zu sehen ist, müssen Sie, zusätzlich zur Beschreibung des Parameters, einen Standardwert und den Bereich der gültigen Werte festlegen.

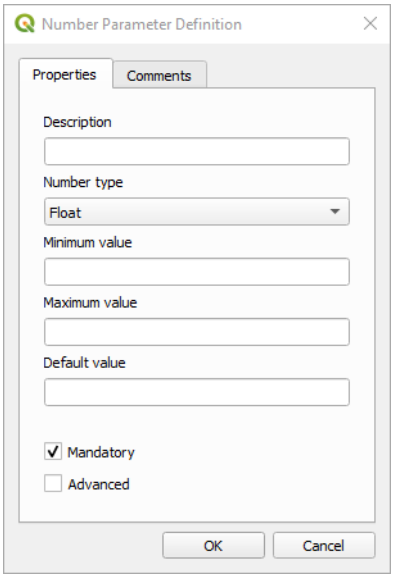


Abb. 23.19: Definition der Modellparameter

Sie können Ihre Eingabe als zwingend für Ihr Modell definieren, indem Sie das Kontrollkästchen Zwingend erforderlich ankreuzen. Durch Ankreuzen des Kontrollkästchens Erweitert können Sie die Eingabe so einstellen, dass sie sich im Abschnitt Erweitert befindet. Dies ist besonders nützlich, wenn das Modell viele Parameter hat und einige davon nicht trivial sind, Sie diese aber dennoch wählen möchten.

Für jede hinzugefügte Eingabe wird der Modellierungsebene ein neues Element hinzugefügt.

Abb. 23.20: Darstellungen von Modellparameter

Sie können Eingaben auch hinzufügen, indem Sie den Eingabetyp aus der Liste an die gewünschten Stelle im Modellierungsfenster ziehen. Wenn Sie einen Parameter einer bestehenden Eingabe ändern möchten, doppelklicken Sie einfach darauf, und es erscheint der gleiche Dialog wie bei der Erstellung.

Wenn Sie ein Modell innerhalb eines anderen Modells verwenden, werden die erforderlichen Ein- und Ausgaben in der Modellansicht angezeigt.

Definition des Arbeitsablaufs

Im folgenden Beispiel werden wir zwei Eingaben und zwei Algorithmen hinzufügen. Ziel des Modells ist es, die Höhenwerte eines DEM-Raster-Layers mit Hilfe des Algorithmus `Drape` in einen Layer mit Linien zu kopieren und dann die Gesamtsteigung des Layers mit Hilfe des Algorithmus `Climb Along Line` zu berechnen.

Auf der Registerkarte *Eingaben* wählen Sie die beiden Eingaben als `Vektorlayer` für die Linie und `Rasterlayer` für das DEM. Wir sind nun bereit, die Algorithmen zum Workflow hinzuzufügen.

Algorithms can be found in the *Toolbox* panel, grouped much in the same way as they are in the Processing toolbox.

Abb. 23.21: Modell-Eingaben

Um einen Algorithmus zu einem Modell hinzuzufügen, doppelklicken Sie auf seinen Namen oder ziehen Sie ihn per Drag & Drop, genau wie bei den Eingaben. Wie bei den Eingaben können Sie die Beschreibung des Algorithmus ändern und einen Kommentar hinzufügen. Beim Hinzufügen eines Algorithmus erscheint ein Ausführungsdialog, dessen Inhalt dem des Bedienfelds ähnelt, das beim Ausführen des Algorithmus aus der Toolbox angezeigt wird. Die folgende Abbildung zeigt die beiden Algorithmus-Dialoge `Drapieren (Z-Wert von Raster übernehmen)` und `Entlang Linie klettern`.

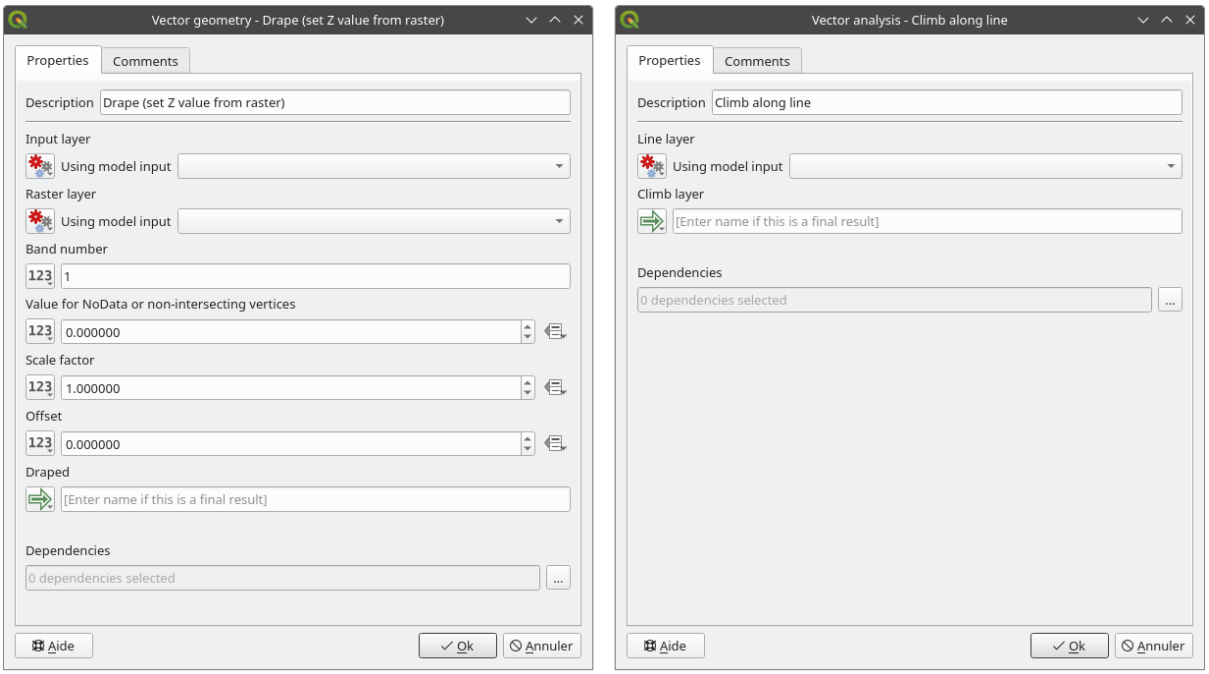


Abb. 23.22: Modell-Algorithmus-Parameter

Wie Sie sehen können, gibt es jedoch einige Unterschiede. Neben jedem Parameter befindet sich ein Dropdown-Menü, mit dem Sie festlegen können, wie der Parameter während des Arbeitsablaufs verwendet werden soll:

- 123 Wert: erlaubt es Ihnen, dem Parameter einen statischen Wert zuzuweisen. Abhängig vom Parametertyp können Sie mit dem Widget eine Zahl (`5,0`), eine Zeichenkette (`meinText`) eingeben, im QGIS-Projekt oder aus einem Ordner geladene Layer auswählen, Elemente aus einer Liste auswählen, …
- Vorberechneter Wert: öffnet das Dialogfeld *Ausdruckseditor* und ermöglicht es Ihnen, einen Ausdruck zu definieren, um den Parameter zu füllen. Modelleingaben sowie einige andere Layer-Statistiken stehen als **Variablen** zur Verfügung und werden oben im Dialogfeld „Suche“ des Ausdrucksditors aufgelistet. Der Ausdruck wird einmal ausgewertet, bevor der untergeordnete Algorithmus ausgeführt wird, und während der Ausführung dieses Algorithmus verwendet.
- Modell Eingabe: ermöglicht die Verwendung einer Eingabe, die dem Modell als Parameter hinzugefügt wurde. Wenn Sie auf diese Option klicken, werden alle geeigneten Eingaben für den Parameter aufgelistet.
- Algorithmus Ausgabe: erlaubt es, die Ausgabe eines anderen Algorithmus als Eingabe des aktuellen Algorithmus zu verwenden. Wie bei den Modelleingaben listet diese Option alle geeigneten Eingaben für den Parameter auf.
- Der **Ausgabeparameter** verfügt ebenfalls über die oben genannten Optionen in seinem Dropdown-Menü:
  - Hinzufügen statischer Ausgaben für untergeordnete Algorithmen, z. B. Speichern der Ausgabe eines untergeordneten Algorithmus in einem vordefinierten Geopaket oder Postgres Layer
  - Verwendung ausdrucksbasierter Ausgabewerte für untergeordnete Algorithmen, z. B. automatische Generierung eines Dateinamens auf der Grundlage des heutigen Datums und Speichern der Ausgaben in dieser Datei

- eine Modelleingabe verwenden, z. B. die Modelleingabe *Datei/Ordner*, um eine Ausgabedatei oder einen Ordner anzugeben
- eine andere Algorithmusausgabe verwenden, z. B. die Ausgabe des Algorithmus *Verzeichnis erstellen* (aus *Modellierwerkzeuge*)
- eine zusätzliche Option Modell Ausgabe macht die Ausgabe des Algorithmus im Modell verfügbar. Wenn ein vom Algorithmus erzeugter Layer nur als Eingabe für einen anderen Algorithmus verwendet werden soll, brauchen Sie dieses Textfeld nicht zu bearbeiten.

In der folgenden Abbildung sehen Sie die beiden als `Modelleingabe` definierten Eingabeparameter und den temporären Output Layer:

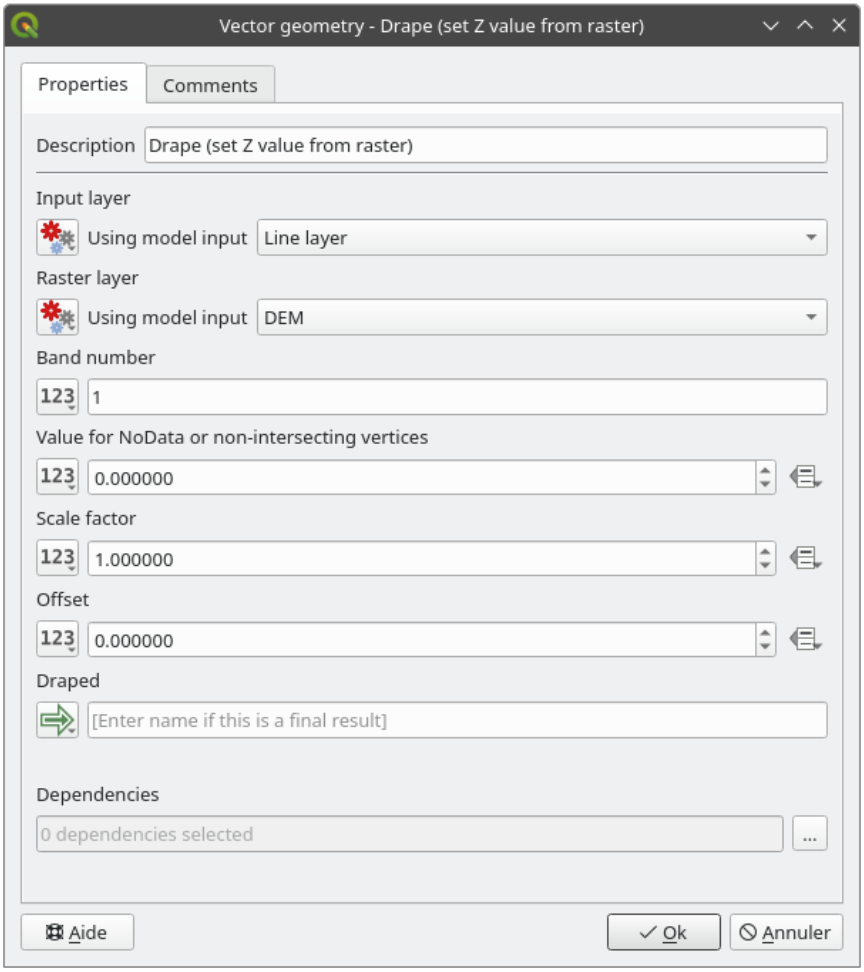


Abb. 23.23: Algorithmus Eingabe- und Ausgabeparameter

Sie finden auch einen zusätzlichen Parameter namens *Abhängigkeiten*, der beim Aufruf des Algorithmus aus der Toolbox nicht verfügbar ist. Mit diesem Parameter können Sie die Reihenfolge festlegen, in der die Algorithmen ausgeführt werden, indem Sie einen Algorithmus ausdrücklich als *Elternteil* des aktuellen Algorithmus definieren. Dadurch wird der *übergeordnete* Algorithmus vor dem aktuellen Algorithmus ausgeführt.

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.

Sobald allen Parametern gültige Werte zugewiesen wurden, klicken Sie auf *OK* und der Algorithmus wird der

Oberfläche hinzugefügt. Er wird mit den Elementen auf der Oberfläche (Algorithmen oder Eingaben) verknüpft, die Objekte liefern, die als Eingaben für den Algorithmus verwendet werden.

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.

Die Verknüpfungen zwischen den Elementen werden automatisch aktualisiert, und am oberen und unteren Rand jedes Algorithmus befindet sich eine Schaltfläche +. Wenn Sie auf die Schaltfläche klicken, werden alle Eingänge und Ausgänge des Algorithmus aufgelistet, so dass Sie sich einen schnellen Überblick verschaffen können.

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.

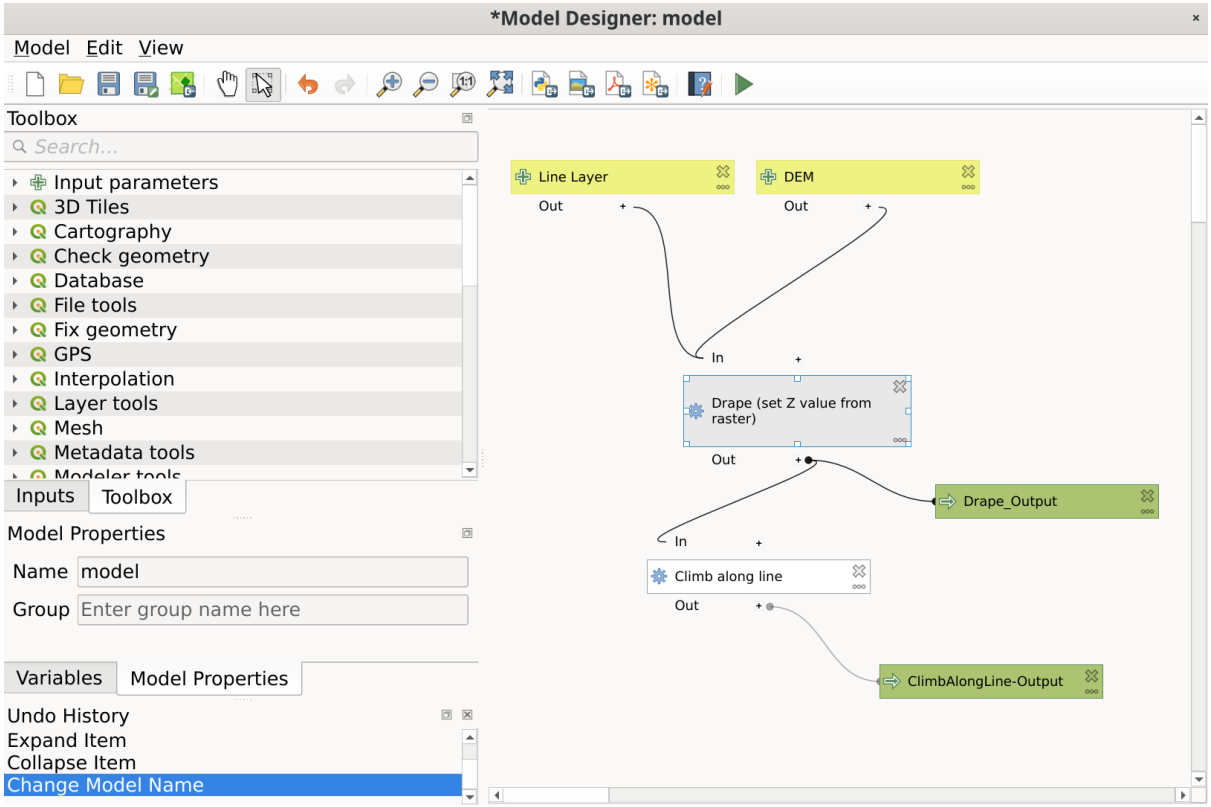


Abb. 23.24: Ein komplettes Modell

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:

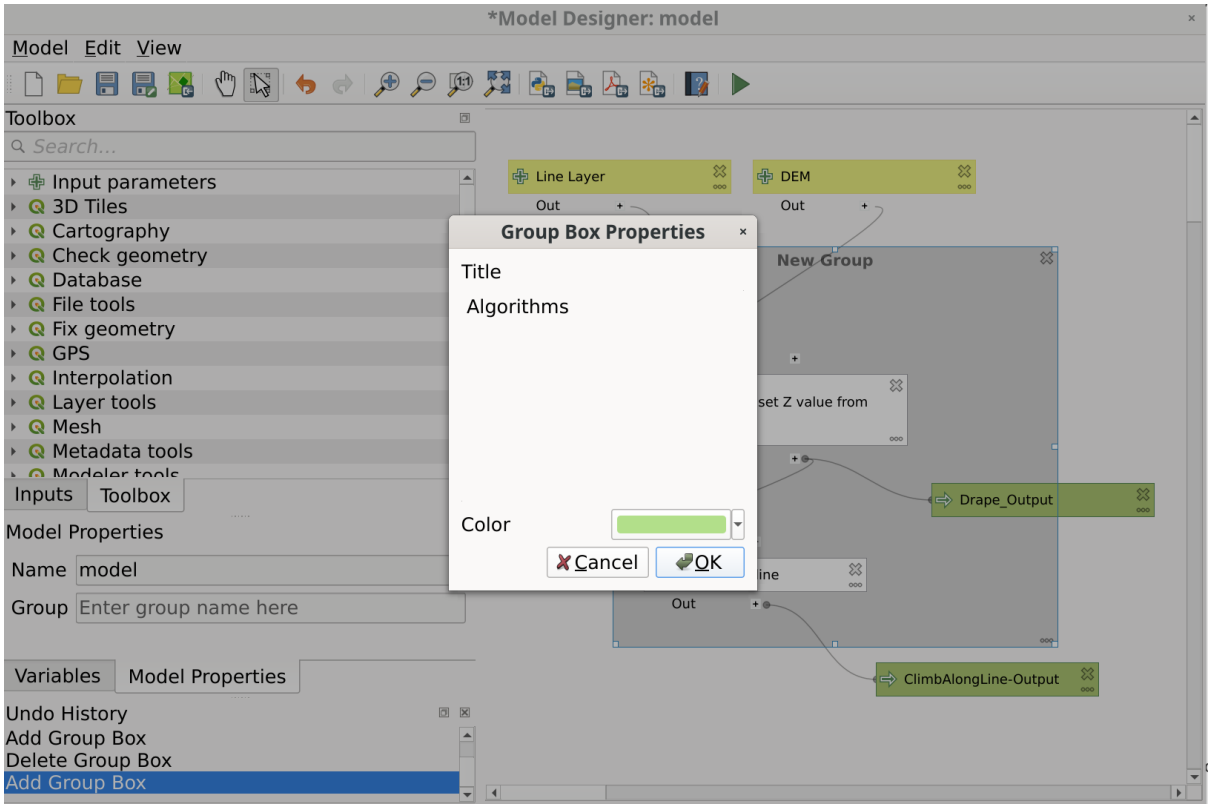


Abb. 23.25: Modell Gruppenrahmen

Sie können den Namen und die Farbe der Boxen ändern. Gruppenrahmen sind sehr nützlich, wenn sie zusammen mit dem Werkzeug *Ansicht* ► *Zoomen auf* ► verwendet werden, mit dem Sie auf einen bestimmten Teil des Modells zoomen können. Sie können auch mit dem Mausrad hinein- und herauszoomen.

Sie können die Reihenfolge der Eingaben und deren Auflistung im Hauptdialog des Modells ändern. Am unteren Ende des Bedienfeldes `Eingaben` finden Sie die Schaltfläche `Modell-Eingaben neu ordnen`. Wenn Sie darauf klicken, öffnet sich ein neuer Dialog, in dem Sie die Reihenfolge der Eingaben ändern können:

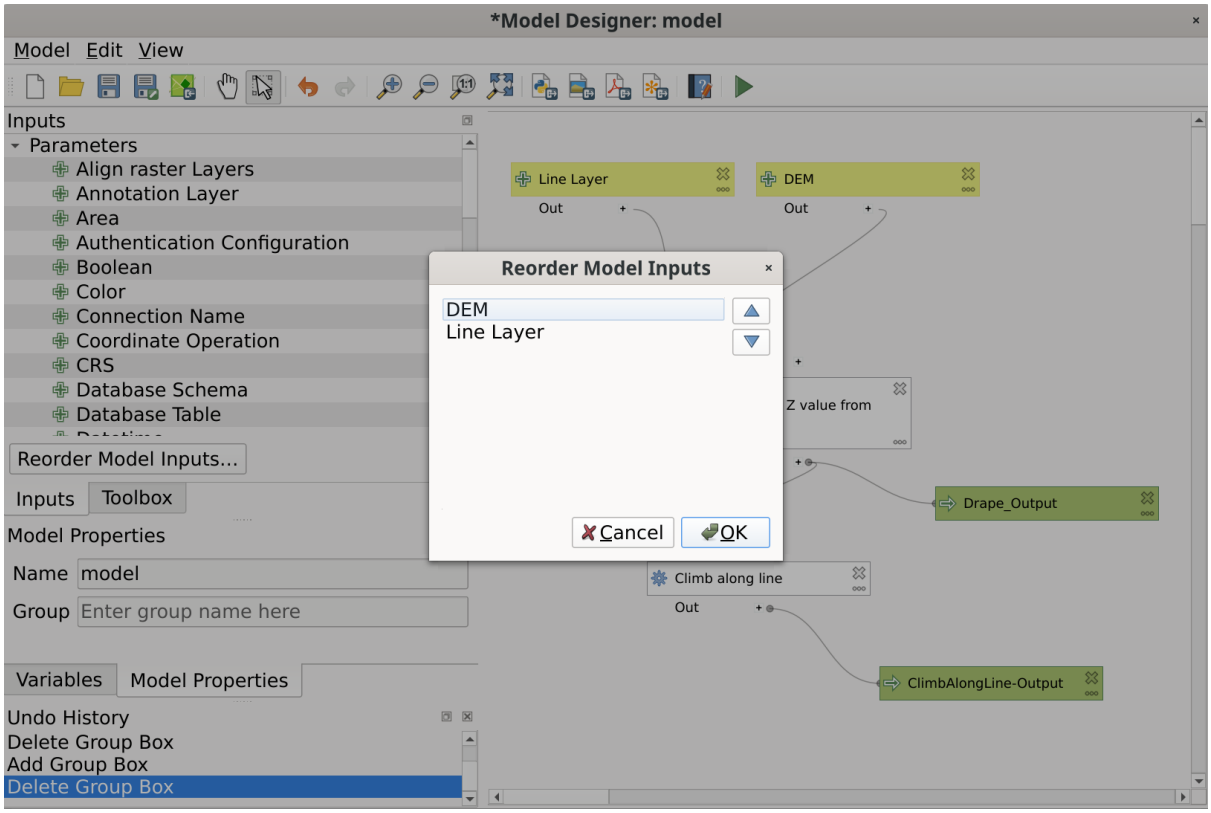


Abb. 23.26: Modell-Eingaben neu ordnen

Es besteht auch die Möglichkeit, eine bestimmte Reihenfolge festzulegen, die die Ausgaben des Modells beim Laden der Ergebnisse in ein Projekt verwenden müssen. Dadurch kann der Modellersteller sicherstellen, dass die Layer in der Kartenansicht logisch angeordnet sind, wenn er ein Modell ausführt, z. B. indem er einen Vektorlayer über einen Rasterlayer oder einen Punktelayer über einen Polygonlayer legt. Der Modellersteller kann auch einen optionalen „Gruppennamen“ für die Ausgaben festlegen, um die Ausgaben innerhalb des Layer-Baums automatisch zu gruppieren, indem er einen neuen Gruppennamen verwendet oder sie einer bestehenden Gruppe hinzufügt. Im `Modell`-Menü finden Sie den Eintrag `Outputlayern neu ordnen ...`. Wenn Sie darauf klicken, öffnet sich ein neues Dialogfenster, in dem Sie die Reihenfolge der Output Layer ändern können:

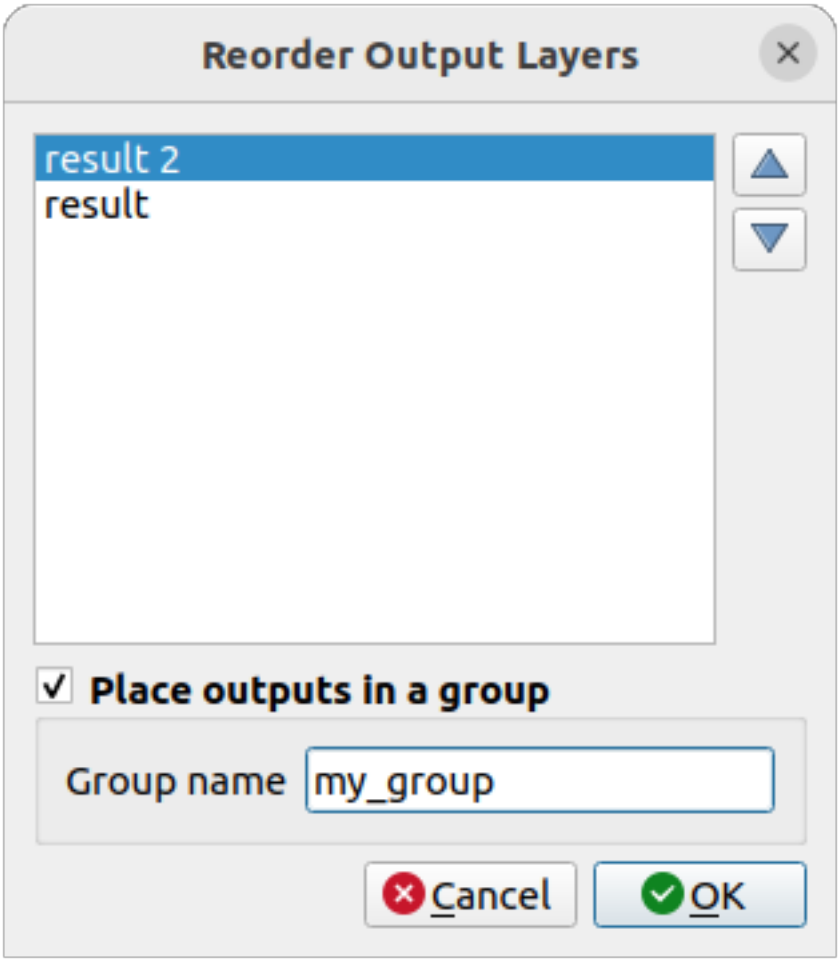


Abb. 23.27: Ausgabelayer neu ordnen

Kommentare können auch zu Eingaben oder Algorithmen im Modellierer hinzugefügt werden. Dies kann über die Registerkarte *Kommentare* des Elements oder durch einen Rechtsklick erfolgen. In der gleichen Registerkarte kann eine Farbe für einzelne Modellkommentare manuell eingestellt werden. Kommentare sind nur in der Kartenansicht des Modellierers sichtbar und nicht im Dialog des endgültigen Algorithmus; sie können durch Deaktivieren von *Ansicht* ► *Kommentare anzeigen* ausgeblendet werden.

Ihr Modell kann auf verschiedene Arten ausgeführt werden:

- Sie können das gesamte Modell ausführen, indem Sie in der Symbolleiste auf Modell ausführen… klicken, mit der rechten Maustaste auf das Modell im *Browser*-Bedienfeld oder *Modell* ► *Modell ausführen…* klicken: Wenn Sie das Modell mit dem Editor ausführen, werden alle nicht standardmäßigen Werte in den Eingaben gespeichert. Das bedeutet, dass bei einer späteren Ausführung des Modells über den Editor diese Werte bei jedem weiteren Lauf bereits im Dialogfeld vorausgefüllt sind.
- Sie können Elemente des Modells auswählen und nur diesen Teil des Modells ausführen: Klicken Sie auf die Option *Ausgewählte Schritte ausführen…* im Menü *Modelle* ► oder im Kontextmenü eines ausgewählten Algorithmus. Der Anfangszustand wird aus früheren Ausführungen des Modells über den Editor übernommen, sodass die Ergebnisse aus früheren Schritten des Modells für die ausgewählten Schritte verfügbar sind. Auf diese Weise können Teile eines großen Modells korrigiert werden, ohne dass das gesamte Modell ständig zum Testen ausgeführt werden muss. Dies ist besonders nützlich, wenn frühere Schritte im Modell zeitaufwändig sind.
- Sie können einen Teil des Modells ausführen, beginnend mit einem bestimmten Algorithmus: Klicken Sie mit der rechten Maustaste auf den Algorithmus und wählen Sie *Von hier aus ausführen…*. Ebenso werden

Werte aus früheren Ausführungen übernommen.

- Um den Algorithmus aus der Processing Toolbox verwenden zu können, muss er gespeichert und der Modeller-Dialog geschlossen werden, damit die Toolbox ihren Inhalt aktualisieren kann.

Ihr Modell dokumentieren

Sie müssen Ihr Modell dokumentieren, und das können Sie direkt im Modellierer tun. Klicken Sie auf die Schaltfläche Modellhilfe bearbeiten, und ein Dialogfeld wie das folgende wird angezeigt.

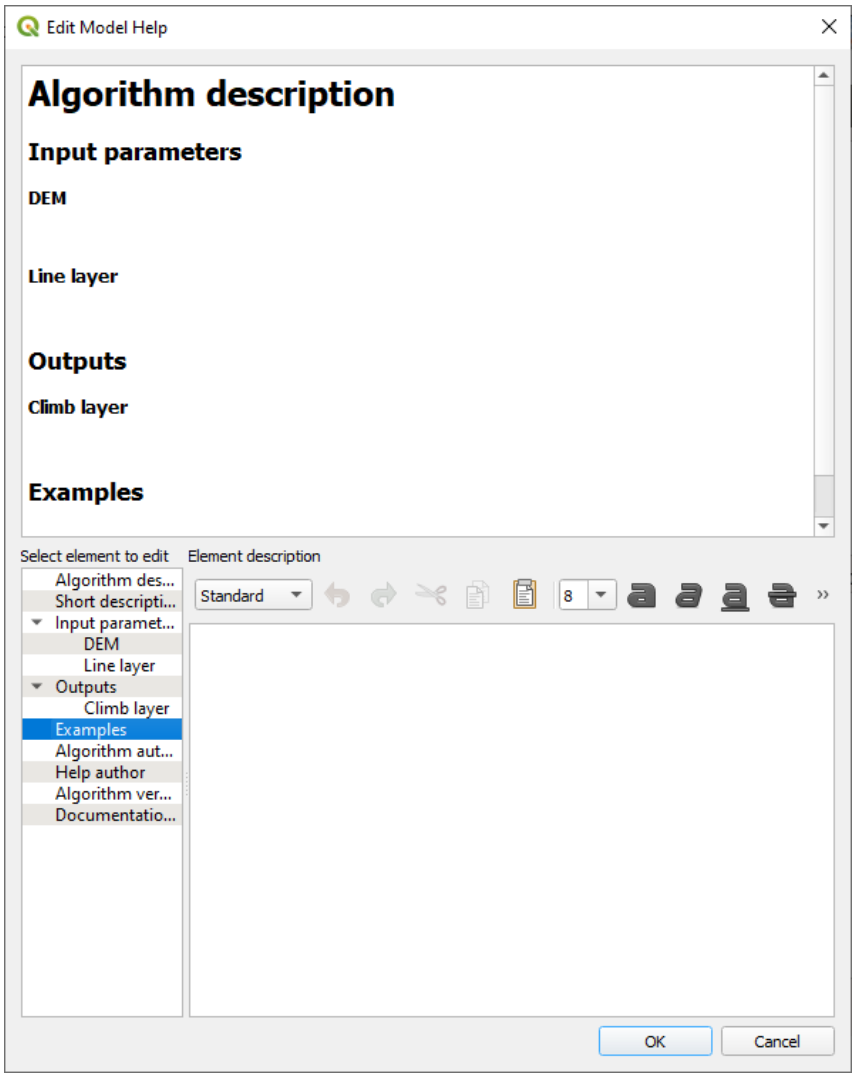


Abb. 23.28: Hilfe bearbeiten

Auf der rechten Seite sehen Sie eine einfache HTML-Seite, die anhand der Beschreibung der Eingabeparameter und der Ergebnisse des Algorithmus erstellt wurde, zusammen mit einigen zusätzlichen Elementen wie einer allgemeinen Beschreibung des Modells oder seines Autors. Außerdem gibt es einen Abschnitt mit Beispielen, in den Sie Ihre eigenen Beispiele eingeben können, um die Verwendung des Modells zu erläutern. Wenn Sie den Hilfe-Editor zum ersten Mal öffnen, sind alle diese Beschreibungen leer, aber Sie können sie mithilfe der Elemente auf der linken Seite des Dialogs bearbeiten. Wählen Sie ein Element im oberen Teil aus und schreiben Sie die Beschreibung in das Textfeld darunter.

Modell-Hilfe wird als Teil eines Modells selbst gespeichert.

23.5.3 Speichern und laden von Modellen

Speicherung von Modellen

Verwenden Sie die Schaltfläche

, um das aktuelle Modell zu speichern und die Schaltfläche

, um ein zuvor gespeichertes Modell zu öffnen. Modelle werden mit der Erweiterung `.model3` gespeichert. Wenn das Modell bereits über das Modellierfenster gespeichert wurde, werden Sie nicht zur Eingabe eines Dateinamens aufgefordert. Da bereits eine Datei mit dem Modell verknüpft ist, wird diese Datei für nachfolgende Speicherungen verwendet.

Bevor Sie ein Modell speichern, müssen Sie in den Textfeldern im oberen Teil des Fensters einen Namen und eine Gruppe für das Modell eingeben.

Modelle, die im Ordner `models` gespeichert sind (Standardordner, wenn Sie zur Eingabe eines Dateinamens zum Speichern des Modells aufgefordert werden), erscheinen in der Toolbox in der entsprechenden Rubrik. Wenn die Werkzeugkiste aufgerufen wird, durchsucht sie den Ordner `models` nach Dateien mit der Erweiterung `.model3` und lädt die darin enthaltenen Modelle. Da ein Modell selbst ein Algorithmus ist, kann es wie jeder andere Algorithmus zur Werkzeugkiste hinzugefügt werden.

Modelle können auch innerhalb des Projekts mit der Schaltfläche

gespeichert werden. Modelle, die mit dieser Methode gespeichert werden, werden nicht als :file:``.model3` Dateien auf die Platte geschrieben, sondern in die Projektdatei eingebettet.

Projektmodelle sind im Menü
Projektmodelle der Toolbox und im Menüpunkt *Projekt* ► *Modelle* verfügbar.

Die Verarbeitungsmodellierung kann im Verarbeitungskonfigurationsdialog in der *Modelle* Gruppe eingestellt werden.

Modelle, die aus dem Ordner `models` geladen wurden, erscheinen nicht nur in der Werkzeugkiste, sondern auch in der Algorithmenliste im Reiter *Algorithms* des Modellierfensters. Das bedeutet, dass Sie ein Modell als Teil eines größeren Modells einbinden können, genau wie andere Algorithmen.

Die Modelle werden im *Browser* Bedienfeld angezeigt und können von dort aus gestartet werden.

Exportieren eines Modells als Python Skript

Wie wir in einem späteren Kapitel sehen werden, können Processing-Algorithmen von der QGIS-Python-Konsole aus aufgerufen werden, und neue Processing-Algorithmen können mit Python erstellt werden. Eine schnelle Möglichkeit, ein solches Python-Skript zu erstellen, besteht darin, ein Modell zu erstellen und es dann als Python-Datei zu exportieren.

Klicken Sie dazu auf die Schaltfläche

in der Kartenansicht des Modells oder klicken Sie mit der rechten Maustaste auf den Namen des Modells in der Werkzeugkiste und wählen Sie

.

Exportieren eines Modells als Bild, PDF oder SVG-Datei

Ein Modell kann auch als Bild, SVG oder PDF (für Illustrationszwecke) exportiert werden, indem Sie auf

,

oder

klicken.

23.5.4 Ein Modell bearbeiten

Sie können das Modell, das Sie gerade erstellen, bearbeiten und dabei den Workflow und die Beziehungen zwischen den Algorithmen und Eingaben, die das Modell definieren, anpassen.

Sie können auch ein vorhandenes Modell bearbeiten, indem Sie im *Browser-Panel* die Option *Modell bearbeiten…* auswählen.

Wenn Sie mit der rechten Maustaste auf einen Algorithmus im Darstellungsbereich klicken, sehen Sie ein Kontextmenü wie das folgende:

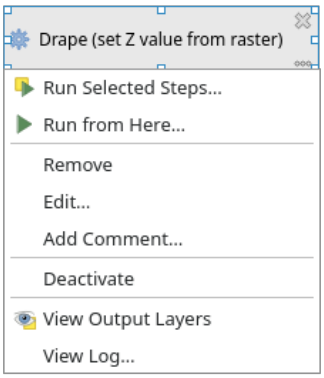


Abb. 23.29: Rechtsklick auf ein Modell-Element

- Optionen für *Ausführen eines Teils des Modells* wurden bereits zuvor vorgestellt.
- Durch Auswahl der Option *Entfernen* wird der ausgewählte Algorithmus entfernt. Ein Algorithmus kann nur entfernt werden, wenn keine anderen Algorithmen von ihm abhängig sind. Das heißt, wenn keine Ausgabe des Algorithmus als Eingabe in einem anderen Algorithmus verwendet wird.
- Wenn Sie die Option *Bearbeiten…* auswählen, wird das Parameterdialogfeld angezeigt, in dem Sie die Eingaben und Parameterwerte ändern können. Nicht alle im Modell verfügbaren Eingabeelemente werden als verfügbare Eingaben angezeigt. Layer oder Werte, die in einem fortgeschritteneren Schritt des vom Modell definierten Workflows generiert werden, sind nicht verfügbar, wenn sie zirkuläre Abhängigkeiten verursachen.

  Wählen Sie die neuen Werte aus und klicken Sie wie gewohnt auf die Schaltfläche *OK*. Die Verbindungen zwischen den Modellelementen ändern sich im Darstellungsbereich entsprechend.
- Mit *Kommentar hinzufügen…* können Sie dem Algorithmus einen Kommentar hinzufügen, um das Verhalten besser zu beschreiben.

Bemerkung

Durch einen Rechtsklick auf einen Eingabeparameter können Sie außerdem *Entfernen*, *Bearbeiten…* und *Kommentar hinzufügen…* auswählen.

- Ein Modell kann teilweise ausgeführt werden, indem einige seiner Algorithmen deaktiviert werden. Wählen Sie dazu die Option *Abschalten* im Kontextmenü, das erscheint, wenn Sie mit der rechten Maustaste auf ein Algorithmenelement klicken. Der ausgewählte Algorithmus und alle Algorithmen im Modell, die von ihm abhängen, werden grau dargestellt und nicht als Teil des Modells ausgeführt.

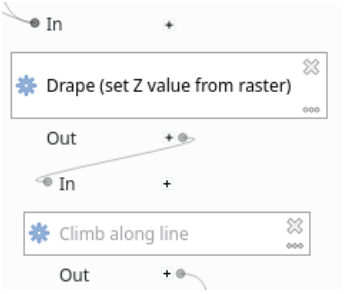


Abb. 23.30: Modell mit deaktiviertem Algorithmus

Wenn Sie mit der rechten Maustaste auf einen nicht aktiven Algorithmus klicken, wird die Menüoption *Aktivieren* angezeigt, mit der Sie ihn wieder aktivieren können.

- Wenn Sie ein Modell über den Designer bearbeiten (und nachdem Sie dieses Modell ausgeführt haben), können

  Sie mit einem Rechtsklick auf einen beliebigen untergeordneten Schritt im Modell die Option *Ausgabe-Layer anzeigen* auswählen. Dadurch werden die Ausgabe-Layer dieses Schritts als neue Layer zum aktuellen QGIS-Projekt hinzugefügt.

  Diese Aktion ist für alle untergeordneten Algorithmen im Modell verfügbar, auch wenn das Modell nicht so konfiguriert ist, dass die Ausgaben dieser untergeordneten Elemente als Modellausgaben verwendet werden. Dies ist als hilfreiche Debugging-Aktion konzipiert. Wenn ein Modell fehlschlägt (oder unerwartete Ergebnisse liefert), können Sie das Modell durchlaufen und die Ausgaben für vermutlich problematische Schritte anzeigen. Dadurch müssen Sie keine temporären Ausgaben zu einem Modell hinzufügen und es zum Testen erneut ausführen.

  Darüber hinaus ist diese Aktion nach dem Ausführen des Modells immer verfügbar, auch wenn das Modell selbst fehlgeschlagen ist (z. B. aufgrund eines falsch konfigurierten Schritts später im Modell).

- Mit der Aktion *Protokoll anzeigen* können Sie das Protokoll für jeden untergeordneten Algorithmus anzeigen, nachdem Sie den Algorithmus-Dialog geschlossen haben.

23.6 Die Schnittstelle zur Stapelverarbeitung

23.6.1 Einführung

Alle Algorithmen (einschließlich Modellen) können als Batch-Prozess ausgeführt werden. Das heißt, sie können nicht nur mit einem einzigen Set von Eingaben, sondern mit mehreren davon ausgeführt werden, wobei der Algorithmus so oft wie nötig ausgeführt wird. Dies ist nützlich bei der Verarbeitung großer Datenmengen, da der Algorithmus nicht mehrmals aus der Toolbox gestartet werden muss.

Um einen Algorithmus als Batch-Prozess ausführen, klicken Sie mit der rechten Maustaste auf seinen Namen in der Toolbox und wählen Sie die Option *Ausführen als Batch-Prozess*.

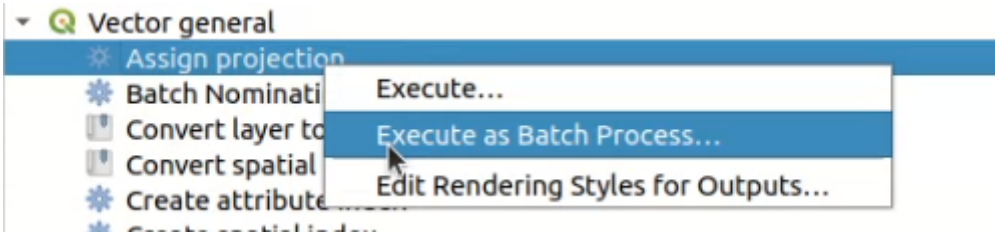


Abb. 23.31: Stapelverarbeitung mit der rechten Maustaste

Wenn Sie den Ausführungsdialog des Algorithmus geöffnet haben, können Sie die Stapelverarbeitung auch von dort starten, drücken Sie auf den *Als Stapelprozess ausführen…* Knopf.

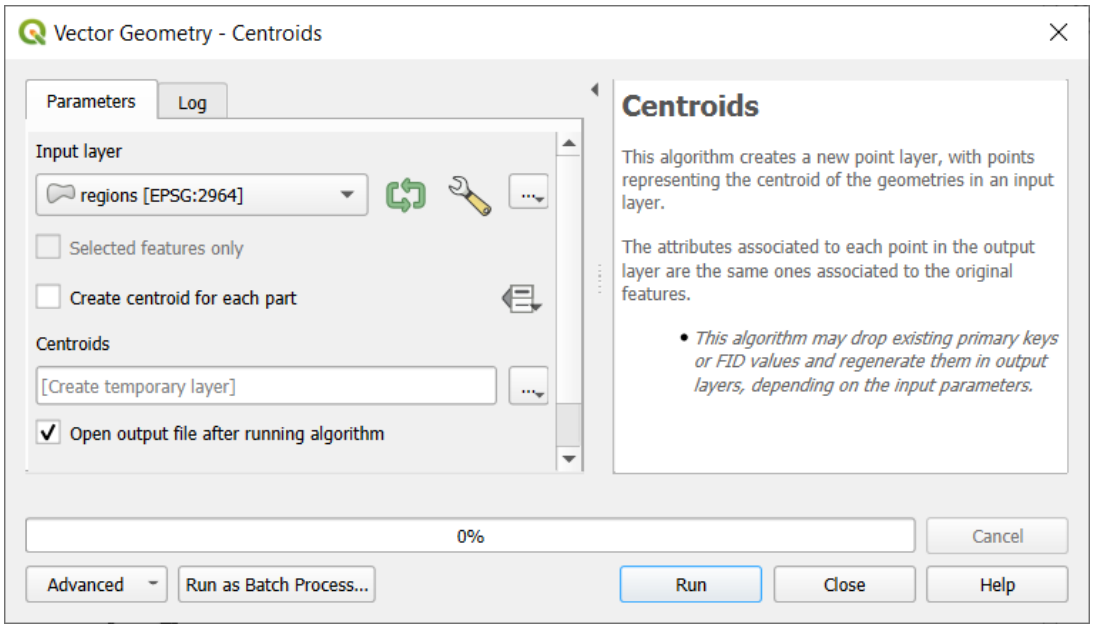


Abb. 23.32: Batch-Prozess vom Algorithmus Dialog

23.6.2 Die Parameter-Tabelle

Die Ausführung eines Batch-Prozesses ist vergleichbar mit der einmaligen Ausführung eines Algorithmus. Die Parameterwerte müssen definiert werden, aber in diesem Fall brauchen wir nicht nur einen einzigen Wert für jeden Parameter, sondern eine Reihe von ihnen, einen für jedes Mal, wenn der Algorithmus ausgeführt werden muss. Die Werte werden mithilfe einer Tabelle wie der folgenden eingegeben, wobei jede Zeile eine Iteration und die Spalten die Parameter des Algorithmus darstellen.

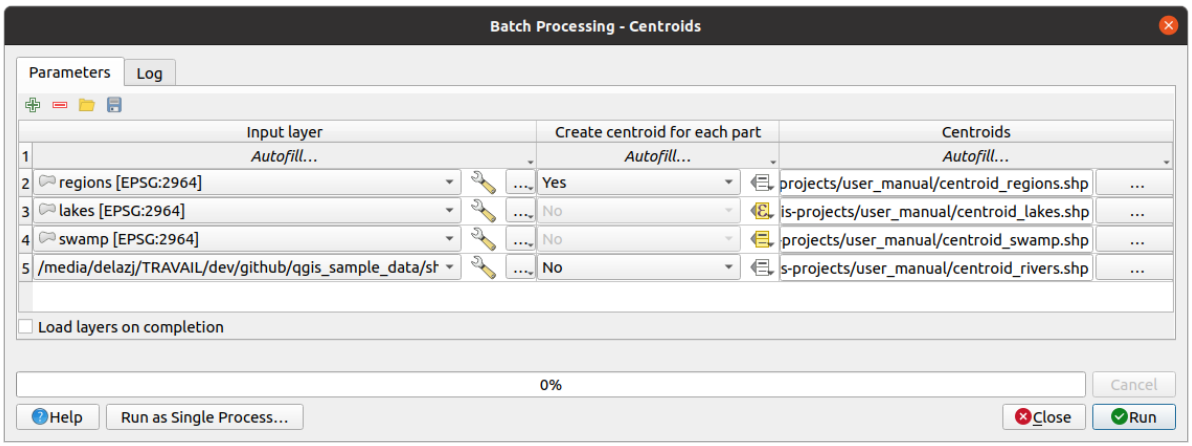


Abb. 23.33: Stapelverarbeitung

In der oberen Werkzeugleiste können Sie:

- Erweiterten Modus einschalten: Diese Schaltfläche ist nur verfügbar, wenn der Algorithmus Parameter hat, die als fortgeschritten markiert sind. Mit dieser Schaltfläche können Sie solche Parameter im Batch-Dialog ein- oder ausblenden.

- Zeile hinzufügen: fügt einen neuen Verarbeitungseintrag für die Konfiguration hinzu
- Zeile(n) entfernen: entfernt ausgewählte Zeilen aus der Tabelle. Die Zeilenauswahl erfolgt durch Anklicken der Nummer auf der linken Seite und erlaubt *Tastaturkombination* für Mehrfachauswahl.
- Öffnen einer Stapelverarbeitungs-Konfigurationsdatei
- Speichern der Stapelverarbeitungskonfiguration in eine `.JSON`-Datei, die anschließend ausgeführt werden kann

Standardmäßig enthält die Tabelle nur zwei Zeilen:

- In der ersten Zeile wird in jeder Zelle ein *Autofüllung…* ► Dropdown-Menü mit *Optionen* angezeigt, um die Zellen darunter schnell zu füllen. Die verfügbaren Optionen hängen vom Parametertyp ab.
- Die zweite Zeile (und jede weitere) stellt eine einzelne Ausführung des Algorithmus dar, und jede Zelle enthält den Wert eines der Parameter. Es ähnelt dem Parameterdialog, den Sie sehen, wenn Sie einen Algorithmus aus der Werkzeugkiste ausführen, allerdings mit einer anderen Anordnung.

Am unteren Ende der Tabelle können Sie einstellen, ob Sie die *Layer bei Abschluss laden* wollen.

Sobald die Größe der Tabelle gesetzt ist, muss sie mit den gewünschten Werten gefüllt werden.

23.6.3 Füllen der Parameter-Tabelle

Für die meisten Parameter ist die Einstellung des Wertes trivial. Das entsprechende Widget, dasselbe wie im *Einzelprozessdialog*, wird zur Verfügung gestellt und erlaubt es, den Wert einfach einzugeben oder ihn aus einer Liste möglicher Werte auszuwählen, abhängig vom Parametertyp. Dies beinhaltet auch das Daten-Definitions-Widget, wenn es kompatibel ist.

Um die Definition des Batch-Prozesses zu automatisieren und zu vermeiden, dass die Tabelle Zelle für Zelle gefüllt wird, können Sie das Menü *Autofüllung…* eines Parameters drücken und eine der folgenden Optionen auswählen, um Werte in der Spalte zu ersetzen:

- *Nach unten füllen* nimmt die Eingabe für den ersten Prozess und gibt sie für alle anderen Prozesse ein.
- *Nach Ausdruck berechnen…* ermöglicht es Ihnen, einen neuen QGIS-Ausdruck zu erstellen, mit dem Sie alle vorhandenen Werte in dieser Spalte aktualisieren können. Vorhandene Parameterwerte (einschließlich derer aus anderen Spalten) können innerhalb des Ausdrucks über *Variablen* verwendet werden. So kann z.B. die Anzahl der Segmente basierend auf dem Pufferabstand der einzelnen Layer festgelegt werden:

```
CASE WHEN @DISTANCE > 20 THEN 12 ELSE 8 END
```

- *Werte nach Ausdruck hinzufügen…* fügt neue Zeilen hinzu, indem es die Werte eines Ausdrucks verwendet, der ein Array zurückgibt (im Gegensatz zu *Nach Ausdruck berechnen…*, das nur mit bestehenden Zeilen arbeitet). Der beabsichtigte Anwendungsfall ist, das Auffüllen des Batch-Dialogs mit komplexen numerischen Reihen zu ermöglichen. Zum Beispiel führt das Hinzufügen von Zeilen für einen Stapelpuffer mit dem Ausdruck `generate_series(100, 1000, 50)` für den Parameter distance zu neuen Zeilen mit den Werten 100, 150, 200, …. 1000.
- Bei der Einstellung eines Datei- oder Layer-Parameters stehen weitere Optionen zur Verfügung:
  - *Dateien nach Muster hinzufügen…*: fügt der Tabelle neue Zeilen für Dateien hinzu, die einem *Dateimuster* in einem Ordner entsprechen, der mit *Suchen in* gekennzeichnet ist. Z.B. `*.shp` fügt der Liste alle `SHP` Dateien in dem Ordner hinzu. Aktivieren Sie das Kontrollkästchen *Rekursiv suchen*, um auch Unterordner zu durchsuchen.
  - *Dateien wählen…* einzeln auf Festplatte
  - *Alle Dateien aus einem Verzeichnis hinzufügen…*
  - *Aus geöffenten Layern wählen…* im aktuellen Projekt

Der Parameter für die Ausgabedaten bietet die gleichen Möglichkeiten wie bei der Ausführung des Algorithmus als einzelner Prozess. Je nach Algorithmus kann die Ausgabe sein:

- übersprungen, wenn die Zelle leer bleibt
- 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).

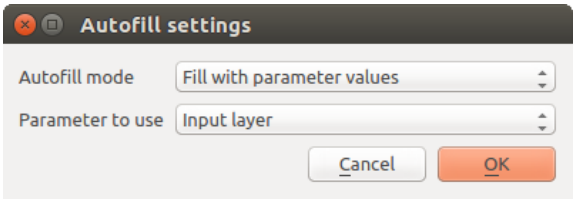


Abb. 23.34: Stapelverarbeitung Speichern

  Wenn der Standardwert (*Nicht automatisch ausfüllen*) ausgewählt ist, wird lediglich der ausgewählte Dateiname in die ausgewählte Zelle der Parametertabelle eingetragen. Wenn eine der anderen Optionen ausgewählt wird, werden alle Zellen **unter** der ausgewählten Zelle automatisch anhand bestimmter Kriterien ausgefüllt:

  - *Mit Zahlen füllen*: fügt schrittweise eine Zahl an den Dateinamen an
  - *Füllen mit Parameterwerten*: Sie können einen Parameter auswählen, dessen Wert in derselben Zeile an den Dateinamen angehängt wird. Dies ist besonders nützlich, um Ausgabedatenobjekte entsprechend den Eingabedaten zu benennen.

- als Layer in einem Datenbankcontainer gespeichert:

```
# 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 Ausführen eines Batch-Prozesses

Um den Batch-Prozess auszuführen, nachdem Sie alle erforderlichen Werte eingegeben haben, klicken Sie einfach auf *Starte*. Das Bedienfeld *Protokoll* wird aktiviert und zeigt Details und Schritte des Ausführungsprozesses an. Der Fortschritt des gesamten Stapelverarbeitungsprozesses wird in der Fortschrittsleiste im unteren Teil des Dialogs angezeigt.

23.7 Verarbeitungs-Algorithmen von der Konsole aus verwenden

Die Konsole ermöglicht es fortgeschrittenen Anwendern ihre Produktivität zu erhöhen und komplexe Operationen, die nicht anhand eines der GUI Elemente der Verarbeitungsumgebung ausgeführt werden können, durchzuführen. Modelle mit mehreren Algorithmen können anhand der Kommandozeilenschnittstelle definiert werden und zusätzliche Operationen wie Schleifen und Bedingungssätze können hinzugefügt werden, um flexiblere und leistungsfähigere Workflows zu erstellen.

Es gibt keine Verarbeitungskonsole in QGIS, aber alle Verarbeitungsbefehle sind stattdessen über die in QGIS integrierte *Python-Konsole* verfügbar. Das bedeutet, dass Sie diese Befehle in Ihre Konsolenarbeit einbinden und Verarbeitungsalgorithmen mit allen anderen Elementen (einschließlich Methoden der QGIS-API) verbinden können, die von dort aus verfügbar sind.

Der Code, den Sie von der Python Konsole aus ausführen können, auch wenn er keine spezifische Verarbeitungs-Methode aufruft, kann in einen neuen Algorithmus überführt werden, den Sie später aus der Werkzeugkiste, der Grafischen Modellierung oder jeder anderen Komponente aufrufen können, so wie Sie es mit jedem anderen Algorithmus tun würden. In der Tat sind einige Algorithmen, die Sie in der Werkzeugkiste finden können, einfache Scripte.

In diesem Abschnitt werden wir uns angucken, wie man Verarbeitungsalgorithmen aus der QGIS Python Konsole heraus verwendet und auch wie man Algorithmen in Python schreibt.

23.7.1 Algorithmen von der Python-Konsole aus aufrufen

Das erste, was Sie tun müssen, ist die Verarbeitungs-Funktionen mit der folgenden Zeile importieren:

```
>>> from qgis import processing
```

Nun gibt es im Grunde nur eine (interessante) Sache, die Sie damit von der Konsole aus tun können: einen Algorithmus ausführen. Dazu wird die Methode `run()` verwendet, die den Namen des auszuführenden Algorithmus als ersten Parameter und dann eine variable Anzahl zusätzlicher Parameter je nach den Anforderungen des Algorithmus erhält. Das Erste, was Sie wissen müssen, ist der Name des auszuführenden Algorithmus. Dabei handelt es sich nicht um den Namen, den Sie in der Toolbox sehen, sondern um einen eindeutigen Befehlszeilennamen. Um den richtigen Namen für Ihren Algorithmus zu finden, können Sie die `processingRegistry` verwenden. Geben Sie die folgende Zeile in Ihre Konsole ein:

```
>>> for alg in QgsApplication.processingRegistry().algorithms():
        print(alg.id(), "->", alg.displayName())
```

Sie sehen dann ungefähr so aus (mit einigen zusätzlichen Bindestrichen, um die Lesbarkeit zu verbessern):

```
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
...
```

Das ist eine Liste aller verfügbaren Algorithmus-IDs, sortiert nach Anbietername und Algorithmusname, zusammen mit den entsprechenden Namen.

Wenn Sie den Kommandozeilennamen des Algorithmus kennen, müssen Sie als Nächstes die richtige Syntax für seine Ausführung festlegen. Dazu muss man wissen, welche Parameter beim Aufruf der Methode `run()` benötigt werden.

Es gibt eine Methode zur detaillierten Beschreibung eines Algorithmus, mit der man eine Liste der Parameter, die ein Algorithmus benötigt, und der Ausgaben, die er erzeugt, erhalten kann. Um diese Informationen zu erhalten, können Sie die Methode `algorithmHelp(id_of_the_algorithm)` verwenden. Verwenden Sie die ID des Algorithmus, nicht den vollständigen beschreibenden Namen.

Ruft man die Methode mit `native:buffer` als Parameter auf (`qgis:buffer` ist ein Alias für `native:buffer` und funktioniert ebenfalls), erhält man folgende Beschreibung:

```
>>> processing.algorithmHelp("native:buffer")
Buffer (native:buffer)

This algorithm computes a buffer area for all the features in an
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
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
END_CAP_STYLE: End cap style

   Parameter type: QgsProcessingParameterEnum

   Available values:
           - 0: Round
           - 1: Flat
           - 2: Square

   Accepted data types:
           - 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
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
----------------

OUTPUT:  <QgsProcessingOutputVectorLayer>
   Buffered
```

Jetzt haben Sie alles, was Sie brauchen, um jeden Algorithmus auszuführen. Wie wir bereits erwähnt haben, können Algorithmen mit Hilfe von: `run()`. Die Syntax lautet wie folgt:

```
>>> processing.run(name_of_the_algorithm, parameters)
```

Dabei ist ‚parameters' ein Verzeichnis mit Parametern, die von dem Algorithmus abhängen, den Sie ausführen wollen, und ist genau die Liste, die Ihnen die Methode `algorithmHelp()` liefert.

```
>>> 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'})
```

Wenn ein Parameter optional ist und Sie ihn nicht verwenden wollen, dann nehmen Sie ihn nicht in das Verzeichnis auf.

Wird ein Parameter nicht angegeben, so wird der Standardwert verwendet.

Abhängig vom Parametertyp werden Werte verschieden eingeführt. Die nächste Liste gibt einen kurzen Überblick darüber, wie man Werte für jeden Typ von Eingabeparameter einführt:

- Rasterlayer, Vektorlayer oder Tabelle. Verwenden Sie einfach eine Zeichenkette mit dem Namen, der das zu verwendende Datenobjekt identifiziert (der Name, den es im QGIS-Inhaltsverzeichnis hat) oder einen Dateinamen (wenn der entsprechende Layer nicht geöffnet ist, wird er geöffnet, aber nicht zur Kartenansicht hinzugefügt). Wenn Sie über eine Instanz eines QGIS-Objekts verfügen, das den Layer repräsentiert, können Sie auch dieses als Parameter übergeben
- Aufzählung. Wenn ein Algorithmus einen Aufzählungsparameter hat, sollte der Wert dieses Parameters mit einem Integer-Wert eingegeben werden. Um die verfügbaren Optionen zu kennen, können Sie den Befehl `algorithmHelp()` verwenden, wie oben beschrieben. Zum Beispiel hat der `native:buffer` Algorithmus eine Aufzählung namens 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
  ```

  In diesem Fall hat der Parameter drei Optionen. Beachten Sie, dass die Reihenfolge nullbasiert ist.
- Boolesch. Verwenden Sie `True` oder `False`.
- Multiple input. Der Wert ist ein String mit Eingabebeschreibungen getrennt durch Semikolons (`;`). Wie im Fall von einfachen Layern oder Tabellen kann jede Eingabebeschreibung der Datenobjektname oder sein Dateipfad sein.

- Tabellen Feld von XXX. Verwenden Sie einen String mit dem Namen des Feldes, das benutzt werden soll. Dieser Parameter unterscheidet zwischen Groß- und Kleinschreibung.
- Fixed Table. Geben Sie die Liste aller Tabellenwerte, die durch Kommas (`,`) getrennt sind und zwischen Anführungsstrichen (`"`) eingeschlossen sind. Die Werte beginnen in der oberen Zeile und gehen von links nach rechts. Sie können auch einen 2D-Array von Werten, die die Tabelle repräsentieren, verwenden.
- CRS. Geben Sie den EPSG Code des gewünschten KBS ein.
- Extent. Sie müssen einen String mit `xmin`, `xmax`, `ymin` und `ymax` Werten getrennt durch Kommas (`,`) eingeben.

Boolean, file, string und numerical parameters brauchen keien zusätzlichen Erläuterungen.

Eingabeparameter wie Strings, Boolesche oder numerische Werte haben Standardwerte. Der Standardwert wird verwendet, wenn der entsprechende Parametereintrag nicht vorhanden ist.

Bei Ausgabedatenobjekten geben Sie den Dateipfad ein, der zum Speichern verwendet werden soll, so wie es auch in der Toolbox geschieht. Wenn das Ausgabeobjekt nicht angegeben wird, wird das Ergebnis in einer temporären Datei gespeichert (oder übersprungen, wenn es sich um eine optionale Ausgabe handelt). Die Erweiterung der Datei bestimmt das Dateiformat. Wenn Sie eine Dateierweiterung eingeben, die vom Algorithmus nicht unterstützt wird, wird das Standarddateiformat für diesen Ausgabetyp verwendet und die entsprechende Erweiterung an den angegebenen Dateipfad angehängt.

Anders als bei der Ausführung eines Algorithmus über die Toolbox werden die Ausgaben nicht zur Kartenansicht hinzugefügt, wenn Sie denselben Algorithmus über die Python-Konsole mit der Methode `run()` ausführen. Diese Methode gibt ein Verzeichnis mit einem oder mehreren Ausgabennamen (die in der Algorithmusbeschreibung angegeben sind) als Schlüssel und den Dateipfaden dieser Ausgaben als Werte zurück:

```
>>> 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
```

Sie können die Ausgabe dann als einen beliebigen allgemeinen Layer in das Projekt laden:

```
>>> buffered_layer = myresult['OUTPUT']
>>> QgsProject.instance().addMapLayer(buffered_layer)
```

Um die Verarbeitungsergebnisse sofort in das Projekt zu laden, können Sie die Methode `runAndLoadResults()` anstelle von `run()` verwenden.

```
>>> processing.runAndLoadResults("native:buffer", {parameters:values})
```

Wenn Sie einen Algorithmus-Dialog von der Konsole aus öffnen wollen, können Sie die Methode `createAlgorithmDialog` verwenden. Der einzige obligatorische Parameter ist der Name des Algorithmus, aber Sie können auch ein Verzeichnis von Parametern definieren, sodass der Dialog automatisch ausgefüllt wird:

```
>>> 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()
```

Die Methode `execAlgorithmDialog` öffnet den Dialog sofort:

```
>>> 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 Scripte erstellen und diese aus der Werkzeugkiste starten.

Sie können Ihre eigenen Algorithmen erstellen, indem Sie Python-Code schreiben. Verarbeitungsskripte erweitern `QgsProcessingAlgorithm`, so dass Sie einige zusätzliche Codezeilen hinzufügen müssen, um obligatorische Funktionen zu implementieren. Sie finden *Neues Skript erstellen* (leeres Blatt) und *Neues Skript aus Vorlage erstellen* (Vorlage, die Code für die obligatorischen Funktionen von `QgsProcessingAlgorithm` enthält) unter dem Dropdown-Menü *Skripte* oben in der Verarbeitungs- Werkzeugkiste. Es öffnet sich der Verarbeitungsskript-Editor, in den Sie Ihren Code eingeben können. Wenn Sie das Skript von dort aus im Ordner `scripts` (dem Standardordner, wenn Sie den Dialog zum Speichern von Dateien öffnen) mit der Erweiterung `.py` speichern, wird der entsprechende Algorithmus erstellt.

Der Name des Algorithmus (derjenige, den Sie in der Toolbox sehen) wird im Code definiert.

Schauen wir uns den folgenden Code an, der einen Verarbeitungsalgorithmus definiert, der eine Pufferoperation mit einem benutzerdefinierten Pufferabstand auf einem vom Benutzer angegebenen Vektor-Layer durchführt, nachdem er den Layer zunächst geglättet hat.

```
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(
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
            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',
            {'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}
```

Nach der Durchführung der erforderlichen Importe werden die folgenden `QgsProcessingAlgorithm`-Funktionen angegeben:

- `name()`: Die ID des Algorithmus (Kleinbuchstaben).
- `DisplayName()`: Ein von Menschen lesbarer Name für den Algorithmus.
- `createInstance()`: Erzeugt eine neue Instanz der Algorithmusklasse.
- `initAlgorithmus()`: Konfigurieren Sie die parameterDefinitionen und outputDefinitionen.

  Hier beschreiben Sie die Parameter und die Ausgabe des Algorithmus. In diesem Fall eine Objektquelle für die Eingabe, eine Objektsenke für das Ergebnis und eine Zahl für den Pufferabstand.

- `processAlgorithm()`: Erledigt die Arbeit.

  Hier führen wir zuerst den `smoothgeometry`-Algorithmus aus, um die Geometrie zu glätten, und dann lassen wir den `buffer`-Algorithmus auf die geglättete Ausgabe laufen. Um Algorithmen innerhalb eines anderen Algorithmus ausführen zu können, müssen wir das Argument `is_child_algorithm` auf `True` setzen. Sie können sehen, wie Eingabe- und Ausgabeparameter als Parameter für die Algorithmen `smoothgeometry` und `buffer` verwendet werden.

Es gibt eine Reihe von verschiedenen Parametertypen für die Ein- und Ausgabe. Die vollständige Liste finden Sie unter *Eingabe- und Ausgabetypen für Verarbeitungsalgorithmen*.

Der erste Parameter der Konstruktoren ist der Name des Parameters, der zweite ist die Beschreibung des Parameters (für die Benutzeroberfläche). Der Rest der Konstruktorparameter ist parameterartspezifisch.

Die Eingabe kann in QGIS-Klassen umgewandelt werden, indem man die `parameterAs`-Funktionen von `QgsProcessingAlgorithm` benutzt. Zum Beispiel um die Zahl für den Pufferabstand als Double zu erhalten:

```
self.parameterAsDouble(parameters, self.INPUT_BUFFERDIST, context)).
```

Die Funktion `processAlgorithm` sollte ein Wörterbuch zurückgeben, das Werte für jede durch den Algorithmus definierte Ausgabe enthält. Dies ermöglicht den Zugriff auf diese Ausgaben von anderen Algorithmen aus, einschließlich anderer Algorithmen innerhalb desselben Modells.

Gut funktionierende Algorithmen sollten so viele Ausgaben definieren und zurückgeben, wie es sinnvoll ist. Nicht-Feature-Ausgaben wie Zahlen und Strings sind sehr nützlich, wenn Ihr Algorithmus als Teil eines größeren Modells ausgeführt wird, da diese Werte als Eingabeparameter für nachfolgende Algorithmen innerhalb des Modells verwendet

werden können. Ziehen Sie in Erwägung, numerische Ausgaben für Dinge wie die Anzahl der verarbeiteten Objekte, die Anzahl der gefundenen ungültigen Objekte, die Anzahl der ausgegebenen Objekte usw. hinzuzufügen. Je mehr Ausgaben Sie zurückgeben, desto nützlicher wird Ihr Algorithmus!

Rückmeldungen

Das `feedback`-Objekt, das an `processAlgorithm()` übergeben wird, sollte für Benutzer-Feedback / Interaktion verwendet werden. Sie können die Funktion `setProgress()` des Objekts `feedback` verwenden, um den Fortschrittsbalken (0 bis 100) zu aktualisieren und den Benutzer über den Fortschritt des Algorithmus zu informieren. Dies ist sehr nützlich, wenn Ihr Algorithmus sehr lange braucht, um abgeschlossen zu werden.

Das Objekt `feedback` stellt eine Methode `isCanceled()` zur Verfügung, die überwacht werden sollte, um den Abbruch des Algorithmus durch den Benutzer zu ermöglichen. Die Methode `pushInfo()` von `feedback` kann verwendet werden, um Informationen an den Benutzer zu senden, und `reportError()` ist praktisch, um nicht schwerwiegende Fehler an Benutzer zu senden.

Algorithmen sollten andere Formen der Rückmeldung an den Benutzer, wie z. B. Druckanweisungen oder die Protokollierung in `QgsMessageLog`, vermeiden und stattdessen immer das Feedback-Objekt verwenden. Dies ermöglicht eine ausführliche Protokollierung für den Algorithmus und ist außerdem thread-sicher (was wichtig ist, da Algorithmen normalerweise in einem Hintergrund-Thread ausgeführt werden).

Umgang mit Fehlern

Wenn Ihr Algorithmus auf einen Fehler stößt, der ihn an der Ausführung hindert, z. B. ungültige Eingabewerte oder eine andere Bedingung, von der er sich nicht erholen kann oder soll, dann sollten Sie eine `QgsProcessingException` auslösen. Z.B.:

```
if feature['value'] < 20:
  raise QgsProcessingException('Invalid input value {}, must be >= 20'.
↪format(feature['value']))
```

Versuchen Sie zu vermeiden, `QgsProcessingException` für nicht-fatale Fehler auszulösen (z.B. wenn ein Objekt eine Null-Geometrie hat), und melden Sie diese Fehler stattdessen einfach über `feedback.reportError()` und überspringen Sie das Objekt. Dies trägt dazu bei, Ihren Algorithmus „modellfreundlich“ zu machen, da es vermeidet, die Ausführung eines gesamten Algorithmus anzuhalten, wenn ein nicht-fataler Fehler auftritt.

Ihre Scripte dokumentieren

Wie bei den Modellen können Sie auch für Ihre Skripte eine zusätzliche Dokumentation erstellen, um zu erklären, was sie tun und wie sie zu verwenden sind.

`QgsProcessingAlgorithm` bietet die `helpString()`, `shortHelpString()` und `helpUrl()` Funktionen für diesen Zweck. Geben Sie diese Funktionen an oder überschreiben Sie sie, um dem Benutzer mehr Hilfe zu bieten.

`shortDescription()` wird im Tooltip verwendet, wenn der Mauszeiger über den Algorithmus in der Toolbox bewegt wird.

23.7.3 Skript-Hooks vor und nach der Ausführung

Skripte können auch als Pre- und Post-Execution Hooks verwendet werden, die vor bzw. nach der Ausführung eines Algorithmus ausgeführt werden. Auf diese Weise können Aufgaben automatisiert werden, die bei der Ausführung eines Algorithmus ausgeführt werden sollen.

Die Syntax ist identisch zu der oben erklärten Syntax, es steht aber eine zusätzliche Globalvariable genannt `alg` zur Verfügung, die den Algorithmus, der gerade ausgeführt wurde (oder ausgeführt wird), repräsentiert.

In der Gruppe *Allgemein* des Dialogs Verarbeitungsoptionen finden Sie zwei Einträge namens *Pre-execution script* und *Post-execution script*, in die Sie die Dateinamen der jeweils auszuführenden Skripte eintragen können.

23.8 Verarbeitung über die Befehlszeile

QGIS wird mit einem Werkzeug namens `QGIS Processing Executor` ausgeliefert, das es Ihnen erlaubt, Processing-Algorithmen und -Modelle (eingebaut oder durch Plugins bereitgestellt) direkt von der Kommandozeile aus zu starten, ohne QGIS Desktop selbst zu starten.

Führen Sie `qgis_process` in einem Kommandozeilenprogramm aus und Sie sollten folgendes erhalten:

```
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.
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
                    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.
```

Bemerkung

Nur installierte Plugins, die in ihrer `metadata.txt` Datei `hasProcessingProvider=yes` angeben, werden erkannt und können vom qgis_process Tool aktiviert oder geladen werden.

Hinweis

Bevor Sie qgis_process auf einem System ohne Fenstermanager (z.B. einem Headless Server) aufrufen, sollten Sie folgende Einstellung setzen:

```
export QT_QPA_PLATFORM=offscreen
```

Der Befehl `list` kann verwendet werden, um eine Liste aller verfügbaren Anbieter und Algorithmen zu erhalten:

```
qgis_process list
```

Der Befehl `help` kann verwendet werden, um weitere Informationen über Befehle oder Algorithmen zu erhalten:

```
qgis_process help qgis:regularpoints
```

Der Befehl `run` kann verwendet werden, um einen Algorithmus oder ein Modell auszuführen. Geben Sie als ersten Parameter den Namen des Algorithmus oder einen Pfad zu einem Modell an:

```
qgis_process run native:buffer -- INPUT=source.shp DISTANCE=2 OUTPUT=buffered.shp
```

Wenn ein Parameter eine Liste von Werten akzeptiert, setzen Sie dieselbe Variable mehrfach:

```
qgis_process run native:mergevectorlayers -- LAYERS=input1.shp LAYERS=input2.shp␣
↪OUTPUT=merged.shp
```

Während der Ausführung eines Algorithmus wird eine textbasierte Rückmeldungsleiste angezeigt, und der Vorgang kann mit `Strg+C` abgebrochen werden.

Der Befehl `run` unterstützt auch weitere Parameter:

- `--json` formatiert die stdout-Ausgabe in einer JSON-strukturierten Weise.
- `--ellipsoid` setzt das Ellipsoid auf den angegebenen Wert.
- `--distance_units` wird die angegebenen Entfernungseinheiten verwenden.
- `--area_units` wird die angegebenen Flächeneinheiten verwenden.
- `--project_path` lädt das angegebene Projekt zur Ausführung des Algorithmus.

Auch komplexe Eingabeparameter werden von qgis_process unterstützt. Um anzuzeigen, dass die Parameter über stdin angegeben werden, muss der `qgis_process`-Befehl dem unten gezeigten Format folgen (mit einem abschließenden – anstelle der üblichen Argumentliste):

```
qgis_process run algorithmId -
```

Das JSON-Objekt muss einen „inputs“-Schlüssel enthalten, bei dem es sich um eine Abbildung der Werte der Eingabeparameter handelt, z.B.:

```
echo "{'inputs': {'INPUT': 'my_shape.shp', 'DISTANCE': 5}}" | qgis_process run␣
↪native:buffer -
```

Zusätzlich können zusätzliche Einstellungen wie Entfernungseinheiten, Flächeneinheiten, Ellipsoid und Projektpfad in dieses JSON-Objekt aufgenommen werden:

```
{
 'ellipsoid': 'EPSG:7019',
 'distance_units': 'feet',
 'area_units': 'ha',
 'project_path': 'C:/temp/my_project.qgs'
 'inputs': {'DISTANCE': 5, 'SEGMENTS': 8 ... }
}
```

Die Angabe von Eingabeparametern über stdin impliziert automatisch das Ausgabeformat `JSON` für die Ergebnisse.

23.9 Schreiben neuer Verarbeitungs-Algorithmen als Python-Skripte

Es gibt zwei Möglichkeiten, Processing-Algorithmen mit Python zu schreiben.

- *Erweitern Sie* `QgsProcessingAlgorithm`
- *Verwenden Sie @alg decorator*

In QGIS können Sie mit *Neues Skript erstellen* im Menü *Skripte* oben in der *Verarbeitungs-Werkzeugkiste* den *Verarbeitungsskript-Editor* öffnen, wo Sie Ihren Code schreiben können. Um die Aufgabe zu vereinfachen, können Sie mit einer Skriptvorlage beginnen, indem Sie *Neues Skript aus Vorlage erstellen* aus demselben Menü wählen. Dies öffnet eine Vorlage, die `QgsProcessingAlgorithm` erweitert.

Wenn Sie das Skript im Ordner `scripts` (dem Standardspeicherort) mit der Erweiterung `.py` speichern, wird der Algorithmus in der *Verarbeitungs-Werkzeugkiste* verfügbar.

23.9.1 Erweitern von QgsProcessingAlgorithm

Der folgende Code

1. verwendet einen Vektor Layer als Eingabe
2. zählt die Anzahl der Objekte
3. führt eine Puffer-Operation durch
4. erzeugt einen Raster Layer aus dem Ergebnis der Puffer-Operation
5. gibt den Puffer-Layer, den Rasterlayer und die Anzahl der Objekte zurück

```
1  from qgis.PyQt.QtCore import QCoreApplication
2  from qgis.core import (QgsProcessing,
3                         QgsProcessingAlgorithm,
4                         QgsProcessingException,
5                         QgsProcessingOutputNumber,
6                         QgsProcessingParameterDistance,
7                         QgsProcessingParameterFeatureSource,
8                         QgsProcessingParameterVectorDestination,
9                         QgsProcessingParameterRasterDestination)
10 from qgis import processing
11
12
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
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):
        # 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(
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
            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')
            )
        )
        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',
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
            {
                # 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
            # 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}
```

Standardfunktionen des Verarbeitungsalgorithmus:

- **createInstance (obligatorisch)**
  Muss eine neue Kopie Ihres Algorithmus zurückgeben. Wenn Sie den Namen der Klasse ändern, stellen Sie sicher, dass Sie auch den hier zurückgegebenen Wert entsprechend aktualisieren!
- **name (obligatorisch)**
  Gibt den eindeutigen Algorithmusnamen zurück, der zur Identifizierung des Algorithmus dient.
- **displayName (obligatorisch)**
  Gibt den übersetzten Algorithmusnamen zurück.

- **group**
  Gibt den Namen der Gruppe zurück, zu der dieser Algorithmus gehört.
- **groupId**
  Gibt die eindeutige ID der Gruppe zurück, zu der dieser Algorithmus gehört.
- **shortHelpString**
  Gibt eine kurze Hilfezeichenkette für den Algorithmus zurück.
- **initAlgorithm (obligatorisch)**
  Hier werden die Ein- und Ausgaben des Algorithmus definiert.

  `INPUT` und `OUTPUT` sind empfohlene Namen für die Haupteingabe- bzw. Hauptausgabeparameter.

  Wenn ein Parameter von einem anderen Parameter abhängt, wird `parentParameterName` verwendet, um diese Beziehung zu spezifizieren (könnte das Feld / Band eines Layers oder die Entfernungseinheiten eines Layers sein).
- **processAlgorithm (obligatorisch)**
  Hier findet die Verarbeitung statt.

  Parameter werden mit speziellen Funktionen abgerufen, zum Beispiel `parameterAsSource` und `parameterAsDouble`.

  `processing.run` kann verwendet werden, um andere Verarbeitungsalgorithmen von einem Verarbeitungsalgorithmus auszuführen. Der erste Parameter ist der Name des Algorithmus, der zweite ist ein Wörterbuch mit den Parametern des Algorithmus. Der Parameter `is_child_algorithm` wird normalerweise auf `True` gesetzt, wenn ein Algorithmus aus einem anderen Algorithmus heraus gestartet wird. `Context` und `Feedback` informieren den Algorithmus über die Umgebung, in der er laufen soll, und den Kanal für die Kommunikation mit dem Benutzer (Abbruch einer Anfrage, Meldung des Fortschritts, textuelles Feedback). Wenn die Parameter des (übergeordneten) Algorithmus als Parameter für „untergeordnete“ Algorithmen verwendet werden, sollten die ursprünglichen Parameterwerte verwendet werden (z.B. `parameters['OUTPUT']`).

  Es ist eine gute Praxis, das Feedback-Objekt so oft wie möglich auf Abbruch zu prüfen! Dies ermöglicht eine schnelle Beendigung, anstatt den Benutzer zu zwingen, auf eine unerwünschte Verarbeitung zu warten.

  Der Algorithmus sollte Werte für alle Ausgabeparameter zurückgeben, die er in einem Verzeichnis definiert hat. In diesem Fall sind das die Puffer- und gerasterten Output-Layer sowie die Anzahl der verarbeiteten Objekte. Die Verzeichnisschlüssel müssen mit den ursprünglichen Namen der Parameter/Ausgaben übereinstimmen.

23.9.2 Der @alg decorator

Mit dem @alg decorator können Sie Ihre eigenen Algorithmen erstellen, indem Sie den Python-Code schreiben und ein paar zusätzliche Zeilen hinzufügen, um zusätzliche Informationen zu liefern, die erforderlich sind, um ihn zu einem richtigen Processing-Algorithmus zu machen. Dies vereinfacht die Erstellung von Algorithmen und die Angabe von Eingaben und Ausgaben.

Eine wichtige Einschränkung des Decorator-Ansatzes ist, dass Algorithmen, die auf diese Weise erstellt werden, immer zum Processing Scripts Provider eines Benutzers hinzugefügt werden – es ist nicht möglich, diese Algorithmen zu einem benutzerdefinierten Provider hinzuzufügen, z.B. zur Verwendung in Plugins.

Der folgende Code verwendet den @alg decorater, um

1. einen Vektorlayer als Eingabe verwenden
2. die Anzahl der Objekte zählen
3. eine Puffer-Operation durchzuführen
4. einen Raster Layer aus dem Ergebnis der Pufferoperation erstellen
5. gibt den Puffer-Layer, den Rasterlayer und die Anzahl der Objekte zurück

```
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')
@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}
```

Wie Sie sehen können, handelt es sich um zwei Algorithmen (‚native:buffer' und ‚qgis:rasterize'). Der letzte (‚qgis:rasterize') erzeugt einen Raster Layer aus dem Puffer Layer, der durch den ersten (‚native:buffer') erzeugt wurde.

Der Teil des Codes, in dem diese Verarbeitung stattfindet, ist nicht schwer zu verstehen, wenn Sie das vorherige Kapitel gelesen haben. Die ersten Zeilen bedürfen jedoch einiger zusätzlicher Erläuterungen. Sie liefern die Informationen, die benötigt werden, um Ihren Code in einen Algorithmus zu verwandeln, der von jeder der GUI-Komponenten, wie der Toolbox oder dem Modelldesigner, ausgeführt werden kann.

Diese Zeilen sind allesamt Aufrufe der `@alg`-decorator-Funktionen, die die Codierung des Algorithmus vereinfachen.

- Der @alg decorator wird verwendet, um den Namen und die Position des Algorithmus in der Werkzeugkiste zu definieren.
- Der @alg.input decorator wird verwendet, um die Eingaben des Algorithmus zu definieren.
- Der @alg.output decorator wird verwendet, um die Ausgaben des Algorithmus zu definieren.

Für die vorhandenen Parameter und ihre Entsprechung lesen Sie bitte *Eingabe- und Ausgabetypen für Verarbeitungsalgorithmen*.

23.9.3 Handhabung der Algorithmus-Ausgabe

Wenn Sie eine Ausgabe deklarieren, die einen Layer (Raster oder Vektor) darstellt, wird der Algorithmus versuchen, diesen in QGIS einzufügen, sobald er fertig ist.

- Ausgabe eines Rasterlayers: QgsProcessingParameterRasterDestination / alg.RASTER_LAYER_DEST.
- Ausgabe eines Vektorlayers: QgsProcessingParameterVectorDestination / alg.VECTOR_LAYER_DEST.

Selbst wenn also die Methode `processing.run()` die von ihr erzeugten Layer nicht zum aktuellen Projekt des Benutzers hinzufügt, werden die beiden Ausgabelayer (Puffer und Rasterpuffer) geladen, da sie an den vom Benutzer eingegebenen Zielen gespeichert werden (oder an temporären Zielen, wenn der Benutzer keine Ziele angibt).

Wenn ein Layer als Ausgabe eines Algorithmus erstellt wird, sollte er auch als solcher deklariert werden. Andernfalls können Sie den Algorithmus im Modellierer nicht richtig verwenden, da das, was deklariert wird, nicht mit dem übereinstimmt, was der Algorithmus tatsächlich erzeugt.

Sie können Zeichenketten, Zahlen und mehr zurückgeben, indem Sie sie im Ergebniswörterbuch angeben (wie für „NUMBEROFFEATURES" gezeigt), aber sie sollten immer ausdrücklich als Ausgaben Ihres Algorithmus definiert werden. Wir ermutigen Algorithmen, so viele nützliche Werte wie möglich auszugeben, da diese für die Verwendung in späteren Algorithmen wertvoll sein können, wenn Ihr Algorithmus als Teil eines Modells verwendet wird.

23.9.4 Kommunikation mit dem Nutzer.

Wenn Ihr Algorithmus eine lange Zeit für die Verarbeitung benötigt, ist es eine gute Idee, den Benutzer über den Fortschritt zu informieren. Hierfür können Sie `Feedback` (`QgsProcessingFeedback`) verwenden.

Der Fortschrittstext und die Fortschrittsleiste können mit zwei Methoden aktualisiert werden: `setProgressText(text)` und `setProgress(percent)`.

Sie können weitere Informationen bereitstellen, indem Sie `pushCommandInfo(text)`, `pushDebugInfo(text)`, `pushInfo(text)` oder `reportError(text)` verwenden.

Wenn in Ihrem Skript ein Problem auftritt, besteht die korrekte Art der Behandlung darin, eine `QgsProcessingException` auszulösen. Sie können eine Nachricht als Argument an den Konstruktor der Ausnahme übergeben. Die Verarbeitung kümmert sich um die Behandlung und die Kommunikation mit dem Benutzer, je nachdem, von wo aus der Algorithmus ausgeführt wird (Toolbox, Modeler, Python-Konsole, …)

23.9.5 Ihre Scripte dokumentieren

Sie können Ihre Skripte unter Verwendung von `helpString()` und `helpUrl()`, Methoden von `QgsProcessingAlgorithm` dokumentieren.

23.9.6 Flags

Sie können die Methode `flags()` von `QgsProcessingAlgorithm` überschreiben, um QGIS mehr über Ihren Algorithmus zu erzählen. Sie können QGIS zum Beispiel mitteilen, dass das Skript vor dem Modellierer verborgen werden soll, dass es abgebrochen werden kann, dass es nicht thread-sicher ist und vieles mehr.

Tipp

Standardmäßig führt Processing Algorithmen in einem separaten Thread aus, damit QGIS während der Ausführung des Processing-Tasks ansprechbar bleibt. Wenn Ihr Algorithmus regelmäßig abstürzt, verwenden Sie wahrscheinlich API-Aufrufe, die nicht sicher in einem Hintergrund-Thread ausgeführt werden können. Versuchen Sie, das Flag QgsProcessingAlgorithm.FlagNoThreading aus der flags()-Methode Ihres Algorithmus zurückzugeben, um Processing zu zwingen, Ihren Algorithmus stattdessen im Hauptthread auszuführen.

23.9.7 Bewährte Verfahren für das Schreiben von Skript-Algorithmen

Hier finden Sie eine kurze Zusammenfassung von Ideen, die Sie bei der Erstellung Ihrer Skript-Algorithmen berücksichtigen sollten, vor allem, wenn Sie sie mit anderen QGIS-Benutzern teilen möchten. Wenn Sie diese einfachen Regeln befolgen, wird die Konsistenz zwischen den verschiedenen Verarbeitungselementen wie der Werkzeugkiste, dem Modellierer oder der Stapelverarbeitungsschnittstelle gewährleistet.

- Laden Sie keine Ergebnislayer. Lassen Sie die Verarbeitung Ihrer Ergebnisse abarbeiten und laden Sie Ihre Layer, wenn nötig.
- Geben Sie immer die Ausgaben an, die Ihr Algorithmus erzeugt.
- Zeigen Sie keine Meldungsfelder an und verwenden Sie keine GUI-Elemente aus dem Skript. Wenn Sie mit dem Benutzer kommunizieren möchten, verwenden Sie die Methoden des Feedback-Objekts (`QgsProcessingFeedback`) oder werfen Sie eine `QgsProcessingException`.

Es sind bereits viele Verarbeitungsalgorithmen in QGIS verfügbar. Sie können Code auf der QGIS Repo finden.

23.10 Eingabe- und Ausgabetypen für Verarbeitungsalgorithmen

Hier ist die Liste der Eingabe- und Ausgabetypen, die in der Verarbeitung unterstützt werden, mit ihren entsprechenden alg-decorator-Konstanten und ihrem Namen im Modellentwurf, falls vorhanden.

23.10.1 Eingabe-Typen

Eingabe-Name	Klasse	Alg constant	Beschreibung
Anmerkungen-Layer	`QgsProcessingParame`	`alg.ANNOTATION_LAYE`	Ein Anmerkungen-Layer
Fläche	`QgsProcessingParame`		Ein numerischer Parameter, der eine Flächenmessung darstellt

Fortsetzung auf der nächsten Seite

Tab. 23.1 – Fortsetzung der vorherigen Seite

Eingabe-Name	Klasse	Alg constant	Beschreibung
Authentifikationskonfi	QgsProcessingParame	alg.AUTH_CFG	Ermöglicht Benutzern die Auswahl aus verfügbaren Authentifizierungskonfigurationen oder die Erstellung neuer Authentifizierungskonfigurationen
Raste-Band	QgsProcessingParame	alg.BAND	Ein Band/Kanal eines Rasterlayers
Boolean	QgsProcessingParame	alg.BOOL	Ein boolescher Wert
Name der Verbindung	QgsProcessingParame		Eine Auswahl aus den verfügbaren registrierten Verbindungen für einen bestimmten Datenanbieter
Farbe	QgsProcessingParame	alg.COLOR	Eine Farbe
Koordinaten Operation	QgsProcessingParame	alg. COORDINATE_OPER	Eine Koordinaten-Operation (für KBS-Transformationen)
KBS	QgsProcessingParame	alg.CRS	Ein Koordinatenbezugssystem
Datenbank-Schema	QgsProcessingParame	alg. DATABASE_SCHEMA	Ein Datenbank-Schema
Datenbank-Tabelle	QgsProcessingParame	alg. DATABASE_TABLE	Eine Datenbank-Tabelle
Datum	ProcessingDateTimeP Date	alg.DATE	Ein reines Datum
Datum/Zeit	ProcessingDateTimeP DateTime	alg.DATETIME	Ein Datums- und Zeitwert
Entfernung	QgsProcessingParame	alg.DISTANCE	Ein numerischer Parameter für Abstandswerte
DXF-Layer	QgsProcessingParame		Eine Liste von Vektorlayern für den Export als DXF-Datei
Aufzählung	QgsProcessingParame	alg.ENUM	Eine Aufzählung, die eine Auswahl aus einer Reihe von vordefinierten Werten ermöglicht
Ausdruck	QgsProcessingParame	alg. EXPRESSION	Ein Ausdruck
Extent	QgsProcessingParame	alg.EXTENT	Eine räumliche Ausdehnung, definiert durch xmin, xmax, ymin, ymax
	QgsProcessingParame	alg.FIELD	Ein Feld in der Attributtabelle eines Vektorlayers
Feld-Aggregate	QgsProcessingParame		Ein Parameter für „Aggregat“-Konfigurationen, die aus einer Definition der gewünschten Ausgabefelder und -typen sowie des Aggregats bestehen, mit dem sie ausgefüllt werden
Felder-Mapper	QgsProcessingParame		Ein Parameter für „Field-Mapping“-Konfigurationen, die aus einer Definition der gewünschten Ausgabefelder, Typen und Ausdrücke bestehen, mit denen sie ausgefüllt werden
Datei/Ordner	QgsProcessingParame	alg.FILE	Ein Dateiname einer vorhandenen Datei
	QgsProcessingParame	alg.FILE_DEST	Ein Dateiname für eine neu erstellte Ausgabedatei

Fortsetzung auf der nächsten Seite

Tab. 23.1 – Fortsetzung der vorherigen Seite

Eingabe-Name	Klasse	Alg constant	Beschreibung
	QgsProcessingParame	alg. FOLDER_DEST	Ein Ordner (Zielordner)
Geometrie	QgsProcessingParame	alg.GEOMETRY	Eine Geometrie
	ProcessingNumberPar Integer	alg.INT	Eine Ganzzahl
Ein leeres Druck-Layout	QgsProcessingParame	alg.LAYOUT	Ein Layout
Druck-Layout Element	QgsProcessingParame	alg. LAYOUT_ITEM	Ein Layout-Element
Karten-Layer	QgsProcessingParame	alg.MAPLAYER	Ein Layer in der Karte
Karten-Thema	QgsProcessingParame	alg.MAP_THEME	Ein Karten-Thema aus dem Projekt
Matrix	QgsProcessingParame	alg.MATRIX	Eine Matrix
Netz-Datensatzgruppen	QgsProcessingParame		Datensatzgruppen eines Netzlayers
Netz-Datensatz Zeit	QgsProcessingParame		Datensatzindex aus einem Netzlayer nach Zeiteinstellung
Netz-Layer	QgsProcessingParame	alg. MESH_LAYER	Ein Netzlayer
Mehrfacheingabe	QgsProcessingParame	alg. MULTILAYER	Ein Satz von Layern
Nummer	ProcessingNumberPar Double	alg.NUMBER	Ein numerischer Wert
Punkt	QgsProcessingParame	alg.POINT	Ein Punkt
Punktwolken-Attribut	QgsProcessingParame		Ein Attribut aus einem Punktwolken-Layer
	QgsProcessingParame	alg. POINTCLOUD_LAYE	Ein Parameter für das Ziel eines Punktwolken-Layers, mit dem der Zielpfad für einen durch den Algorithmus erzeugten Punktwolken-Layer angegeben werden kann
Punktwolken-Layer	QgsProcessingParame	alg. POINTCLOUD_LAYE	Ein Punktwolken-Layer
	QgsProcessingParame	alg. PROVIDER_CONNEC	Eine verfügbare Verbindung für einen Datenbankanbieter
Bereich	QgsProcessingParame	alg.RANGE	Ein Nummernkreis
Rasterlayer	QgsProcessingParame	alg. RASTER_LAYER	Ein Rasterlayer
	QgsProcessingParame	alg. RASTER_LAYER_DE	Ein Rasterlayer-Zielparameter, um den Zielpfad für einen durch den Algorithmus erstellten Rasterlayer anzugeben
Maßstab	QgsProcessingParame	alg.SCALE	Ein Kartenmaßstab
	QgsProcessingParame	alg.SINK	A feature sink
Text	QgsProcessingParame	alg.STRING	Eine Zeichenkette
TIN-Erstellung Layer	QgsProcessingParame		Auswahl mehrerer Layer zur Erstellung eines TIN mit Scheitelpunkten und/oder Bruchlinien
Zeit	ProcessingDateTimeP Time	alg.TIME	Eine reine Zeit
Vektor-Objekte	QgsProcessingParame	alg.SOURCE	Eine Objekt-Quelle
Vektorlayer	QgsProcessingParame	alg. VECTOR_LAYER	Ein Vektorlayer

Fortsetzung auf der nächsten Seite

Tab. 23.1 – Fortsetzung der vorherigen Seite

Eingabe-Name	Klasse	Alg constant	Beschreibung
	QgsProcessingParame	alg. VECTOR_LAYER_DE	Ein Parameter für das Ziel eines Vektorlayers, um den Zielpfad für einen vom Algorithmus erzeugten Vektorlayer anzugeben
Vektor-Kachel-Schreib Layer	QgsProcessingParame		Ein Parameter für das Ziel eines Vektorkachel-Layers, der den Zielpfad für einen durch den Algorithmus erzeugten Vektorkachel-Layer angibt
Volumen	QgsProcessingParame		Ein numerischer Parameter, der ein Volumenmaß darstellt

23.10.2 Ausgabetypen

Klasse	Alg constant	Beschreibung
QgsProcessingOutputBoolean	alg.BOOL	Ein boolescher Wert
QgsProcessingOutputNumber	alg.DISTANCE	Ein numerischer Parameter für Abstandswerte
QgsProcessingOutputFile	alg.FILE	Ein Dateiname einer vorhandenen Datei
QgsProcessingOutputFolder	alg.FOLDER	Ein Ordner
QgsProcessingOutputHtml	alg.HTML	HTML
QgsProcessingOutputNumber	alg.INT	Eine Ganzzahl
QgsProcessingOutputLayerDefinition	alg.LAYERDEF	Eine Layer-Definition
QgsProcessingOutputMapLayer	alg.MAPLAYER	Ein Layer in der Karte
QgsProcessingOutputMultipleLayers	alg.MULTILAYER	Ein Satz von Layern
QgsProcessingOutputNumber	alg.NUMBER	Ein numerischer Wert
QgsProcessingOutputPointCloudLayer	alg. POINTCLOUD_LAYER	Ein Punktwolken-Layer
QgsProcessingOutputRasterLayer	alg.RASTER_LAYER	Ein Rasterlayer
QgsProcessingOutputString	alg.STRING	Eine Zeichenkette
QgsProcessingOutputVectorLayer	alg.VECTOR_LAYER	Ein Vektorlayer
QgsProcessingOutputVectorTileLayer		Ein Vektor-Kachel-Layer

23.11 Konfiguration externer Anwendungen

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.

In diesem Abschnitt werden Sie sehen, wie Sie die Verarbeitungsumgebung konfigurieren, um diese zusätzlichen Anwendungen einzubinden und Sie werden einige besondere Möglichkeiten der Algorithmen erklärt bekommen, die Sie darin finden. Haben Sie das System erst richtig konfiguriert, können Sie externe Algorithmen aus jedem Programmbestandteil, wie der Werkzeugkiste oder dem Modellentwurf heraus ausrufen, genau so, wie Sie es mit anderen Algorithmen auch machen.

Standardmäßig sind Algorithmen deaktiviert, die eine externe Anwendung voraussetzen, die nicht mit QGIS zusammen ausgeliefert wird. Sind diese jedoch auf Ihrer Maschine installiert, können Sie sie in den Einstellungen für die Verarbeitung aktivieren.

23.11.1 Ein Hinweis für Windows Anwender

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 Eine Bemerkung zu den Datenformaten

Falls Sie externe Software benutzen, bedeutet das Öffnen einer Datei in QGIS nicht, dass sie auch in der anderen Software geöffnet und prozessiert werden kann. Üblicherweise kann andere Software lesen, was Sie in QGIS geöffnet haben, aber in Einzelfällen kann das auch nicht der Fall sein. Sollten Sie Datenbanken oder ungewöhnliche Dateiformate benutzen, unabhängig davon, ob es sich um Raster- oder Vektorlayer handelt, könnten Probleme entstehen. Falls das passiert, versuchen Sie, gängige Dateiformate zu benutzen, von denen Sie wissen, dass sie von beiden Programmen verstanden werden und sehen Sie auch die Meldungen (im Protokoll) an, um herauszubekommen, was schief läuft.

Sie könnten z.B. Probleme damit bekommen und ihre Arbeit nicht fertigstellen, wenn Sie einen externen Algorithmus mit einem GRASS Rasterlayer als Eingabe aufrufen. Aus diesem Grund werden derartige Layer als nicht verfügbar für die Algorithmen angezeigt.

Mit Vektorlayern sollten Sie jedoch keine Probleme haben, da QGIS sie automatisch von ihrem originalen Dateiformat in eines konvertiert, das die externe Anwendung verarbeiten kann, bevor es den Layer dahin übergibt. Das verlängert die Dauer der Prozessierung ganz erheblich, besonders für große Layer. Wundern Sie sich also nicht, wenn die Verarbeitung eines Layers aus einer Datenbankverbindung länger dauert, als von einem Shapefile ähnlicher Größe.

Provider, die keine externe Anwendung verwenden, können jeden beliebigen Layer, den Sie in QGIS öffnen können, prozessieren, da Sie ihn für Analysen anhand von QGIS öffnen.

Alle Raster- und Vektorformate, die QGIS ausgibt, können als Eingabelayer benutzt werden. Einige Datenanbieter unterstützen bestimmte Formate nicht aber alle können in die üblichen Formate konvertieren, die später von QGIS automatisch umgewandelt werden können. Wenn Eingabelayer eine Konvertierung benötigen, kann das die Verarbeitungsdauer erhöhen.

23.11.3 Eine Bemerkung zu Vektorlayern mit ausgewählten Objekten

Externe Anwendungen können auch über die Auswahl die in Vektorlayern innerhalb von QGIS vorliegt unterrichtet werden. Das jedoch erfordert das Umschreiben aller Eingabevektorlayer, so als würden Sie ursprünglich in einem Format, das nicht von der externen Anwendung unterstützt wird, vorliegen. Nur wenn eine Auswahl besteht, oder die *Use only selected features* Option in den Verarbeitungsoptionen nicht aktiviert ist, kann ein Layer direkt an eine externe Anwendung übergeben werden.

In anderen Fällen ist es nötig, nur die ausgewählten Objekte zu exportieren, was längere Ausführungszeiten nach sich zieht.

23.11.4 Datenanbieter von Dritten benutzen

Skripte für und Bibliotheken von R

Aktivieren Sie R in der Verarbeitungsumgebung, indem Sie die Erweiterung **Processing R Provider** installieren und R für QGIS konfigurieren.

Sie konfigurieren R in der Auswahl *Datenanbieter* ► *R* des Reiters *Verarbeitung* im Menü *Einstellungen* ► *Optionen*.

Je nach dem von Ihnen verwendeten Betriebssystem, kann es sein, dass Sie in *R folder* eintragen müssen, wo die ausführbaren R-Dateien sich befinden.

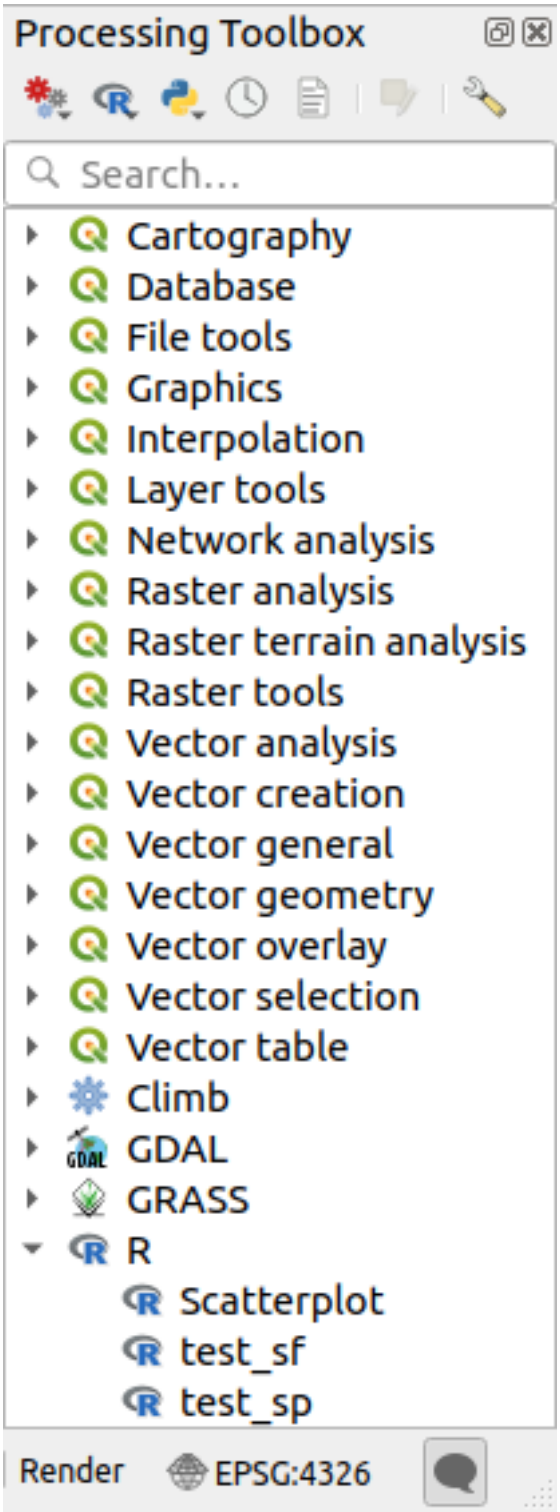
Bemerkung

Auf **Windows** ist die ausführbare Datei von R normalerweise im Verzeichnis (`R-<version>`) unter `C:\Programme\R\`. Geben Sie das Verzeichnis und **NICHT** die Binärdatei an!

Auf **Linux** müssen Sie nur sicherstellen, dass das Verzeichnis von R in der Umgebungsvariablen PATH enthalten ist. Wenn die Eingabe `R` in einem Terminal R startet, dann können Sie loslegen.

Nachdem Sie die Erweiterung **Processing R Provider** installiert haben, finden Sie einige Beispielsskripte in der *Verarbeitungs-Werkzeugkiste*:

- *Scatterplot* führt eine R-Funktion aus, die ein Streudiagramm über zwei numerische Felder des Eingabevektorlayers erzeugt.
- *test_sf* macht einige Berechnungen, die vom `sf`-Package abhängig sind und kann dafür benutzt werden, zu prüfen, ob das R-Package `sf` installiert ist. Wenn das Package nicht installiert ist, wird R versuchen, es für Sie zu installieren (und auch alle Package, von denen es abhängig ist). Dabei benutzt es das *Package repository*, das in den Verarbeitungsoptionen in *Datenanbieter* ► *R* vereinbart ist. Standardmäßig ist das https://cran.r-project.org/. Es kann sein, dass die Installation eine Weile dauert…
- *test_sp* können Sie benutzen, um zu prüfen, ob das Package `sp` installiert ist. Wenn es das nicht ist, wird R versuchen, es für Sie zu installieren.



Wenn Sie R für QGIS richtig konfiguriert haben, sollten Sie diese Skripte ausführen können.

R-Skripte aus der QGIS Collection

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).

Eine Sammlung von Beispielsskripten für R gibt es im QGIS Repository. Führen Sie folgende Schritte aus, um die Erweiterung *QGIS Resource Sharing* zu laden und zu aktivieren.

1. Fügen Sie die Erweiterung *QGIS Resource Sharing* hinzu (dabei kann es sein, dass Sie *Auch experimentelle Erweiterungen anzeigen* in den *Einstellungen* der Erweiterungen anhaken müssen)
2. Öffnen Sie sie (*Erweiterungen* ► *QGIS Resource Sharing* ► *Resource Sharing*)
3. Wählen Sie den Reiter *Settings*
4. Klicken Sie auf *Reload repositories*
5. Wählen Sie den Reiter *All*
6. Wählen Sie *QGIS R script collection* in der Liste aus und klicken Sie auf die Schaltfläche *Install*
7. Die Sammlung sollte jetzt im Reiter *Installed* aufgelistet sein
8. Schließen Sie die Erweiterung
9. Öffnen Sie die *Verarbeitungswerkzeuge*, und wenn alles funktioniert, sehen Sie die Beispielsskripte unter R in mehreren Gruppen (im folgenden Bildschirmfoto sind nur manche Gruppen ausgeklappt).

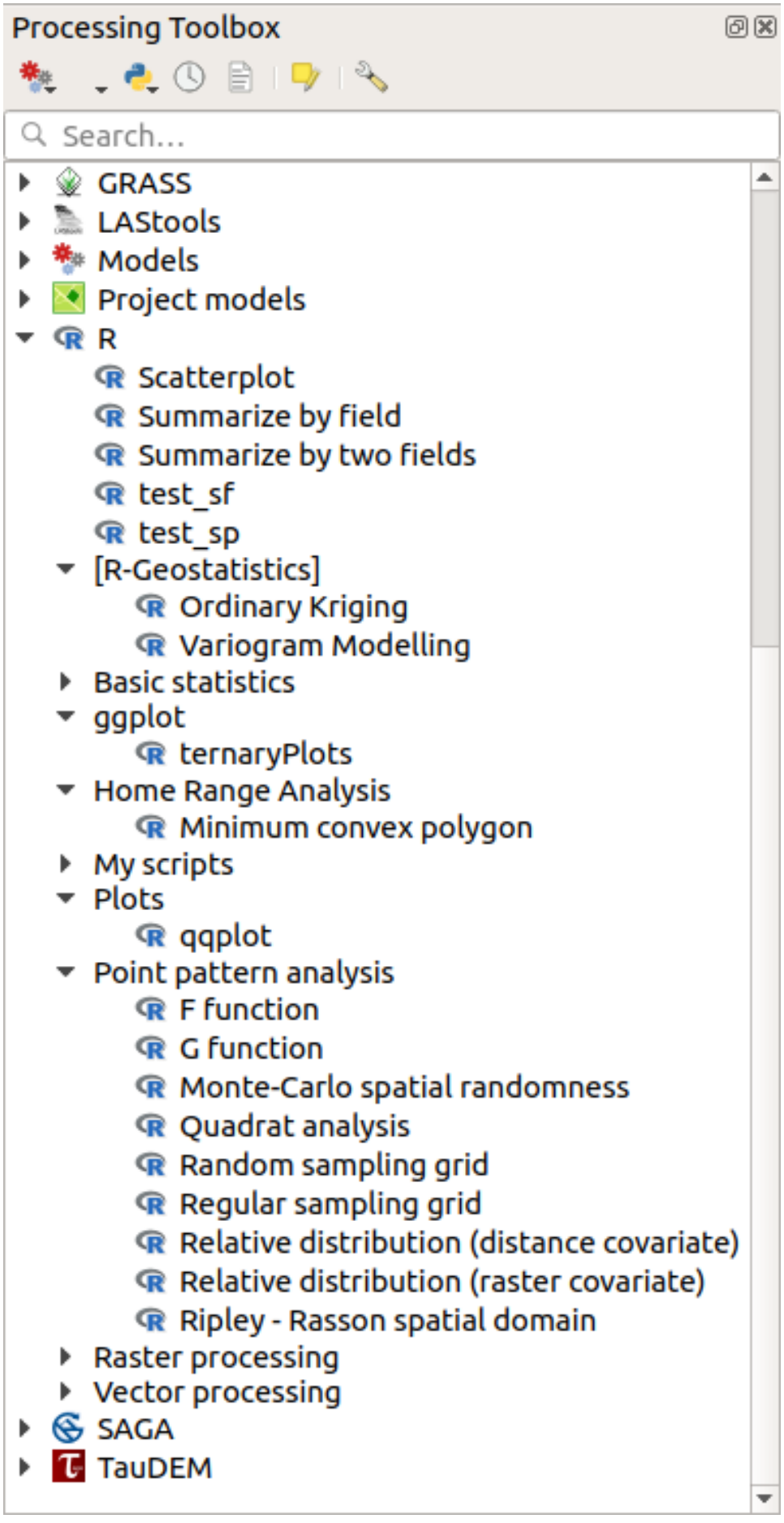


Abb. 23.35: Die *Verarbeitungswerkzeuge* mit einigen R-Skripten

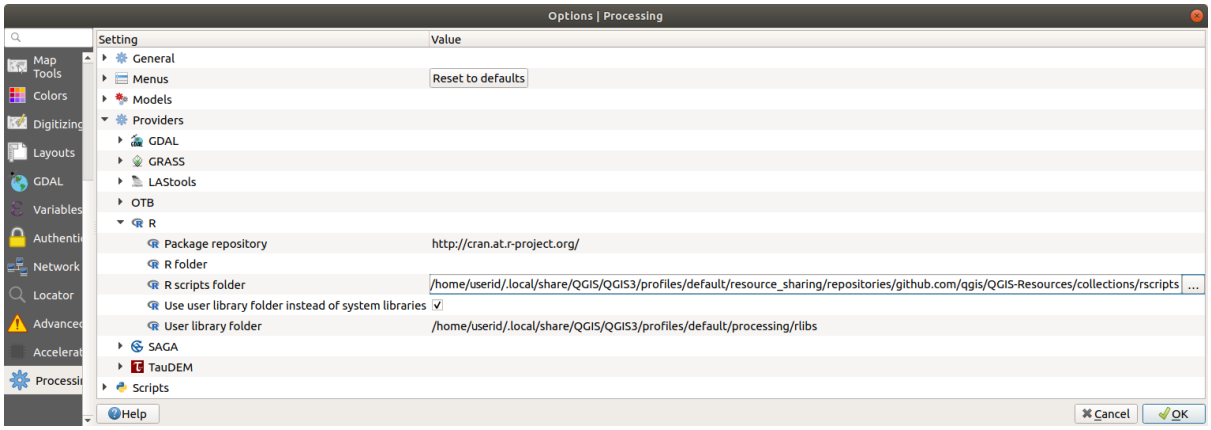
Die Skripte ganz oben sind die Beisspielsskripte der Erweiterung *Processing R Provider*.

10. Sollten die Skripte aus irgendeinem Grund in den *Verarbeitungswerkzeugen* nicht enthalten sein, können Sie folgendes versuchen:

    1. Öffnen Sie die Verarbeitungseinstellungen (*Einstellungen* ► *Optionen* ► *Verarbeitung*)
    2. Gehen Sie zu *Datenanbieter* ► *R* ► *R scripts folder*

        - Auf Ubuntu setzen Sie den Pfad auf (oder, noch besser, nehmen Sie in Path auf):

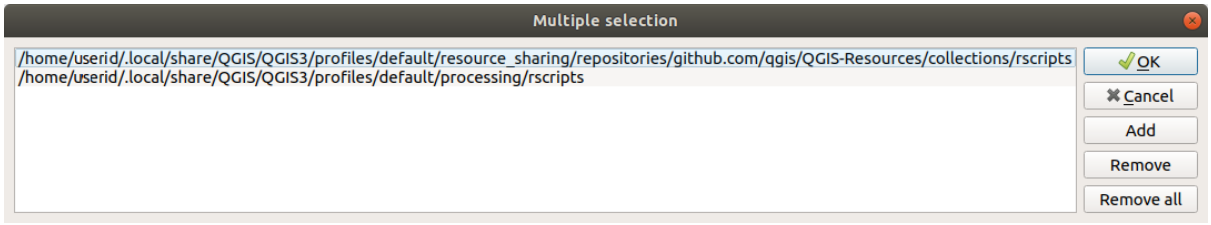
            /home/<user>/.local/share/QGIS/QGIS3/profiles/default/resource_sharing/repositories/github.com/qgis/QGIS-Resources/collections/rscripts



        - Auf Windows setzen Sie den Pfad auf (oder, noch besser, nehmen Sie in Path auf):

            C:\Users\<user>\AppData\Roaming\QGIS\QGIS3\profiles\default\resource_sharing\repositories\github.com\qgis\QGIS-Resources\collections\rscripts

    Machen Sie einen Doppelklick, wenn Sie den Eintrag editieren wollen. Dann können Sie den Pfad entweder einfügen bzw. eintippen oder Sie können sich zum Verzeichnis bewegen, indem Sie die Schaltfläche … benutzen und dann im sich öffnenden Dialog auf die Schaltfläche *Hinzufügen* klicken. Sie können hier mehrere Verzeichnisse angeben. Sie werden mit einem Semikolon („;“) getrennt.



Wenn Sie alle online verfügbaren R-Skripte für QGIS-2 haben möchten, können Sie den Eitnrag *QGIS R script collection (from QGIS 2)* anstatt des Eintrags *QGIS R script collection* auswählen. Es könnte aber sein, dass Skripte, die auf Ein- oder Ausgaben von Vektordaten beruhen, nicht funktionieren werden.

R-Skripte schreiben

Skripte werden genau so geschrieben und R-Befehle genau so aufrufen, wie in R. Dieser Abschnitt zeigt Ihnen die Syntax für R-Befehle in QGIS und wie Sie QGIS-Objekte (Layer, Tabellen) darin benutzen.

Um einen Algorithmus hinzuzufügen, der eine R-Funktion aufruft (oder ein komplexeres R-Skript, dass Sie geschrieben haben und nun in QGIS benutzen wollen), müssen Sie eine Skriptdatei anlegen, die R-Befehle ausführt.

Skriptdateien für R haben die Dateierweiterung `.rsx` und Sie können Sie sehr einfach erzeugen, auch wenn Sie nur über grundlegende Kenntnisse der Syntax und des Skriptens für R verfügen. Die Skriptdateien sollten im Verzeichnis für R-Skripte abgelegt werden. Sie legen dieses Verzeichnis (*R scripts folder*) in der Einstellungsgruppe *R* in den Einstellungen zur Verarbeitung fest.

Jetzt sehen wir uns ein ganz einfaches Beispiel einer Skriptdatei an, die die R-Methode `spsample` aufruft, um ein Zufallsraster innerhalb der Polygone eines gegebenen Polygonlayers erzeugt. Die Methode gehört zum Package `maptools`. Da beinahe alle Algorithmen, die Sie vermutlich in QGIS integrieren möchten, räumliche Daten benutzen oder erzeugen, sind Kenntnisse über diejenigen Packages, die mit räumlichen Daten arbeiten, wie `maptools` und `sp`/`sf` sehr nützlich.

```
##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)
```

Die ersten Zeilen beginnen mit einem doppelten Python-Kommentarzeichen (`##`). Sie legen den Anzeigenamen des Skripts und die Gruppe, in der es angezeigt wird fest und informieren QGIS über seine Ein- und Ausgaben.

Bemerkung

Finden Sie heraus, wie Sie eigene R-Skripte schreiben, indem Sie sich den Abschnitt R Intro im Übungshandbuch ansehen und den Abschnitt *QGIS R Syntax* zu Rate ziehen.

Legen Sie einen Eingabeparameter fest, benutzt QGIS ihn erstens, um die Benutzerschnittstelle so zu erzeugen, dass dieser Parameter vom Benutzer abgefragt wird und zweitens, um eine entsprechende R-Variable zu erzeugen, die als Eingabe in eine R-Funktion benutzt werden kann.

Im Beispiel oben haben wir eine Eingabe vom Typ `vector``festgelegt und sie ``Vector_layer` genannt. Sobald QGIS den Algorithmus ausführt, wird es den vom Nutzer ausgewählten Layer öffnen und in einer Variable namens `Vector_layer` speichern. Damit ist der Parametername der Variablenname, den Sie in R benutzen, um auf den Parameterwert zuzugreifen (deshalb sollten Sie keine reserved words von R als Parameternamen verwenden).

Parameter mit räumlichen Daten, wie Vektor- und Rasterlayer, werden mit den Befehlen `st_read()` (oder `readOGR`) und `brick()` (oder `readGDAL`) gelesen (Sie brauchen sich nicht darum zu kümmern, diese Befehle in Ihre Beschreibungsdatei aufzunehmen – QGIS macht das für Sie) und sie werden dann als `sf` (oder `Spatial*DataFrame`)-Objekte gespeichert.

Tabellenfelder werden als Zeichenkette mit dem Namen des entsprechenden Feldes gespeichert.

Vektordateien lesen Sie mit dem Befehl `readOGR()` anstatt `st_read()` ein, wenn Sie `##load_vector_using_rgdal` festlegen. Damit wird ein `Spatial*DataFrame`-Objekt anstelle eines `sf` Objekts erzeugt.

Rasterdateien lesen Sie mit dem Befehl `readGDAL()` anstatt `brick()` ein, wenn Sie `##load_raster_using_rgdal` festlegen.

Sind Sie bereits fortgeschritten und wollen nicht, dass QGIS das Objekt für den Layer erzeugt, können Sie `##pass_filenames` benutzen und damit festlegen, dass Sie eine Zeichenkette mit dem Dateinamen bevorzugen. In diesem Fall sind Sie dafür verantwortlich, die entsprechende Datei zu öffnen, bevor irgendwelche Operationen auf die enthaltenen Daten ausgeführt werden.

Mit diesen Informationen sollten Sie die ersten Zeilen des R-Skriptes verstehen (die erste Zeile, die nicht mit einem Python-Kommentarzeichen beginnt).

```
library(sp)
spatpoly = as(Vector_layer, "Spatial")
pts=spsample(polyg,numpoints,type="random")
```

Die Funktion``spsample`` wird von der *sp*-Bibliothek bereitgestellt, darum laden wir sie als erstes. Die Variable `Vector_layer` enthält ein `sf`-Objekt. Da wir eine Funktion (`spsample`) aus der *sp*-Bibliothek benutzen werden, müssen wir das `sf`-Objekt mit der *Funktion ``as`* in ein *SpatialPolygonsDataFrame*`-Objekt umwandeln.

Dann rufen wir die Funktion `spsample` mit diesem Objekt und dem Eingabeparameter `numpoints` (der angibt, wie viele Punkte erzeugt werden sollen) auf.

Da wir eine Vektorausgabe namens `Output` angegeben haben, müssen wir eine Variable `Output` erzeugen, die ein `sf`-Objekt enthält.

Das machen wir in zwei Schritten. Zuerst erzeugen wir ein `SpatialPolygonsDataFrame`-Objekt aus dem Funktionsergebnis, dabei benutzen wir die Funktion *SpatialPointsDataFrame* und dann konvertieren wir dieses Objekt in ein `sf`-Objekt und benutzen dabei die Funktion `st_as_sf` (aus der *sf*-Bibliothek).

Sie können alle Zwischenvariablen beliebig benennen. Stellen Sie nur sicher, dass die Variable mit Ihrem Endergebnis den festgelegten Namen hat (in diesem Falle `Output`) und dass sie einen geeigneten Wert enthält (ein `sf`-Objekt für die Ausgabe als Vektorlayer).

In diesem Fall musste das Ergebnis der Methode `spsample` mittels eines `SpatialPointsDataFrame`-Objekts explizit in ein `sf`-Objekt umgewandelt werden, weil es selbst ein Objekt der Klasse `ppp` ist, das nicht an QGIS zurückgegeben werden kann.

Wenn Ihr Algorithmus Rasterlayer erzeugt, hängt es beim Speichern davon ab, ob Sie die Option `##dontuserasterpackage` benutzt haben. Wenn sie sie benutzt haben, werden die Layer mit der Methode `writeGDAL()` gespeichert. Wenn nicht, wird die Methode `writeRaster()` aus dem Package `raster` benutzt.

Wenn Sie die Option `##pass_filenames` benutzt haben, werden Ausgaben mit Hilfe des Packages `raster` erzeugt (mit `writeRaster()`).

Wenn Ihr Algorithmus keinen Layer, sondern stattdessen eine Textausgabe in der Konsole erzeugt, müssen Sie dafür sorgen, dass Sie die Konsole einmal angezeigt bekommen, wenn die Ausführung beendet ist. Dafür beginnen Sie die Befehlszeilen, die das Ergebnis berechnen einfach mit dem Zeichen > (‚größer als'). Es werden nur Zeilen mit dem > am Zeilenanfang angezeigt. Hier ist z.B. die Beschreibung eines Algorithmus, der einen Test auf Normalverteilung auf ein gegebenes Feld (Spalte) der Attribute eines Vektorlayers macht:

```
##layer=vector
##field=field layer
##nortest=group
library(nortest)
>lillie.test(layer[[field]])
```

Die Ausgabe der letzten Zeile wird gedruckt, die Ausgabe der ersten aber nicht (und auch nicht die Ausgaben von anderen Kommandozeilen, die automatisch von QGIS hinzugefügt werden).

Wenn Ihr Algorithmus eine graphische Ausgabe erzeugt (mit der Methode `plot()`), fügen Sie folgende Zeile hinzu (`output_plots_to_html` war früher `showplots`):

```
##output_plots_to_html
```

Dies bewirkt, dass QGIS alle grafischen Ausgaben von R in eine temporäre Datei, die geöffnet wird nachdem die Ausführung von R beendet wurde, umleitet.

Sowohl die graphische als auch die Konsolenausgabe sind in der Ergebnisanzeige der Verarbeitung verfügbar.

Bitte konsultieren Sie die R-Skripte in der offiziellen QGIS collection (Sie haben Sie über die Erweiterung *QGIS Resource Sharing* heruntergeladen und installiert, wie es in *R-Skripte aus der QGIS Collection* erklärt wird). Die meisten sind recht einfach gestrickt und verbessern Ihr Verständnis davon, wie Sie Ihre eigenen Skripte schreiben können.

Bemerkung

Die Bibliotheken `sf`, `rgdal` und `raster` werden standardmäßig geladen, so dass Sie die entsprechenden `library()`-Befehle nicht eingeben müssen. Andere Bibliotheken, die Sie vielleicht benötigen, müssen Sie explizit mit der Eingabe: `library(ggplot2)` laden (lädt die Bibliothek `ggplot2`). Sollte das entsprechende Package auf Ihrem Rechner noch nicht installiert sein, wird Verarbeitung versuchen, es herunterzuladen und zu installieren. Damit ist das Package auch in R selbst verfügbar. **Achtung** falls das Package heruntergeladen werden muss, kann es sein, dass das Skript beim ersten Mal ziemlich lange braucht.

R-Bibliotheken, die installiert werden, wenn Sie sf_test ausführen

Das R-Skript `sp_test` versucht, die R-Packages `sp` und `raster` zu laden.

Das R-Skript *sf_test* versucht, `sf` und `raster` zu laden. Sollten diese beiden Packages nicht installiert sein, könnte R versuchen sie zu laden und zu installieren (und alle Bibliotheken, die sie selbst benötigen).

In `~/.local/share/QGIS/QGIS3/profiles/default/processing/rscripts` werden folgende R-Bibliotheken landen, nachdem Sie `sf_test` von der Verarbeitungs-Werkzeugkiste auf Ubuntu mit Version 2.0 der Erweiterung *Processing R Provider* und frisch installiertem *R* 3.4.4 (nur *apt*-Package `r-base-core`) ausgeführt haben:

```
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.

Standardmäßig versucht das Processing-Framework, seinen GRASS-Konnektor so zu konfigurieren, dass er die mit QGIS gelieferte GRASS-Distribution verwendet. Dies sollte auf den meisten Systemen problemlos funktionieren. Sollten dennoch Probleme auftreten, müssen Sie den GRASS-Konnektor möglicherweise manuell konfigurieren. Wenn Sie eine andere GRASS-Installation verwenden möchten, können Sie die Einstellung so ändern, dass sie auf den Ordner verweist, in dem die andere Version installiert ist. Damit die Algorithmen korrekt funktionieren, ist GRASS 7 erforderlich.

Wenn Sie Linux verwenden, müssen Sie lediglich sicherstellen, dass GRASS korrekt installiert ist und ohne Probleme über ein Terminalfenster ausgeführt werden kann.

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

Um LAStools in QGIS zu verwenden, müssen Sie LAStools auf Ihren Computer herunterladen und installieren und das LAStools-Plugin (verfügbar im offiziellen Repository) in QGIS installieren.

Auf Linux-Plattformen benötigen Sie Wine, um einige der Tools ausführen zu können.

LAStools wird in den Verarbeitungsoptionen aktiviert und konfiguriert (*Einstellungen* ► *Optionen*, *Verarbeitung*-Registerkarte, *Datenanbieter* ► *LAStools*), wo Sie den Speicherort von LAStools (*LAStools-Ordner*) und Wine (*Wine-Ordner*) angeben können. Unter Ubuntu ist der Standardordner für Wine `/usr/bin`.

OTB-Anwendungen

OTB-Anwendungen werden innerhalb des QGIS-Verarbeitungsframeworks vollständig unterstützt.

OTB (Orfeo ToolBox) ist eine Bildverarbeitungsbibliothek für Fernerkundungsdaten. Sie bietet auch Anwendungen, die Bildverarbeitungsfunktionen bereitstellen. Die Liste der Anwendungen und ihre Dokumentation sind im OTB CookBook verfügbar.

> **Bemerkung**
>
> Beachten Sie, dass OTB nicht mit QGIS ausgeliefert wird und separat installiert werden muss. Binärpakete für OTB finden Sie auf der Download-Seite.

So konfigurieren Sie die QGIS-Verarbeitung, um die OTB-Bibliothek zu finden:

1. Öffnen Sie die Verarbeitungseinstellungen: *Einstellungen* ► *Optionen* ► *Verarbeitung*
2. Sie finden `OTB` unter dem Menüpunkt *Datenanbieter*:
    1. Erweitern Sie den Eintrag *OTB*.
    2. Legen Sie den *OTB-Ordner* fest. Dies ist der Speicherort Ihrer OTB-Installation.
    3. Legen Sie den *OTB-Anwendungsordner* fest. Dies ist der Speicherort Ihrer OTB-Anwendungen (`1/lib/otb/applications`).
    4. Klicken Sie auf *OK*, um die Einstellungen zu speichern und das Dialogfeld zu schließen.

Wenn die Einstellungen korrekt sind, sind die OTB-Algorithmen in der *Processing Toolbox* verfügbar.

Dokumentation der OTB-Einstellungen, die in QGIS Processing verfügbar sind

- **OTB-Ordner**: Dies ist das Verzeichnis, in dem OTB verfügbar ist.
- **OTB-Anwendungsordner**: Dies ist der Speicherort der OTB-Anwendungen.

  Es sind mehrere Pfade zulässig.
- **Logger-Ebene** (optional): Ebene des Loggers, der von OTB-Anwendungen verwendet werden soll.

  Die Protokollierungsstufe steuert die Menge der während der Algorithmusausführung ausgegebenen Details. Mögliche Werte für die Protokollierungsstufe sind `INFO`, `WARNING`, `CRITICAL` und `DEBUG`. Der Standardwert ist `INFO`. Dies ist eine fortgeschrittene Benutzerkonfiguration.
- **Maximal zu verwendender Arbeitsspeicher** (optional): Standardmäßig verwenden OTB-Anwendungen den gesamten verfügbaren Arbeitsspeicher des Systems.

  Sie können OTB jedoch mit dieser Option anweisen, eine bestimmte RAM-Größe (in MB) zu verwenden. Der Wert 256 wird vom OTB-Verarbeitungsanbieter ignoriert. Dies ist eine fortgeschrittene Benutzerkonfiguration.

- **Geoid-Datei** (optional): Pfad zur Geoid-Datei.

  Diese Option legt den Wert der Parameter elev.dem.geoid und elev.geoid in OTB-Anwendungen fest. Durch die globale Festlegung dieses Werts können Benutzer ihn für mehrere Verarbeitungsalgorithmen gemeinsam nutzen. Standardmäßig leer.

- **SRTM-Kachelordner** (optional): Verzeichnis, in dem SRTM-Kacheln verfügbar sind.

  SRTM-Daten können lokal gespeichert werden, um das Herunterladen von Dateien während der Verarbeitung zu vermeiden. Diese Option legt den Wert der Parameter elev.dem.path und elev.dem in OTB-Anwendungen fest. Durch die globale Festlegung dieses Werts können Benutzer ihn für mehrere Verarbeitungsalgorithmen gemeinsam nutzen. Standardmäßig leer.

Kompatibilität und Fehlerbehebung

Ab OTB 6.6.1 sind neue Versionen von OTB mindestens mit der jeweils aktuellsten QGIS-Version kompatibel.

Wenn Sie Probleme mit OTB-Anwendungen in QGIS Processing haben, melden Sie diese bitte unter Verwendung des Labels `qgis` im OTB bug tracker.

Weitere Informationen zu OTB und QGIS finden Sie im OTB Cookbook.

KAPITEL 24

Verarbeitungsalgorithmen

Im Folgenden werden die Verarbeitungsalgorithmen (nach Anbietern) und ihre Parameter beschrieben (wie sie in der Benutzeroberfläche dargestellt werden).

24.1 QGIS Verarbeitungsalgorithmen

In den QGIS Verarbeitungsalgorithmen werden verschiedene Analyse- und Geoverarbeitungsoperationen implementiert, die meist nur die QGIS-API verwenden. Daher funktionieren fast alle diese Algorithmen „out of the box“ ohne zusätzliche Konfiguration.

Dieser Anbieter übernimmt einige Algorithmen von Plugins und fügt auch eigene Algorithmen hinzu.

24.1.1 3D Tiles

Convert B3DM to GLTF

Converts files from the legacy `.B3DM` format to `.GLTF` or `.GLB`.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input B3DM	`INPUT`	[3D Tile]	Input file to convert.
Output file	`OUTPUT`	[3D Tile] Standard: `[In temporäre Datei speichern]`	Specify the output 3D tile file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output file	`OUTPUT`	[3D Tile]	Output layer in `.GLTF` format.

Pythoncode

Algorithm ID: `native:b3dmtogltf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Convert GLTF to vector features

Converts `.GLTF` or `.GLB` file contents to standard vector layer formats.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input GLTF	`INPUT`	[3D Tile]	Input file to convert.
Output polygons Optional	`OUTPUT_POLYGON`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer to store polygon features. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Output lines Optional	`OUTPUT_LINES`	[vector: line] Standard: `[Create temporary layer]`	Specify the output vector layer to store line features. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output polygons	`OUTPUT_POLYGON`	[Vektor: Polygon]	A vector layer storing the polygon features in input layer.
Output lines	`OUTPUT_LINES`	[vector: line]	A vector layer storing the line features in input layer.

Pythoncode

Algorithm ID: `native:gltftovector`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.2 Kartografie

Punkte an Objekten ausrichten

Berechnet die erforderliche Drehung, um Punktelemente mit dem nächstgelegenen Objekt aus einem anderen Referenz-Layer auszurichten. Der Ausgabe-Layer wird um ein neues Feld erweitert, das den Winkel (in Grad, im Uhrzeigersinn) zum nächstgelegenen Referenzobjekt enthält.

Optional kann die Symbologie des Ausgabe-Layers so eingestellt werden, dass sie automatisch das berechnete Rotationsfeld verwendet, um Punktsymbole zu drehen. Bei Bedarf kann ein maximaler Abstand für die Punkte festgelegt werden, um zu vermeiden, dass isoliert liegende Punkte an weit entfernten Objekten ausgerichtet werden.

Hinweis

Dieser Algorithmus ist für Anwendungsfälle wie die Ausrichtung von Gebäudepunktsymbolen auf die nächstgelegene Straßenrichtung konzipiert.

Erlaubt auch die *Insitu-Bearbeitung* von Punkt-Objekten

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	Punkt-Objekte, für die die Drehung berechnet werden soll
Referenz-Layer	`REFERENCE_LAYE`	[Vektor: Geometrie]	Layer mit den Objekten, aus denen die Rotation berechnet werden soll

Fortsetzung auf der nächsten Seite

Tab. 24.5 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Maximal zu berücksichtigende Entfernung Optional	`MAX_DISTANCE`	[numerisch: Double] Standard: Nicht gesetzt	Wenn innerhalb dieser Entfernung kein Referenz-Objekt gefunden wird, wird dem Punkt keine Drehung zugewiesen.
Winkel-Feldname	`FIELD_NAME`	[Text] Standard: ‚rotation'	Feld, in dem der Rotationswert gespeichert wird.
Symbologie automatisch zuweisen	`APPLY_SYMBOLOG`	[Boolean] Standard: True	Dreht die Symbolmarkierung der Objekte anhand des ermittelten Winkelwerts.
Ausgerichteter Layer	`OUTPUT`	[Vektor: Punkt] Standard: `[In temporäre Datei speichern]`	Legen Sie fest, wie der Ausgabe-Layer gespeichert werden soll. *Mögliche Optionen sind*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgerichteter Layer	`OUTPUT`	[Vektor: Punkt]	Der neue Punkt-Layer mit einem Rotationsfeld. Wenn der Layer direkt in QGIS geladen wird, wird standardmäßig die Symbologie des Eingabe-Layers angewendet, mit einer datendefinierten Drehung für die Punktdarstellung.

Pythoncode

Algorithm-Kennung: `native:angletonearest`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Stildatenbanken kombinieren

Kombiniert mehrere QGIS-Stildatenbanken zu einer einzigen Stildatenbank. Wenn Elemente desselben Typs mit demselben Namen in verschiedenen Quelldatenbanken vorhanden sind, werden diese umbenannt, damit sie in der kombinierten Ausgabedatenbank eindeutige Namen haben.

Siehe auch

Create style database from project

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabe-Datenbanken	`INPUT`	[Datei] [Liste]	Dateien, die QGIS-Stilelemente enthalten
Zu kombinierende Objekte	`OBJECTS`	[Aufzählung] [Liste]	Arten von Stilelementen in den Eingabedatenbanken, die Sie in die neue Datenbank übernehmen möchten. Dies können sein: • 0 — *Symbole* • 1 — *Farbverläufe* • 2 — *Textformate* • 3 — *Beschriftungseinstellungen*
Ausgabe-Datenbank	`OUTPUT`	[Datei] Standard: `[In temporäre Datei speichern]`	Ausgabe: `.XML`-Datei, die die ausgewählten Stilelemente kombiniert. *Mögliche Optionen sind*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Anzahl der Farbverläufe	`COLORRAMPS`	[numerisch: Ganzzahl]	Number of color ramps
Anzahl der Beschriftungseinstel	`LABELSETTINGS`	[numerisch: Ganzzahl]	Number of label settings
Ausgabe-Datenbank	`OUTPUT`	[Datei]	Ausgabe `.XML`-Datei, die die ausgewählten Stilelemente kombiniert
Anzahl der Symbole	`SYMBOLS`	[numerisch: Ganzzahl]	Number of symbols
Anzahl der Textformate	`TEXTFORMATS`	[numerisch: Ganzzahl]	Number of text formats

Pythoncode

Algorithm-Kennung: `native:combinestyles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Create categorized renderer from styles

Sets a vector layer's renderer to a categorized renderer using matching symbols from a style database. If no style file is specified, symbols from the user's current *symbol library* are used instead.

A specified expression or field is used to create categories for the renderer. Each category is individually matched to the symbols which exist within the specified QGIS XML style database. Whenever a matching symbol name is found, the category's symbol will be set to this matched symbol.

If desired, outputs can also be tables containing lists of the categories which could not be matched to symbols, and symbols which were not matched to categories.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Vector layer to apply a categorized style to
Categorize using expression	`FIELD`	[Ausdruck]	Field or expression to categorize the features
Style database (leave blank to use saved symbols)	`STYLE`	[Datei]	File (`.XML`) containing the symbols to apply to the input layer categories. The file can be obtained from the Style Manager *Share symbols* tool. If no file is specified, QGIS local symbols library is used.
Use case-sensitive match to symbol names	`CASE_SENSITIVE`	[Boolean] Standard: falsch	If True (checked), applies a case sensitive comparison between the categories and symbols names
Ignore non-alphanumeric characters while matching	`TOLERANT`	[Boolean] Standard: falsch	If True (checked), non-alphanumeric characters in the categories and symbols names will be ignored, allowing greater tolerance during the match.
Non-matching categories Optional	`NON_MATCHING_C`	[vector: table] Standard: `[Ausgabe überspringen]`	Output table for categories which do not match any symbol in the database. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortsetzung auf der nächsten Seite

Tab. 24.9 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Non-matching symbol names Optional	`NON_MATCHING_S`	[vector: table] Standard: `[Ausgabe überspringen]`	Output table for symbols from the provided style database which do not match any category. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Non-matching categories	`NON_MATCHING_C`	[vector: table]	Lists categories which could not be matched to any symbol in the provided style database
Non-matching symbol names	`NON_MATCHING_S`	[vector: table]	Lists symbols from the provided style database which could not match any category
Categorized layer	`OUTPUT`	[wie Eingabe]	The input vector layer with the categorized style applied. No new layer is output.

Pythoncode

Algorithm ID: `native:categorizeusingstyle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Create style database from project

Extracts all style objects (symbols, color ramps, text formats and label settings) from a QGIS project.

The extracted symbols are saved to a QGIS style database (`XML` format), which can be managed and imported via the *Style Manager* dialog.

Siehe auch

Stildatenbanken kombinieren

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input project (leave blank to use current) Optional	`INPUT`	[Datei]	A QGIS project file to extract the style items from
Objects to extract	`OBJECTS`	[Aufzählung] [Liste]	Types of style items in the input project you would like to put in the new database. These can be: • 0 — *Symbole* • 1 — *Farbverläufe* • 2 — *Textformate* • 3 — *Beschriftungseinstellungen*
Ausgabe-Datenbank	`OUTPUT`	[Datei] Standard: `[In temporäre Datei speichern]`	Specify the output `.XML` file for the selected style items. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Anzahl der Farbverläufe	`COLORRAMPS`	[numerisch: Ganzzahl]	Number of color ramps
Anzahl der Beschriftungseinstel	`LABELSETTINGS`	[numerisch: Ganzzahl]	Number of label settings
Ausgabe-Datenbank	`OUTPUT`	[Datei]	Output `.XML` file for the selected style items
Anzahl der Symbole	`SYMBOLS`	[numerisch: Ganzzahl]	Number of symbols
Anzahl der Textformate	`TEXTFORMATS`	[numerisch: Ganzzahl]	Number of text formats

Pythoncode

Algorithm ID: `native:stylefromproject`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Export atlas layout as image

Exports the atlas of a print layout as image files (e.g. PNG or JPEG images).

If a coverage layer is set, the selected layout's atlas settings exposed in this algorithm will be overwritten. In this case, an empty filter or sort by expression will turn those settings off.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Atlas layout	`LAYOUT`	[layout]	Layout to export
Coverage layer Optional	`COVERAGE_LAYER`	[Vektor: Geometrie]	Layer to use to generate the atlas
Filterausdruck	`FILTER_EXPRESS`	[Ausdruck]	Expression to use to filter out atlas features
Sort expression Optional	`SORTBY_EXPRESS`	[Ausdruck]	Expression to use to sort the atlas features
Reverse sort order	`SORTBY_REVERSE`	[Boolean] Standard: falsch	Determines if sorting should be inverted. Used when a sort expression is provided.
Output filename expression	`FILENAME_EXPRE`	[Ausdruck] Default: ‚output_'‖@atlas_feat	Expression for use to generate filenames
Output folder	`FOLDER`	[Verzeichnis]	Destination folder where the images will be generated

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Map layers to assign to unlocked map item(s) Optional	`LAYERS`	[enumeration] [layer]	Layers to display in the map item(s) whose contents are not locked
Image format	`EXTENSION`	[Aufzählung] Default: png	File format of the generated output(s). The list of available formats varies depending on OS and installed drivers.
DPI Optional	`DPI` Standard: Nicht gesetzt	[numerisch: Double]	DPI of the output file(s). If not set, the value in the print layout settings will be used.
Generate world file	`GEOREFERENCE`	[Boolean] Standard: True	Determines if a world file should be generated
Export RDF metadata	`INCLUDE_METADA`	[Boolean] Standard: True	Determines if RDF metadata (title, author, …) should be generated
Enable antialiasing	`ANTIALIAS`	[Boolean] Standard: True	Determines if antialiasing should be enabled

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Image file	`OUTPUT`	[Datei]	Image files generated by the atlas layout

Pythoncode

Algorithm ID: `native:atlaslayouttoimage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Export atlas layout as PDF (multiple files)

Exports the atlas of a print layout to multiple PDF files.

If a coverage layer is set, the selected layout's atlas settings exposed in this algorithm will be overwritten. In this case, an empty filter or sort by expression will turn those settings off.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Atlas layout	`LAYOUT`	[layout]	Layout to export
Coverage layer Optional	`COVERAGE_LAYER`	[Vektor: Geometrie]	Layer to use to generate the atlas
Filterausdruck	`FILTER_EXPRESS`	[Ausdruck]	Expression to use to filter out atlas features
Sort expression Optional	`SORTBY_EXPRESS`	[Ausdruck]	Expression to use to sort the atlas features
Reverse sort order	`SORTBY_REVERSE`	[Boolean] Standard: falsch	Determines if sorting should be inverted. Used when a sort expression is provided.
Output filename Optional	`OUTPUT_FILENAM`	[Ausdruck]	Name pattern of the PDF output files.
Output folder	`OUTPUT_FOLDER`	[Verzeichnis]	Destination folder for the output PDF files.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Map layers to assign to unlocked map item(s) Optional	`LAYERS`	[enumeration] [layer]	Layers to display in the map item(s) whose contents are not locked
DPI Optional	`DPI` Standard: Nicht gesetzt	[numerisch: Double]	DPI of the output file(s). If not set, the value in the print layout settings will be used.
Always export as vectors	`FORCE_VECTOR`	[Boolean] Standard: falsch	Determines if vectorial data should be left as vectors
Always export as raster	`FORCE_RASTER`	[Boolean] Standard: falsch	Forces all the items in the map to be rasterized. This parameter takes precedence over the `FORCE_VECTOR` parameter.
Append georeference information	`GEOREFERENCE`	[Boolean] Standard: True	Determines if a world file should be generated
Export RDF metadata	`INCLUDE_METADA`	[Boolean] Standard: True	Determines if RDF metadata (title, author, …) should be generated
Disable tiled raster layer exports	`DISABLE_TILED`	[Boolean] Standard: falsch	Determines if raster should be tiled
Simplify geometries to reduce output file size	`SIMPLIFY`	[Boolean] Standard: True	Determines if geometries should be simplified to reduce output file size
Text export	`TEXT_FORMAT`	[Aufzählung] Standard: 0	Determines if text should be exported as path or text objects. *Possible options* are: • 0 - Always export text as paths (recommended) • 1 - Always export texts as text objects • 2 - Prefer exporting texts as text objects
Image compression	`IMAGE_COMPRESS`	[Aufzählung] Standard: 0	Determines compression level of the image and how suitable the file could be for printing outputs or post-production in external applications. Possible options are: • 0 - Lossy (JPEG) • 1 - Lossless

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
PDF file	`OUTPUT`	[Datei]	PDF file corresponding to the exported atlas layout

Pythoncode

Algorithm ID: `native:atlaslayouttomultiplepdf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Export atlas layout as PDF (single file)

Exports the atlas of a print layout as a single PDF file.

If a coverage layer is set, the selected layout's atlas settings exposed in this algorithm will be overwritten. In this case, an empty filter or sort by expression will turn those settings off.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Atlas layout	`LAYOUT`	[layout]	Layout to export
Coverage layer Optional	`COVERAGE_LAYER`	[Vektor: Geometrie]	Layer to use to generate the atlas
Filterausdruck	`FILTER_EXPRESS`	[Ausdruck]	Expression to use to filter out atlas features
Sort expression Optional	`SORTBY_EXPRESS`	[Ausdruck]	Expression to use to sort the atlas features
Reverse sort order	`SORTBY_REVERSE`	[Boolean] Standard: falsch	Determines if sorting should be inverted. Used when a sort expression is provided.
PDF file	`OUTPUT`	[Datei] Default: [Save to temporary file]	Name (including path) of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Map layers to assign to unlocked map item(s) Optional	`LAYERS`	[enumeration] [layer]	Layers to display in the map item(s) whose contents are not locked
DPI Optional	`DPI` Standard: Nicht gesetzt	[numerisch: Double]	DPI of the output file(s). If not set, the value in the print layout settings will be used.
Always export as vectors	`FORCE_VECTOR`	[Boolean] Standard: falsch	Determines if vectorial data should be left as vectors

Fortsetzung auf der nächsten Seite

Tab. 24.20 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Always export as raster	FORCE_RASTER	[Boolean] Standard: falsch	Forces all the items in the map to be rasterized. This parameter takes precedence over the FORCE_VECTOR parameter.
Append georeference information	GEOREFERENCE	[Boolean] Standard: True	Determines if a world file should be generated
Export RDF metadata	INCLUDE_METADA	[Boolean] Standard: True	Determines if RDF metadata (title, author, …) should be generated
Disable tiled raster layer exports	DISABLE_TILED	[Boolean] Standard: falsch	Determines if raster should be tiled
Simplify geometries to reduce output file size	SIMPLIFY	[Boolean] Standard: True	Determines if geometries should be simplified to reduce output file size
Text export	TEXT_FORMAT	[Aufzählung] Standard: 0	Determines if text should be exported as path or text objects. *Possible options* are: • 0 - Always export text as paths (recommended) • 1 - Always export texts as text objects • 2 - Prefer exporting texts as text objects
Image compression	IMAGE_COMPRESS	[Aufzählung] Standard: 0	Determines compression level of the image and how suitable the file could be for printing outputs or post-production in external applications. Possible options are: • 0 - Lossy (JPEG) • 1 - Lossless

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
PDF file	OUTPUT	[Datei]	PDF file corresponding to the exported atlas layout

Pythoncode

Algorithm ID: `native:atlaslayouttopdf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Export print layout as image

Exports a print layout as an image file (e.g. PNG or JPEG images)

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Print layout	`LAYOUT`	[layout]	Layout to export
Image file	`OUTPUT`	[Datei] Default: [Save to temporary file]	Name (including path) of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Map layers to assign to unlocked map item(s) Optional	`LAYERS`	[enumeration] [layer]	Layers to display in the map item(s) whose contents are not locked
DPI Optional	`DPI` Standard: Nicht gesetzt	[numerisch: Double]	DPI of the output file(s). If not set, the value in the print layout settings will be used.
Generate world file	`GEOREFERENCE`	[Boolean] Standard: True	Determines if a world file should be generated
Export RDF metadata	`INCLUDE_METADA`	[Boolean] Standard: True	Determines if RDF metadata (title, author, …) should be generated
Enable antialiasing	`ANTIALIAS`	[Boolean] Standard: True	Determines if antialiasing should be enabled

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Image file	`OUTPUT`	[Datei]	Image file corresponding to the exported print layout

Pythoncode

Algorithm ID: `native:printlayouttoimage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Export print layout as PDF

Exports a print layout as a PDF file.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Print Layout	`LAYOUT`	[layout]	Layout to export
PDF file	`OUTPUT`	[Datei] Default: [Save to temporary file]	Name (including path) of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Map layers to assign to unlocked map item(s) Optional	`LAYERS`	[enumeration] [layer]	Layers to display in the map item(s) whose contents are not locked
DPI Optional	`DPI` Standard: Nicht gesetzt	[numerisch: Double]	DPI of the output file(s). If not set, the value in the print layout settings will be used.
Always export as vectors	`FORCE_VECTOR`	[Boolean] Standard: falsch	Determines if vectorial data should be left as vectors
Always export as raster	`FORCE_RASTER`	[Boolean] Standard: falsch	Forces all the items in the map to be rasterized. This parameter takes precedence over the `FORCE_VECTOR` parameter.
Append georeference information	`GEOREFERENCE`	[Boolean] Standard: True	Determines if a world file should be generated
Export RDF metadata	`INCLUDE_METADA`	[Boolean] Standard: True	Determines if RDF metadata (title, author, …) should be generated

Fortsetzung auf der nächsten Seite

Tab. 24.26 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Disable tiled raster layer exports	`DISABLE_TILED`	[Boolean] Standard: falsch	Determines if raster should be tiled
Simplify geometries to reduce output file size	`SIMPLIFY`	[Boolean] Standard: True	Determines if geometries should be simplified to reduce output file size
Text export	`TEXT_FORMAT`	[Aufzählung] Standard: 0	Determines if text should be exported as path or text objects. *Possible options* are: • 0 - Always export text as paths (recommended) • 1 - Always export texts as text objects • 2 - Prefer exporting texts as text objects
Image compression	`IMAGE_COMPRESS`	[Aufzählung] Standard: 0	Determines compression level of the image and how suitable the file could be for printing outputs or post-production in external applications. Possible options are: • 0 - Lossy (JPEG) • 1 - Lossless
Export layers as separate PDF files	`SEPARATE_LAYER`	[Boolean] Standard: falsch	If True, then a separate PDF file will be created per layer per map item in the layout. Additionally, separate PDF files may be created for other complex layout items, resulting in a set of PDF files which contain logical atomic components of the layout.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
PDF file	`OUTPUT`	[Datei]	PDF file(s) corresponding to the exported print layout

Pythoncode

Algorithm ID: `native:printlayouttopdf`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Extract labels

Extracts label information from a rendered map at a given extent and scale.

If a map theme is provided, the rendered map will match the visibility and symbology of that theme. If left blank, all visible layers from the project will be used. Extracted label information include: position (served as point geometries), the associated layer name and feature ID, label text, rotation (in degree, clockwise), multiline alignment, and font details.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Map extent	`EXTENT`	[Ausdehnung]	Extent of the map to extract the labels from Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Map scale	`SCALE`	[scale]	Extracted labels will be rendered using their properties set at this scale.
Map theme Optional	`MAP_THEME`	[maptheme]	A map theme displaying the layers to extract the labels from. If unset, labels of the currently visible layers are extracted.
Include unplaced labels	`INCLUDE_UNPLAC`	[Boolean] Standard: True	Specify whether all overlapping labels should be extracted, including the conflicting (thus unplaced) ones.

Fortsetzung auf der nächsten Seite

Tab. 24.28 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Extracted labels	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output vector layer for the extent(s). *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Map resolution (in DPI) Optional	`DPI` Default: 96.0	[numerisch: Double]	

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extracted labels	`OUTPUT`	[Vektor: Punkt]	Point vector layer representing the fetched labels. Each feature has attributes identifying its source (layer, feature ID) and the assigned labeling properties (text, font, size, rotation, …). A default style with labeling and null symbol is also applied to the layer. **Warnung** Because some of the generated fields have name with more than 10 characters, using the ESRI shapefile format (`.SHP`) to store the output may lead to unexpected rendering while loading the layer in QGIS.

Pythoncode

Algorithm ID: `native:extractlabels`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Print layout map extent to layer

Creates a polygon layer containing the extent of a print layout map item (or items), with attributes specifying the map size (in layout units, i.e. the *reference map* units), scale and rotation.

If the map item parameter is specified, then only the matching map extent will be exported. If it is not specified, all map extents from the layout will be exported.

Optionally, a specific output CRS can be specified. If it is not specified, the original map item CRS will be used.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Print layout	`LAYOUT`	[Aufzählung]	A print layout in the current project
Map item Optional	`MAP`	[Aufzählung] Default: *All the map items*	The map item(s) whose information you want to extract. If none is provided then all the map items are processed.
Extent	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer for the extent(s). *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Overrride CRS Optional	`CRS`	[KBS] Default: *The layout CRS*	Select the CRS for the layer in which the information will be reported.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Map height	`HEIGHT`	[numerisch: Double]	
Extent	`OUTPUT`	[Vektor: Polygon]	Output polygon vector layer containing extents of all the input layout map item(s)
Map rotation	`ROTATION`	[numerisch: Double]	
Map scale	`SCALE`	[numerisch: Double]	
Map width	`WIDTH`	[numerisch: Double]	

Pythoncode

Algorithm ID: `native:printlayoutmapextenttolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Set layer style

Applies a provided style to a layer. The style must be defined in a `QML` file.

No new output are created: the style is immediately assigned to the layer.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[layer]	Input layer you want to apply the style to
Style file	`STYLE`	[Datei]	Path to the `.qml` file of the style

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
	`OUTPUT`	[wie Eingabe]	The input layer with the new style assigned. No new layer is created.

Pythoncode

Algorithm ID: `native:setlayerstyle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Topological coloring

Assigns a color index to polygon features in such a way that no adjacent polygons share the same color index, whilst minimizing the number of colors required.

The algorithm allows choice of method to use when assigning colors.

A minimum number of colors can be specified if desired. The color index is saved to a new attribute named **color_id**.

The following example shows the algorithm with four different colors chosen; as you can see each color class has the same amount of features.

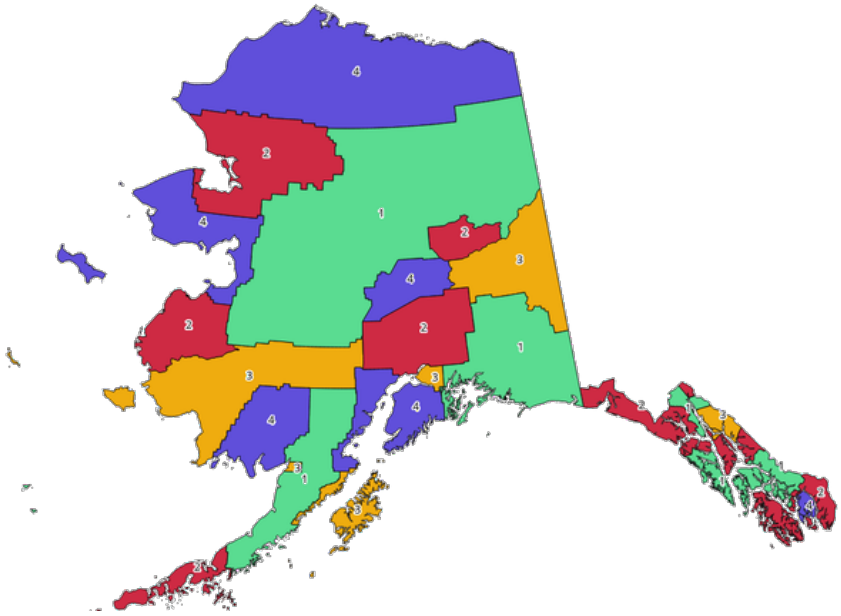


Abb. 24.1: Topological colors example

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	The input polygon layer
Minimum number of colors	`MIN_COLORS`	[numerisch: Ganzzahl] Default: 4	The minimum number of colors to assign. Minimum 1, maximum 1000.
Minimum distance between features	`MIN_DISTANCE`	[numerisch: Double] Default: 0.0	Prevent nearby (but non-touching) features from being assigned equal colors. Minimum 0.0.

Fortsetzung auf der nächsten Seite

Tab. 24.36 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Balance color assignment	`BALANCE`	[Aufzählung] Standard: 0	Options are: • 0 — By feature count Attempts to assign colors so that the count of features assigned to each individual color index is balanced. • 1 — By assigned area Assigns colors so that the total area of features assigned to each color is balanced. This mode can be useful to help avoid large features resulting in one of the colors appearing more dominant on a colored map. • 2 — By distance between colors Assigns colors in order to maximize the distance between features of the same color. This mode helps to create a more uniform distribution of colors across a map.
Colored	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Colored	`OUTPUT`	[Vektor: Polygon]	Polygon vector layer with an added `color_id` column

Pythoncode

Algorithm ID: `qgis:topologicalcoloring`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

Transfer annotations from main layer

Transfers all *annotations* from the main annotation layer in a project to a new annotation layer. Items placement can then be adjusted within the layer stack.

Parameter

Bezeichnung	Name	Typ	Beschreibung
New layer name	`LAYER_NAME`	[Text] Default: ‚Annotations'	Name of the annotations layer to create

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
New layer name	`OUTPUT`	[layer]	A layer with items from the main annotation layer

Pythoncode

Algorithm ID: `native:transferannotationsfrommain`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungs-Algorithmen über die Python-Konsole.

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.

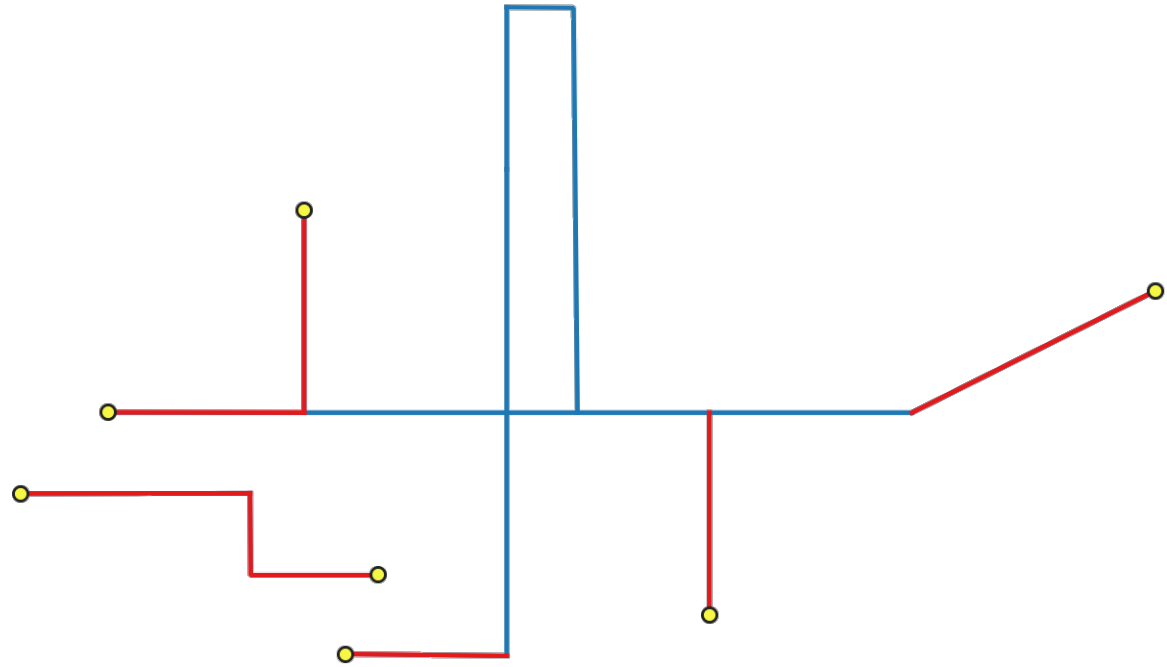


Abb. 24.2: Identifying dangle-end features (in red) and vertices (yellow).

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.
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.
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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Bemerkung

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.
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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Siehe auch

Delete duplicated vertices, *Remove duplicate vertices*

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

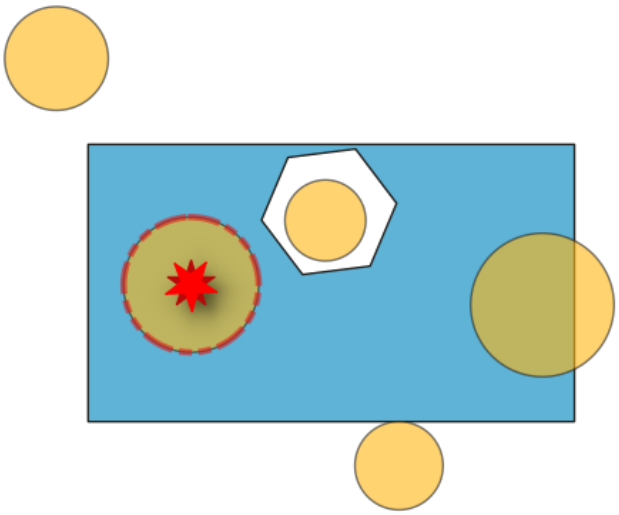


Abb. 24.3: Reporting errors on features inside a polygon.

Siehe auch

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.

Fortsetzung auf der nächsten Seite

Tab. 24.52 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* for details on how to run processing algorithms from the Python console.

Holes

Detects holes in polygon geometries and reports them as errors.

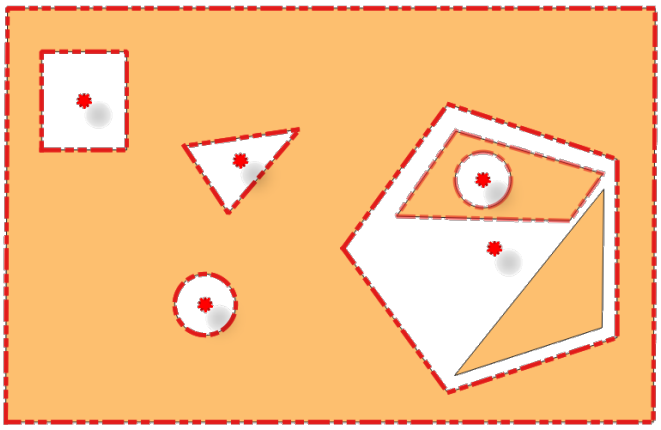


Abb. 24.4: Reporting errors for holes on a polygon feature.

Siehe auch

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

Fortsetzung auf der nächsten Seite

Tab. 24.55 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Siehe auch

Line intersections, *Intersection*, *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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Siehe auch

Line intersections, *Intersection*, *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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

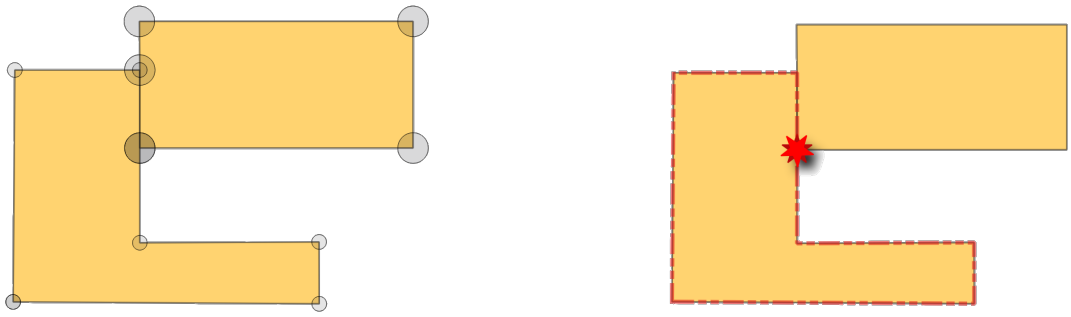


Abb. 24.5: Reporting errors for missing vertices on polygon features.

Siehe auch

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

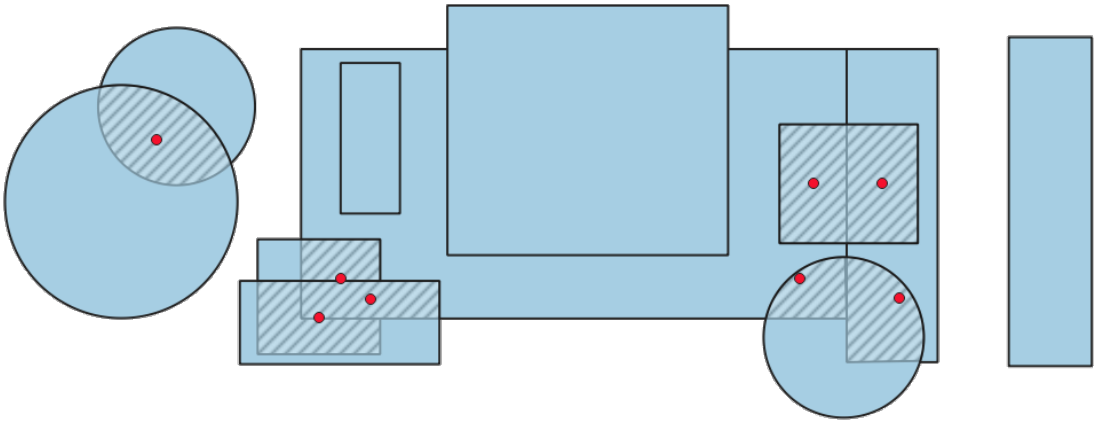


Abb. 24.6: Hashed polygons indicate overlapping areas smaller than the specified minimum.

Siehe auch

Delete overlaps, *Validate coverage*

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.

Fortsetzung auf der nächsten Seite

Tab. 24.67 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

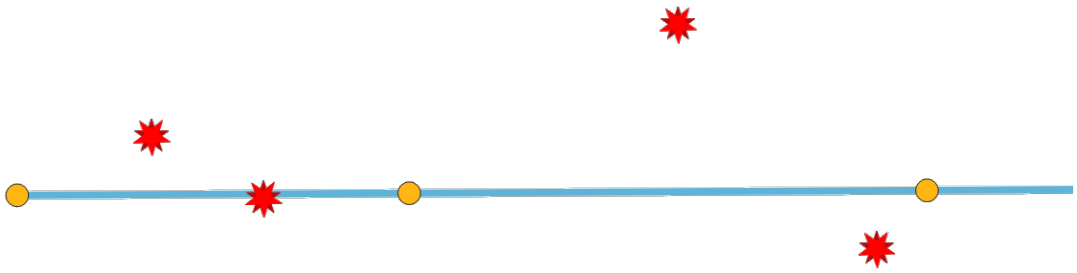


Abb. 24.7: Reporting errors on points not covered by a line.

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

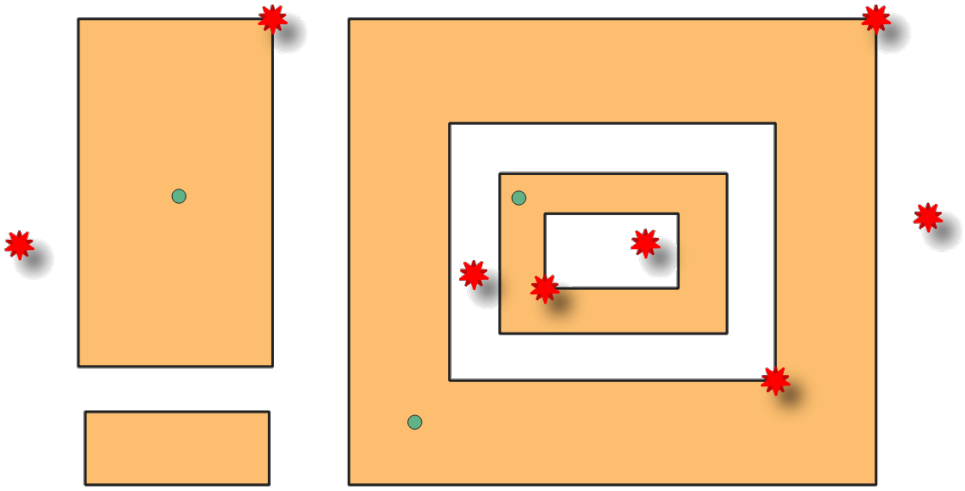


Abb. 24.8: Reporting errors on points outside polygons.

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Bemerkung

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)`.

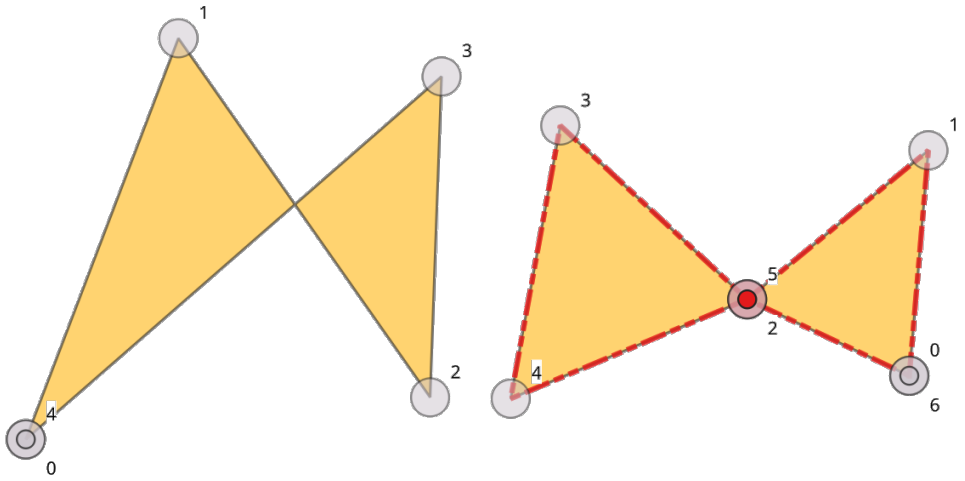


Abb. 24.9: Self-intersection vs self-contact.

Siehe auch

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.

Fortsetzung auf der nächsten Seite

Tab. 24.76 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Bemerkung

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).

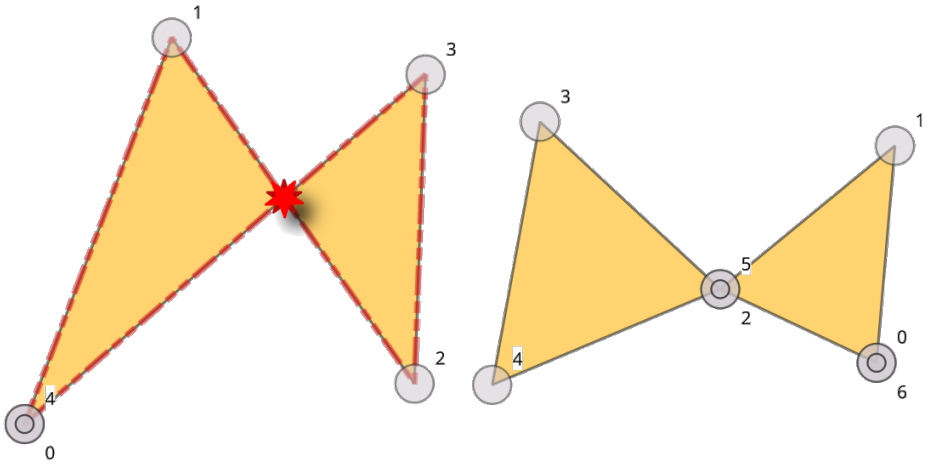


Abb. 24.10: Self-intersection vs self-contact.

Siehe auch

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Bemerkung

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)`.

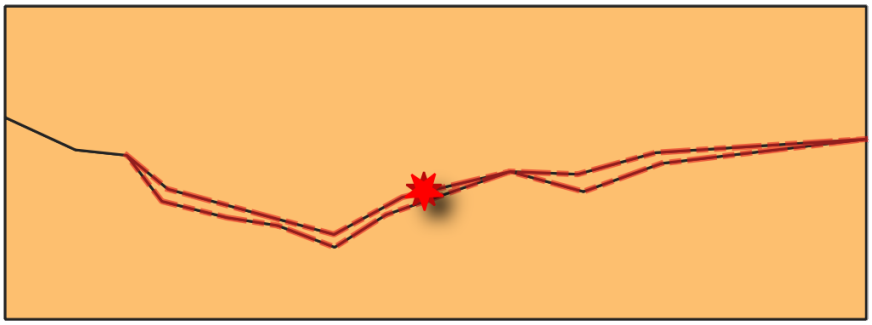


Abb. 24.11: Reporting errors on features with sliver polygons.

Siehe auch

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

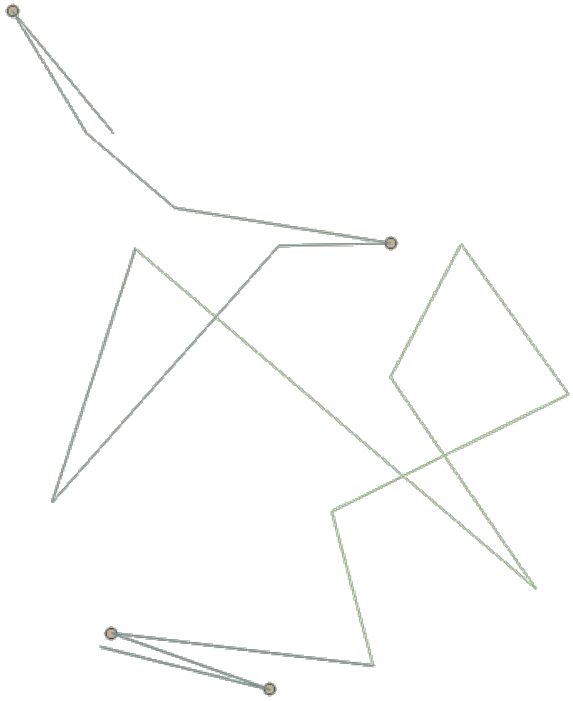


Abb. 24.12: Reporting errors on a line feature for angles lower than 15°.

Siehe auch

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.

Fortsetzung auf der nächsten Seite

Tab. 24.85 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Bemerkung

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)`.

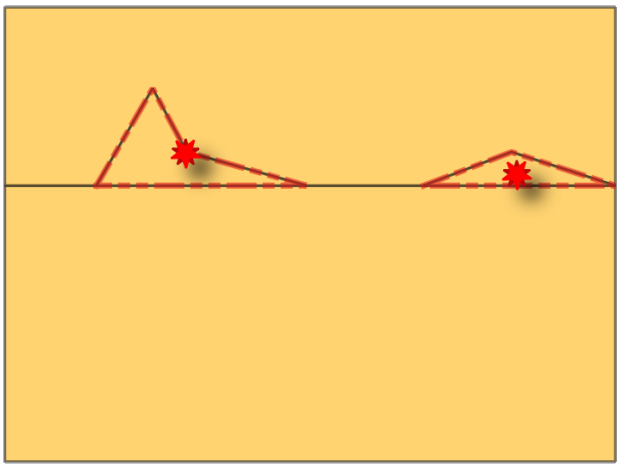


Abb. 24.13: Reporting errors on features with small polygons (below given area threshold).

Siehe auch

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

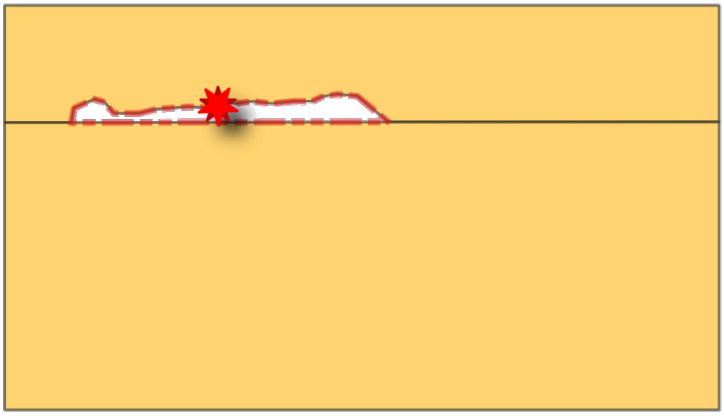


Abb. 24.14: Reporting errors on polygon features for gaps smaller than the specified threshold.

Siehe auch

Fill gaps, *Validate coverage*

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.

Fortsetzung auf der nächsten Seite

Tab. 24.91 – Fortsetzung der vorherigen Seite

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.

Fortsetzung auf der nächsten Seite

Tab. 24.93 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

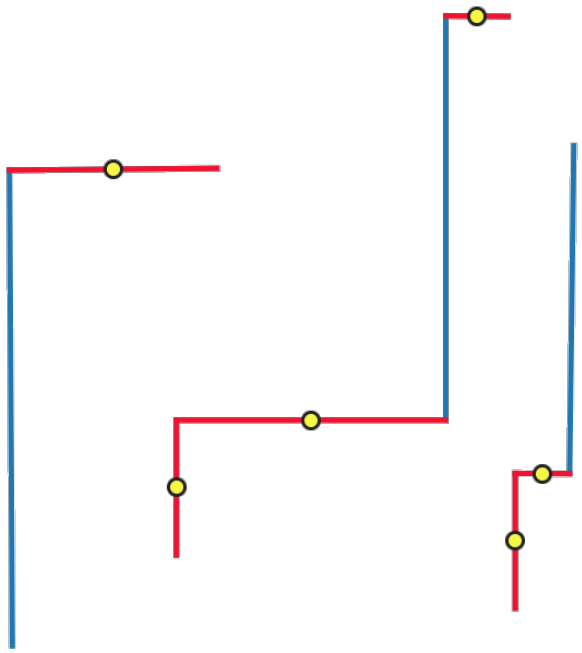


Abb. 24.15: Reporting errors (in red) on single line segments shorter than the given threshold.

Siehe auch

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.

Fortsetzung auf der nächsten Seite

Tab. 24.94 – Fortsetzung der vorherigen Seite

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.

Fortsetzung auf der nächsten Seite

Tab. 24.96 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Siehe auch

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.

Fortsetzung auf der nächsten Seite

Tab. 24.97 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* for details on how to run processing algorithms from the Python console.

24.1.4 Datenbank

Export to PostgreSQL

Exports a vector layer to a PostgreSQL database, creating a new relation. If a relation with the same name exists, it can be removed before the new relation is created. Prior to this a connection between QGIS and the PostgreSQL database has to be created (see eg *Erstellen einer gespeicherten Verbindung*).

Parameter

Bezeichnung	Name	Typ	Beschreibung
Layer to import	`INPUT`	[Vektor: alle]	Vector layer to add to the database
Database (connection name)	`DATABASE`	[Text]	Name of the database connection (not the database name). Existing connections will be shown in the combobox.
Schema (schema name) Optional	`SCHEMA`	[Text] Default: ‚public‘	Name of the schema to store the data. It can be a new one or already exist.
Table to import to (leave blank to use layer name) Optional	`TABLENAME`	[Text] Default: ‚‘	Defines a table name for the imported vector file. If nothing is added, the layer name will be used.
Primary key field Optional	`PRIMARY_KEY`	[tablefield: any]	Sets the primary key field from an existing field in the vector layer. A column with **unique** values can be used as Primary key for the database.
Geometry column	`GEOMETRY_COLUM`	[Text] Default: ‚geom‘	Defines the name of the geometry column in the new PostgreSQL table. Geometry information for the features is stored in this column.
Encoding Optional	`ENCODING`	[Text] Default: ‚UTF-8‘	Defines the encoding of the output layer
Overwrite	`OVERWRITE`	[Boolean] Standard: True	If the specified table exists, setting this option to `True` will make sure that it is deleted and a new table will be created before the features are added. If this option is `False` and the table exists, the algorithm will throw an exception („relation already exists“).
Create spatial index	`CREATEINDEX`	[Boolean] Standard: True	Specifies whether to create a spatial index or not

Fortsetzung auf der nächsten Seite

Tab. 24.100 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Convert field names to lowercase	LOWERCASE_NAME	[Boolean] Standard: True	Converts the field names of the input vector layer to lowercase
Drop length constraint on character fields	DROP_STRING_LE	[Boolean] Standard: falsch	Should length constraints on character fields be dropped or not
Create single-part geometries instead of multi-part	FORCE_SINGLEPA	[Boolean] Standard: falsch	Should the features of the output layer be single-part instead of multi-part. By default the existing geometries information are preserved.

Ausgaben

The algorithm has no output.

Pythoncode

Algorithm ID: `native:importintopostgis`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export to SpatiaLite

Exports a vector layer to a SpatiaLite database. Prior to this a connection between QGIS and the SpatiaLite database has to be created (see eg *SpatiaLite-Layer*).

Parameter

Bezeichnung	Name	Typ	Beschreibung
Layer to import	INPUT	[Vektor: alle]	Vector layer to add to the database
File database	DATABASE	[Vektor: alle]	The SQLite/SpatiaLite database file to connect to
Table to import to (leave blank to use layer name) Optional	TABLENAME	[Text] Default: ‚‘	Defines the table name for the imported vector file. If nothing is specified, the layer name will be used.
Primary key field Optional	PRIMARY_KEY	[tablefield: any]	Use a field in the input vector layer as the primary key
Geometry column	GEOMETRY_COLUM	[Text] Default: ‚geom‘	Defines the name of the geometry column in the new SpatiaLite table. Geometry information for the features is stored in this column.

Fortsetzung auf der nächsten Seite

Tab. 24.101 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Encoding Optional	ENCODING	[Text] Default: ‚UTF-8‘	Defines the encoding of the output layer
Overwrite	OVERWRITE	[Boolean] Standard: True	If the specified table exists, setting this option to `True` will make sure that it is deleted and a new table will be created before the features of the layer is added. If this option is `False` and the table exists, the algorithm will throw an exception („table already exists“).
Create spatial index	CREATEINDEX	[Boolean] Standard: True	Specifies whether to create a spatial index or not
Convert field names to lowercase	LOWERCASE_NAME	[Boolean] Standard: True	Convert the field names of the input vector layer to lowercase
Drop length constraint on character fields	DROP_STRING_LE	[Boolean] Standard: falsch	Should length constraints on character fields be dropped or not
Create single-part geometries instead of multi-part	FORCE_SINGLEPA	[Boolean] Standard: falsch	Should the features of the output layer be single-part instead of multi-part. By default the existing geometries information are preserved.

Ausgaben

The algorithm has no output.

Pythoncode

Algorithm ID: `native:importintospatialite`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Package layers

Adds layers to a GeoPackage.

If the GeoPackage exists and `Overwrite existing GeoPackage` is checked, it will be overwritten (removed and recreated). If the GeoPackage exists and `Overwrite existing GeoPackage` is not checked, the layer will be appended.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`LAYERS`	[vector: any] [list]	The (vector) layers to import into the GeoPackage. Raster layers are not supported. If a raster layer is added, a `QgsProcessingException` will be thrown.
Overwrite existing GeoPackage	`OVERWRITE`	[Boolean] Standard: falsch	If the specified GeoPackage exists, setting this option to `True` will make sure that it is deleted and a new one will be created before the layers are added. If set to `False`, layers will be appended.
Save layer styles into GeoPackage	`SAVE_STYLES`	[Boolean] Standard: True	Save the layer styles.
Save only selected features	`SELECTED_FEATU`	[Boolean] Standard: falsch	If a layer has a selection, setting this option to `True` will result in only selected features being saved. For layers without a selection all features will be saved.
Export related layers following relations defined in the project	`EXPORT_RELATED`	[Boolean] Standard: falsch	If an input layer has *relations* set in the project, setting this option to `True` will result in exporting also its related layer(s). If the layer has features selected, then only their related features will be exported unless the related layer was also an input layer.
Destination GeoPackage	`OUTPUT`	[Datei] Standard: `[In temporäre Datei speichern]`	Specify where to store the GeoPackage file. One of • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Layers within new package	`OUTPUT_LAYERS`	[string] [list]	The list of layers added to the GeoPackage.

Pythoncode

Algorithm ID: `native:package`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

PostgreSQL execute and load SQL

Allows a SQL database query to be performed on a PostgreSQL database connected to QGIS and loads the result. The algorithm **won't** create a new layer: it is designed to run queries on the layer itself.

Beispiel

1. Set all the values of an existing field to a fixed value. The SQL query string will be:

   ```
   UPDATE your_table SET field_to_update=20;
   ```

   In the example above, the values of the field `field_to_update` of the table `your_table` will be all set to `20`.

2. Create a new `area` column and calculate the area of each feature with the `ST_AREA` PostGIS function.

   ```
   -- 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);
   ```

Siehe auch

PostgreSQL execute SQL, *Execute SQL*, *SpatiaLite execute SQL*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Database (connection name)	`DATABASE`	[Text]	The database connection (not the database name). Existing connections will be shown in the combobox.
SQL query	`SQL`	[Text]	Defines the SQL query, for example `'UPDATE my_table SET field=10'`.
Unique ID field name	`ID_FIELD`	[Text] Default: id	Sets the primary key field (a column in the result table)
Geometry field name Optional	`GEOMETRY_FIELD`	[Text] Default: ‚geom‘	Name of the geometry column (a column in the result table)

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
SQL layer	`OUTPUT`	[Vektor: alle]	The resulting vector layer to be loaded into QGIS.

Pythoncode

Algorithm ID: `qgis:postgisexecuteandloadsql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

PostgreSQL execute SQL

Allows a SQL database query to be performed on a PostgreSQL database connected to QGIS. The algorithm **won't** create a new layer: it is designed to run queries on the layer itself.

Beispiel

1. Set all the values of an existing field to a fixed value. The SQL query string will be:

   ```
   UPDATE your_table SET field_to_update=20;
   ```

   In the example above, the values of the field `field_to_update` of the table `your_table` will be all set to `20`.

2. Create a new `area` column and calculate the area of each feature with the `ST_AREA` PostGIS function.

   ```
   -- 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);
   ```

Siehe auch

PostgreSQL execute and load SQL, *Execute SQL*, *SpatiaLite execute SQL*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Database (connection name)	`DATABASE`	[Text]	The database connection (not the database name). Existing connections will be shown in the combobox.
SQL query	`SQL`	[Text]	Defines the SQL query, for example `'UPDATE my_table SET field=10'`.

Ausgaben

No output is created. The SQL query is executed in place.

Pythoncode

Algorithm ID: `native:postgisexecutesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

SpatiaLite execute SQL

Allows a SQL database query to be performed on a SpatiaLite database. The database is determined by an input layer or file. The algorithm **won't** create a new layer: it is designed to run queries on the layer itself.

Siehe auch

PostgreSQL execute SQL, *Execute SQL*

For some SQL query examples see *PostgreSQL Query Examples*.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Database layer (or file)	`DATABASE`	[Datei]	The SQLite/SpatiaLite database file to connect to
SQL query	`SQL`	[Text] Default: ‚‘	Defines the SQL query, for example `'UPDATE my_table SET field=10'`.

Ausgaben

No output is created. The SQL query is executed in place.

Pythoncode

Algorithm ID: `native:spatialiteexecutesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

SpatiaLite execute SQL (registered DB)

Allows a SQL database query to be performed on a SpatiaLite database connected to QGIS. The algorithm **won't** create a new layer: it is designed to run queries on the layer itself.

Siehe auch

PostgreSQL execute SQL, *Execute SQL*

For some SQL query examples see *PostgreSQL Query Examples*.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Database	`DATABASE`	[Aufzählung] Default: not set	Select a SQLite/SpatiaLite database connected to the current session
SQL query	`SQL`	[Text] Default: ‚'	Defines the SQL query, for example `'UPDATE my_table SET field=10'`.

Ausgaben

No output is created. The SQL query is executed in place.

Pythoncode

Algorithm ID: `native:spatialiteexecutesqlregistered`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.5 File tools

Download file via HTTP(S)

Downloads a URL to the file system with an HTTP(S) GET or POST request.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
URL	`URL`	[Text]	The URL of the file to download.
File destination	`OUTPUT`	[Text] Standard: `[In temporäre Datei speichern]`	Specification of the file destination. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Methode	`METHOD`	[Aufzählung] Standard: 0	The HTTP method to use for the request. Options are: • 0 — GET • 1 — POST
Data Optional	`DATA`	[Text]	The data to add in the body if the request is a POST.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
File destination	`OUTPUT`	[Text]	The location of the downloaded file

Python-Code

Algorithm ID: `native:filedownloader`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

HTTP(S) POST/GET request

Performs a HTTP(S) POST/GET request and returns the HTTP status code and the reply data. If an error occurs then the error code and the message will be returned. Optionally, the result can be written to a file on the disk. By default the algorithm will warn on errors. Optionally, the algorithm can be set to treat HTTP errors as failures.

Parameter

Bezeichnung	Name	Typ	Beschreibung
URL or file path	`URL`	[Text]	The URL or path of the file to open.
Methode	`METHOD`	[Aufzählung] Standard: 0	The HTTP method to use for the request. Options are: • 0 — GET • 1 — POST
POST Data Optional	`DATA`	[Text]	The data to add in the body if the request is a POST.
Authentication Optional	`AUTH_CONFIG`	[authconfig] Default: No authentication	An authentication configuration to pass
Consider HTTP errors as failures	`FAIL_ON_ERROR`	[Boolean] Standard: falsch	If set, the algorithm will fail on encountering an HTTP error.
File destination Optional	`OUTPUT`	[Text] Default: `[Skip Output]`	The result can be written to a file instead of being returned as a string. Specification of the file destination. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
File destination	`OUTPUT`	[Datei]	The downloaded file with the returned result
HTTP Status	`STATUS_CODE`	[Text]	The HTTP Status
Network error code	`ERROR_CODE`	[Text]	The error code when it fails (like as well on 404 HTTP Status etc.)
Network error message	`ERROR_MESSAGE`	[Text]	A string containing the error message in case of failure
Result data	`RESULT_DATA`	[Text]	A string containing the result data in case of success

Python-Code

Algorithm ID: `native:httprequest`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Open file or URL

Opens files in their default associated application, or URLs in the user's default web browser.

Parameter

Bezeichnung	Name	Typ	Beschreibung
URL or file path	`URL`	[Text]	The URL or path of the file to open.

Ausgaben

The algorithm has no output.

Python-Code

Algorithm ID: `native:openurl`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

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.

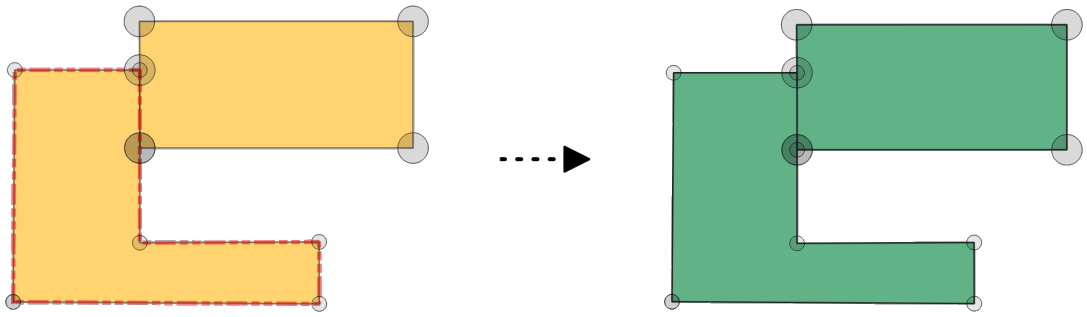


Abb. 24.16: Fixing missing vertices along polygon borders.

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.

Fortsetzung auf der nächsten Seite

Tab. 24.115 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Siehe auch

Promote to multipart

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Siehe auch

Remove duplicate vertices

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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*

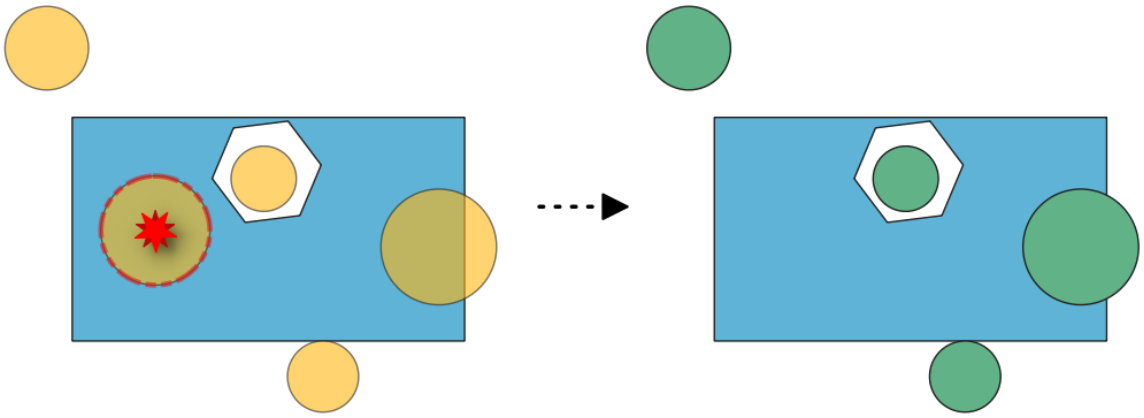


Abb. 24.17: Fixing errors from the „*Features inside polygon*“ algorithm check.

Siehe auch

Delete duplicate geometries, *Drop geometries*

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.

Fortsetzung auf der nächsten Seite

Tab. 24.124 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

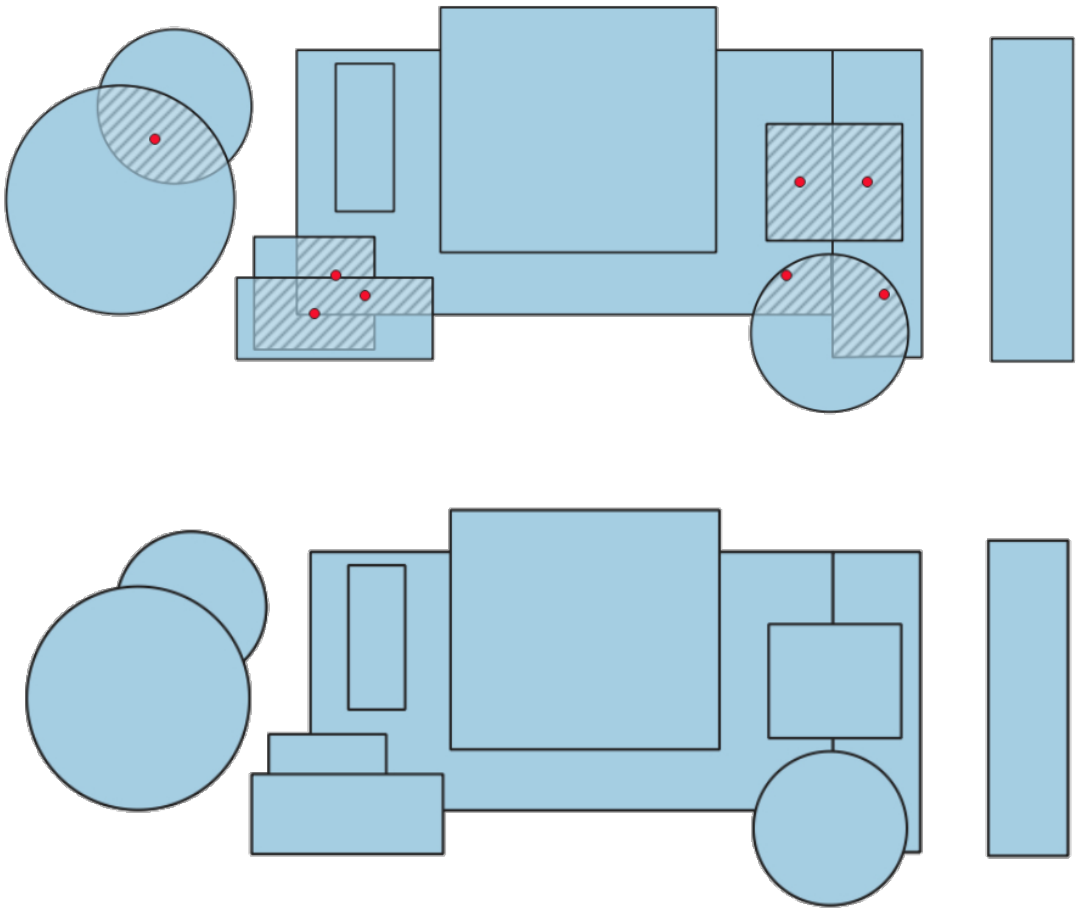


Abb. 24.18: Deleting overlapping areas.

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

> **Achtung**
>
> This algorithm removes the vertex at the reported small angles, generating new segments that may form a new small angle.

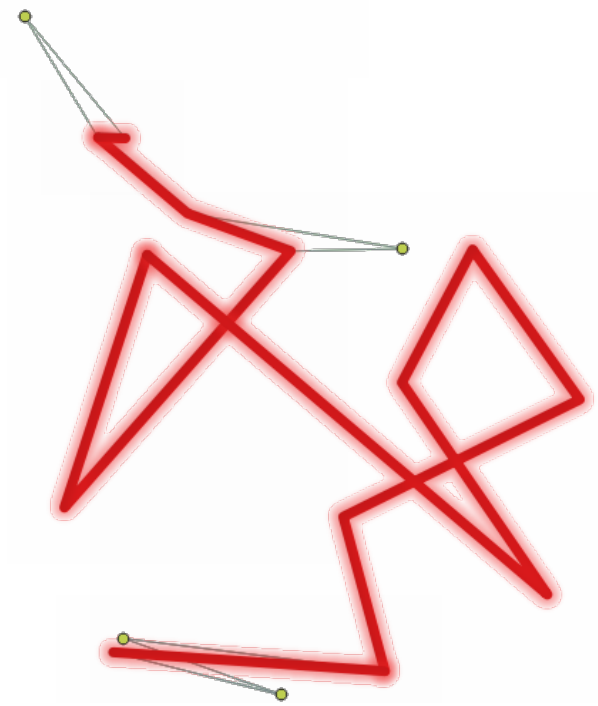


Abb. 24.19: Fixing reported errors on a line feature for angles lower than 15°.

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.

Fortsetzung auf der nächsten Seite

Tab. 24.129 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

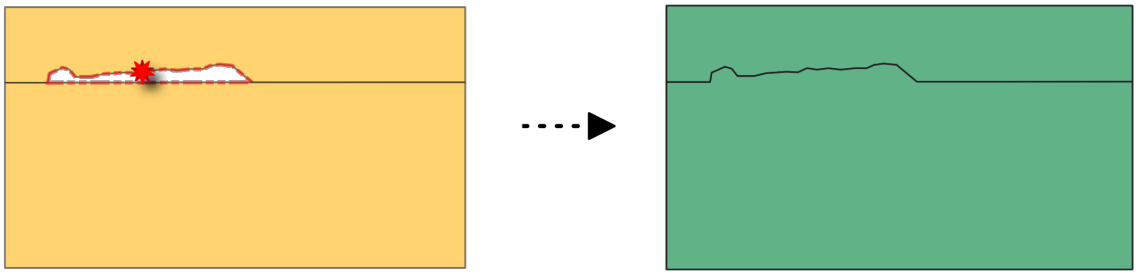


Abb. 24.20: Fixing gap between polygon features using the `Add to largest neighbouring area` method.

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.

Fortsetzung auf der nächsten Seite

Tab. 24.132 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

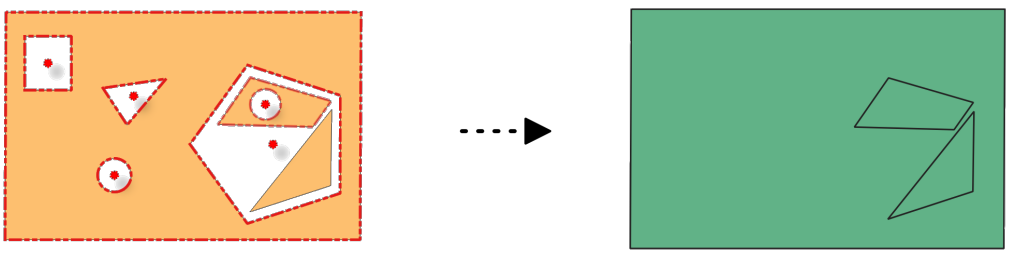


Abb. 24.21: Deleting holes in a polygon feature.

Siehe auch

Delete holes

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

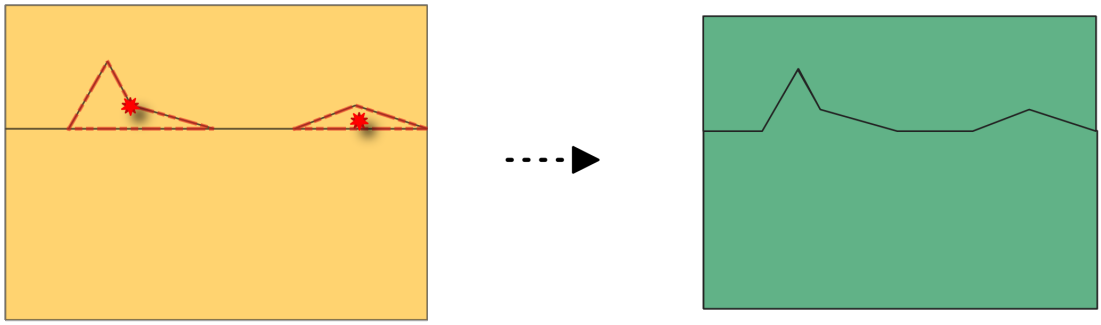


Abb. 24.22: Before and after fixing small polygons (method: merge with neighboring polygon with largest area).

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.

Fortsetzung auf der nächsten Seite

Tab. 24.138 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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.

Bemerkung

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)`.

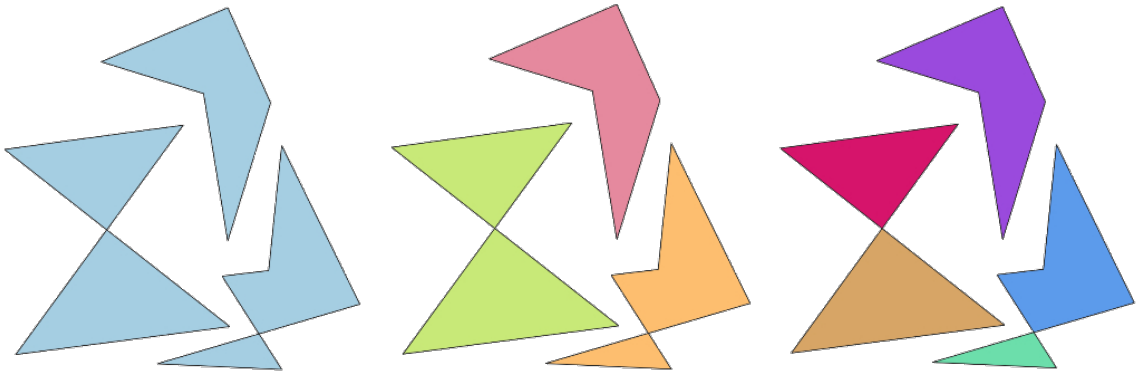


Abb. 24.23: Fixing self-intersection using various methods: as multi-part (middle), and multiple single-part geometries (right).

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

Fortsetzung auf der nächsten Seite

Tab. 24.141 – Fortsetzung der vorherigen Seite

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* for details on how to run processing algorithms from the Python console.

24.1.7 GPS

Convert GPS data

Uses the GPSBabel tool to convert a GPS data file from a range of formats to the GPX standard format.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input file	`INPUT`	[Datei]	File containing the data to convert
Format	`FORMAT`	[Aufzählung]	Format of the file to convert, from this list.
Feature type	`FEATURE_TYPE`	[Aufzählung] Standard: 0	The type of data to convert • 0 — Waypoints • 1 — Routes • 2 — Tracks
Ergebnis	`OUTPUT`	[Vektor: alle] Standard: `[In temporäre Datei speichern]`	Specification of the output GPX file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT_LAYER`	[Vektor: alle]	Output layer with data in GPX standard format

Pythoncode

Algorithm ID: `native:convertgpsdata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Convert GPX feature type

Uses the GPSBabel tool to convert GPX features from one type to another (e.g. converting all waypoint features to a route).

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input file	`INPUT`	[Datei]	File containing the data to convert
Conversion	`CONVERSION`	[Aufzählung] Standard: 0	The type of conversion to apply • 0 — Waypoints from a route • 1 — Waypoints from a track • 2 — Routes from waypoints • 3 — Tracks from waypoints
Ergebnis	`OUTPUT`	[vector: point or line] Standard: `[In temporäre Datei speichern]`	Specification of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ergebnis	`OUTPUT`	[Vektor: alle]	Output layer with converted GPX features

Pythoncode

Algorithm ID: `native:convertgpxfeaturetype`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Download GPS data from device

Uses the GPSBabel tool to download data from a GPS device into the GPX standard format.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Device	`DEVICE`	[Aufzählung] Default:*Garmin serial*	The GPS device used to create the data. Must be declared in *GPS Settings* dialog.
Port	`PORT`	[Aufzählung]	The port the device is connected to. Available ports depend on the OS.
Feature type	`FEATURE_TYPE`	[Aufzählung] Standard: 0	The type of data to convert • 0 — Waypoints • 1 — Routes • 2 — Tracks
Ergebnis	`OUTPUT`	[Vektor: alle] Standard: `[In temporäre Datei speichern]`	Specification of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT_LAYER`	[Vektor: alle]	Output layer with data in GPX standard format

Pythoncode

Algorithm ID: `native:downloadgpsdata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Upload GPS data to device

Uses the GPSBabel tool to upload data to a GPS device from the GPX standard format.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input file	`INPUT`	[Datei]	`.GPX` file containing the data to upload
Device	`DEVICE`	[Aufzählung] Default:*Garmin serial*	The GPS device you would like to upload the data to. Must be declared in *GPS Settings* dialog.
Port	`PORT`	[Aufzählung]	The port the device is connected to. Available ports depend on the OS.
Feature type	`FEATURE_TYPE`	[Aufzählung] Standard: 0	The type of data to upload • 0 — Waypoints • 1 — Routes • 2 — Tracks

Ausgaben

No output is provided. If successful, data are loaded to the device.

Pythoncode

Algorithm ID: `native:uploadgpsdata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.8 Interpolation

Heatmap (kernel density estimation)

Creates a density (heatmap) raster of an input point vector layer using kernel density estimation.

The density is calculated based on the number of points in a location, with larger numbers of clustered points resulting in larger values. Heatmaps allow easy identification of *hotspots* and clustering of points.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Point layer	`INPUT`	[Vektor: Punkt]	Point vector layer to use for the heatmap
Radius	`RADIUS`	[numerisch: Double] Default: 100.0	Heatmap search radius (or kernel bandwidth) in map units. The radius specifies the distance around a point at which the influence of the point will be felt. Larger values result in greater smoothing, but smaller values may show finer details and variation in point density.
Output raster size	`PIXEL_SIZE`	[numerisch: Double] Default: 0.1	Pixel size of the output raster layer in layer units. In the GUI, the size can be specified by the number of rows (`Number of rows`) / columns (`Number of columns`) **or** the pixel size(`Pixel Size X` / `Pixel Size Y`). Increasing the number of rows or columns will decrease the cell size and increase the file size of the output raster. The values in `Rows`, `Columns`, `Pixel Size X` and `Pixel Size Y` will be updated simultaneously - doubling the number of rows will double the number of columns, and the cell size will be halved. The extent of the output raster will remain the same (approximately).
Radius from field Optional	`RADIUS_FIELD`	[tablefield: numeric]	Sets the search radius for each feature from an attribute field in the input layer.
Weight from field Optional	`WEIGHT_FIELD`	[tablefield: numeric]	Allows input features to be weighted by an attribute field. This can be used to increase the influence certain features have on the resultant heatmap.
Kernel shape	`KERNEL`	[Aufzählung] Vorgabe: *0*	Controls the rate at which the influence of a point decreases as the distance from the point increases. Different kernels decay at different rates, so a triweight kernel gives features greater weight for distances closer to the point then the Epanechnikov kernel does. Consequently, triweight results in “sharper” hotspots and Epanechnikov results in “smoother” hotspots. There are many shapes available (please see the Wikipedia page for further information): • 0 — Quartic • 1 — Triangular • 2 — Uniform • 3 — Triweight • 4 — Epanechnikov

Fortsetzung auf der nächsten Seite

Tab. 24.151 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Decay ratio (Triangular kernels only) Optional	`DECAY`	[numerisch: Double] Default: *0.0*	Can be used with Triangular kernels to further control how heat from a feature decreases with distance from the feature. • A value of 0 (=minimum) indicates that the heat will be concentrated in the center of the given radius and completely extinguished at the edge. • A value of 0.5 indicates that pixels at the edge of the radius will be given half the heat as pixels at the center of the search radius. • A value of 1 means the heat is spread evenly over the whole search radius circle. (This is equivalent to the 'Uniform' kernel.) • A value greater than 1 indicates that the heat is higher towards the edge of the search radius than at the center.
Output value scaling	`OUTPUT_VALUE`	[Aufzählung] Vorgabe: *0*	Allow to change the values of the output heatmap raster. One of: • 0 — Raw • 1 — Scaled
Heatmap	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer with kernel density values. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Heatmap	`OUTPUT`	[Raster]	Raster layer with kernel density values

Example: Creating a Heatmap

For the following example, we will use the `airports` vector point layer from the QGIS sample dataset (see *Beispieldaten herunterladen*). Another excellent QGIS tutorial on making heatmaps can be found at http://qgistutorials.com.

In Abb. 24.24, the airports of Alaska are shown.

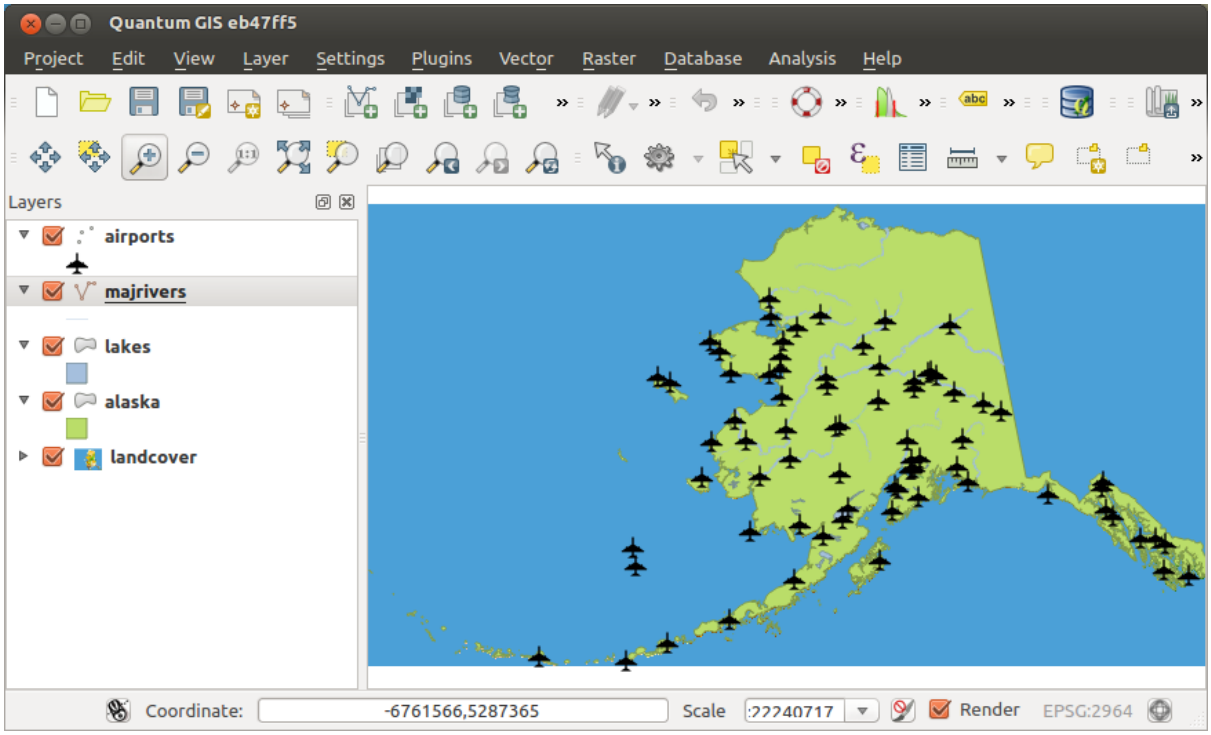


Abb. 24.24: Airports of Alaska

1. Open the *Heatmap (Kernel Density Estimation)* algorithm from the QGIS *Interpolation* group
2. In the *Point layer* field, select `airports` from the list of point layers loaded in the current project.
3. Change the *Radius* to `1000000` meters.
4. Change the *Pixel size X* to `1000`. The *Pixel size Y*, *Rows* and *Columns* will be automatically updated.
5. Click on *Run* to create and load the airports heatmap (see Abb. 24.26).

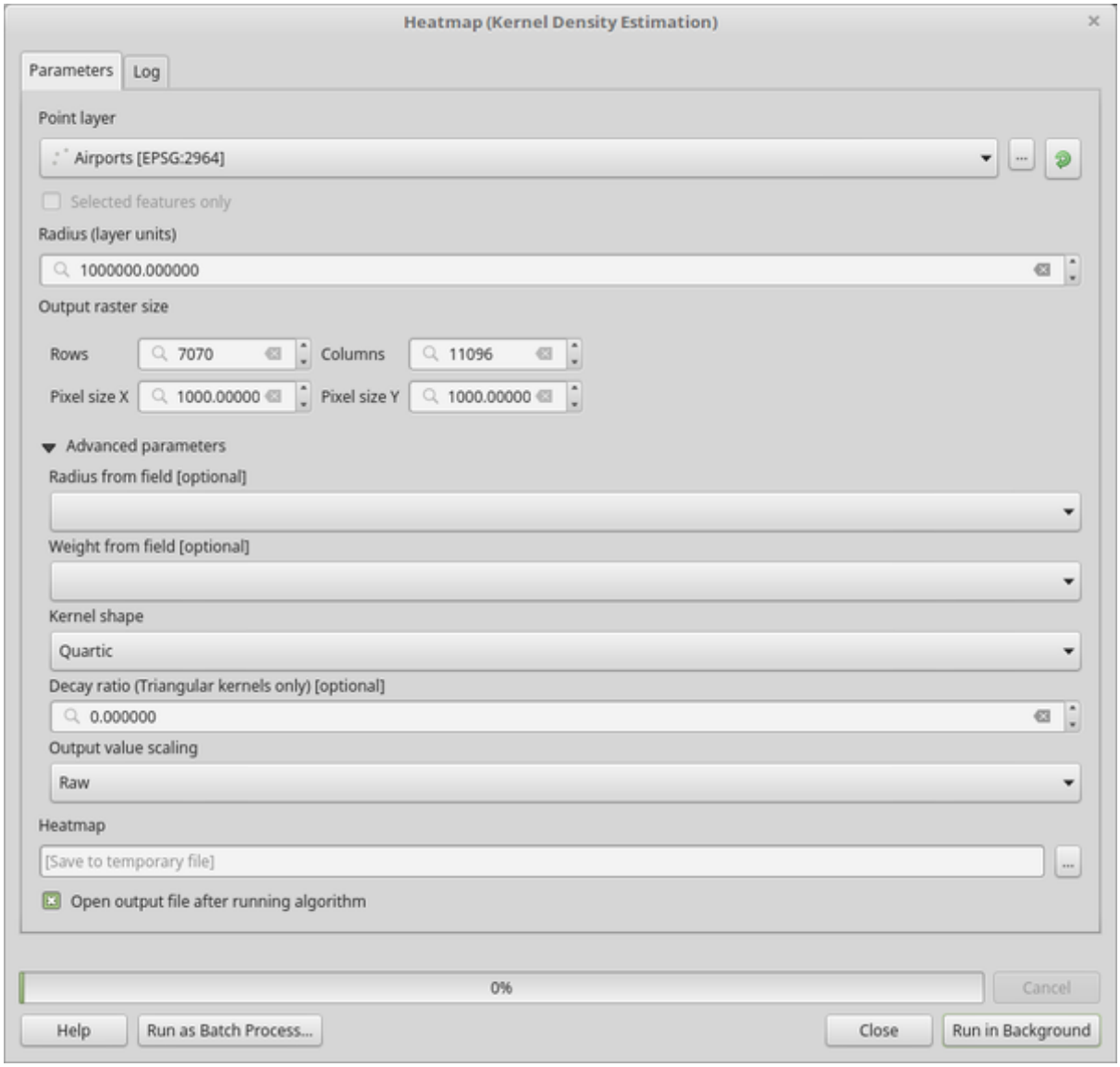


Abb. 24.25: The Heatmap Dialog

QGIS will generate the heatmap and add it to your map window. By default, the heatmap is shaded in greyscale, with lighter areas showing higher concentrations of airports. The heatmap can now be styled in QGIS to improve its appearance.

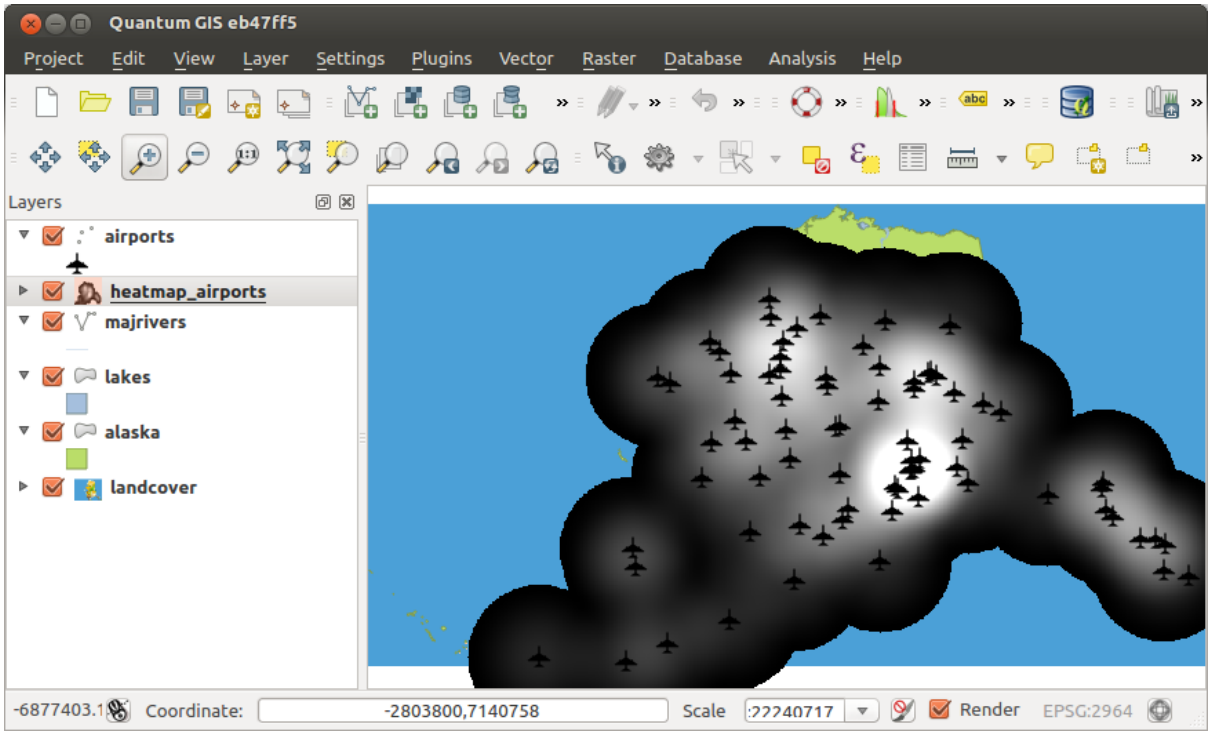


Abb. 24.26: The heatmap after loading looks like a grey surface

1. Open the properties dialog of the `heatmap_airports` layer (select the layer `heatmap_airports`, open the context menu with the right mouse button and select *Properties*).
2. Select the *Symbology* tab.
3. Change the *Render type* to ‚Singleband pseudocolor‘.
4. Select a suitable *Color ramp* , for instance `YlOrRd`.
5. Click the *Classify* button.
6. Press *OK* to update the layer.

The final result is shown in Abb. 24.27.

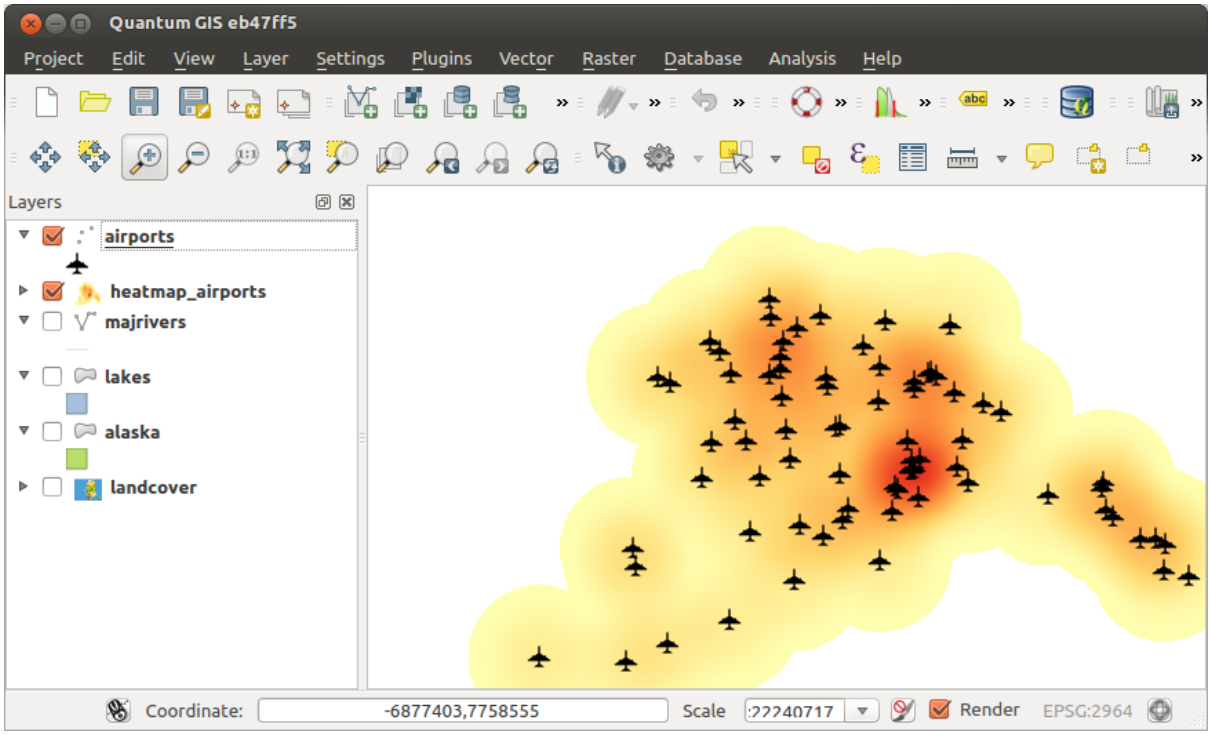


Abb. 24.27: Styled heatmap of airports of Alaska

Pythoncode

Algorithm ID: `qgis:heatmapkerneldensityestimation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

IDW Interpolation

Generates an Inverse Distance Weighted (IDW) interpolation of a point vector layer.

Sample points are weighted during interpolation such that the influence of one point relative to another declines with distance from the unknown point you want to create.

The IDW interpolation method also has some disadvantages: the quality of the interpolation result can decrease, if the distribution of sample data points is uneven.

Furthermore, maximum and minimum values in the interpolated surface can only occur at sample data points.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input layer(s)	`INTERPOLATION_`	[Text]	Vector layer(s) and field(s) to use for the interpolation, coded in a string (see the `ParameterInterpolationData` class in InterpolationWidgets for more details). The following GUI elements are provided to compose the interpolation data string: • **Vector layer** [vector: any] • **Interpolation attribute** [tablefield: numeric]: Attribute to use in the interpolation • **Use Z-coordinate for interpolation** [boolean]: Uses the layer's stored Z values (Default: False) For each of the added layer-field combinations, a type can be chosen: • *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. In the string, the layer-field elements are separated by `'::|::'`. The sub-elements of the layer-field elements are separated by `'::~::'`.
Distance coefficient P	`DISTANCE_COEFF`	[numerisch: Double] Default: 2.0	Sets the distance coefficient for the interpolation. Minimum: 0.0, maximum: 100.0.

Fortsetzung auf der nächsten Seite

Tab. 24.153 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Extent (xmin, xmax, ymin, ymax)	`EXTENT`	[Ausdehnung]	Extent of the output raster layer. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Output raster size	`PIXEL_SIZE`	[numerisch: Double] Default: 0.1	Pixel size of the output raster layer in layer units. In the GUI, the size can be specified by the number of rows (`Number of rows`) / columns (`Number of columns`) **or** the pixel size(`Pixel Size X` / `Pixel Size Y`). Increasing the number of rows or columns will decrease the cell size and increase the file size of the output raster. The values in `Rows`, `Columns`, `Pixel Size X` and `Pixel Size Y` will be updated simultaneously - doubling the number of rows will double the number of columns, and the cell size will be halved. The extent of the output raster will remain the same (approximately).
Interpolated	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Raster layer of interpolated values. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Interpolated	`OUTPUT`	[Raster]	Raster layer of interpolated values

Pythoncode

Algorithm ID: `qgis:idwinterpolation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Line Density

Calculates for each raster cell, the density measure of linear features within a circular neighbourhood. This measure is obtained by summing all the line segments intersecting the circular neighbourhood and dividing this sum by the area of such neighbourhood. A weighting factor can be applied to the line segments.

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

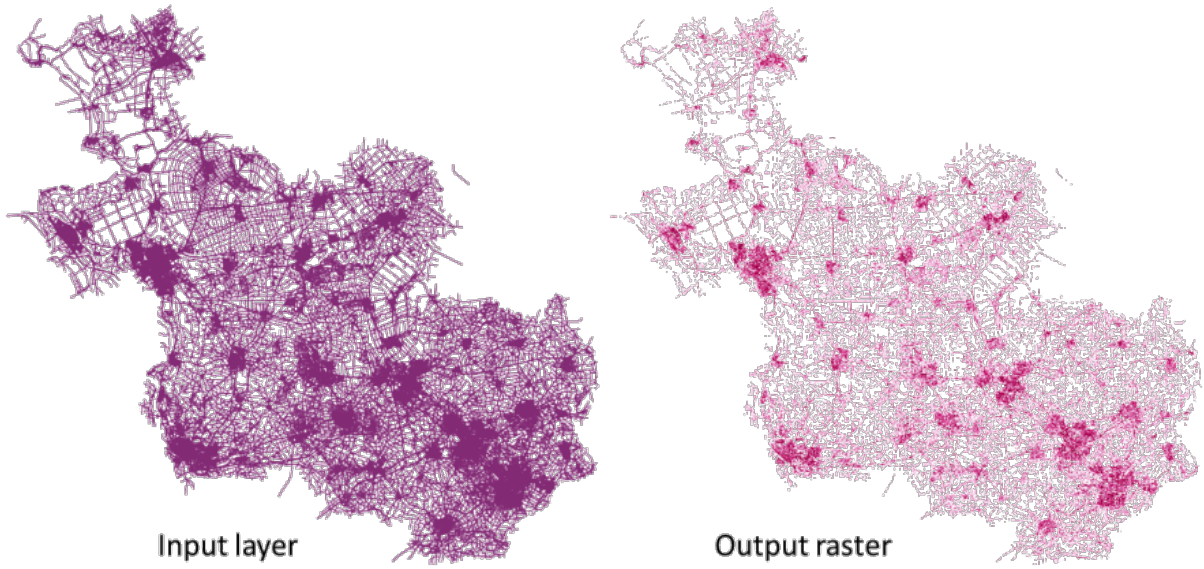


Abb. 24.28: Line density example. Input layer source: Roads Overijssel - The Netherlands (OSM).

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input line layer	`INPUT`	[vector: line]	Input vector layer containing line features
Weight field	`WEIGHT`	[tablefield: numeric]	Field of the layer containing the weight factor to use during the calculation

Fortsetzung auf der nächsten Seite

Tab. 24.155 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Search Radius	`RADIUS`	[numerisch: Double] Default: 10.0	Radius of the circular neighbourhood. Units can be specified here.
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Default: 10.0	Pixel size of the output raster layer in layer units. The raster has square pixels.
Line density raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	The output as a raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Line density raster	`OUTPUT`	[Raster]	The output line density raster layer.

Pythoncode

Algorithm ID: `native:linedensity`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

TIN Interpolation

Generates a Triangulated Irregular Network (TIN) interpolation of a point vector layer.

With the TIN method you can create a surface formed by triangles of nearest neighbor points. To do this, circumcircles around selected sample points are created and their intersections are connected to a network of non overlapping and as compact as possible triangles. The resulting surfaces are not smooth.

The algorithm creates both the raster layer of the interpolated values and the vector line layer with the triangulation boundaries.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input layer(s)	`INTERPOLATION_`	[Text]	Vector layer(s) and field(s) to use for the interpolation, coded in a string (see the `ParameterInterpolationData` class in InterpolationWidgets for more details). The following GUI elements are provided to compose the interpolation data string: • **Vector layer** [vector: any] • **Interpolation attribute** [tablefield: numeric]: Attribute to use in the interpolation • **Use Z-coordinate for interpolation** [boolean]: Uses the layer's stored Z values (Default: False) For each of the added layer-field combinations, a type can be chosen: • *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. In the string, the layer-field elements are separated by `'::|::'`. The sub-elements of the layer-field elements are separated by `'::~::'`.

Fortsetzung auf der nächsten Seite

Tab. 24.158 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Interpolation method	`METHOD`	[Aufzählung] Standard: 0	Set the interpolation method to be used. One of: • *Linear* • *Clough-Toucher (cubic)*
Extent (xmin, xmax, ymin, ymax)	`EXTENT`	[Ausdehnung]	Extent of the output raster layer. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Output raster size	`PIXEL_SIZE`	[numerisch: Double] Default: 0.1	Pixel size of the output raster layer in layer units. In the GUI, the size can be specified by the number of rows (`Number of rows`) / columns (`Number of columns`) **or** the pixel size(`Pixel Size X` / `Pixel Size Y`). Increasing the number of rows or columns will decrease the cell size and increase the file size of the output raster. The values in `Rows`, `Columns`, `Pixel Size X` and `Pixel Size Y` will be updated simultaneously - doubling the number of rows will double the number of columns, and the cell size will be halved. The extent of the output raster will remain the same (approximately).
Interpolated	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	The output TIN interpolation as a raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortsetzung auf der nächsten Seite

Tab. 24.158 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Triangulation	`TRIANGULATION`	[vector: line] Standard: `[Ausgabe überspringen]`	The output TIN as a vector layer. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Interpolated	`OUTPUT`	[Raster]	The output TIN interpolation as a raster layer
Triangulation	`TRIANGULATION`	[vector: line]	The output TIN as a vector layer.

Pythoncode

Algorithm ID: `qgis:tininterpolation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.9 Layer tools

Export layer(s) information

Creates a polygon layer with features corresponding to the extent of selected layer(s).

Additional layer details (CRS, provider name, file path, layer name, subset filter, abstract and attribution) are attached as attributes to each feature.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`LAYERS`	[layer] [list]	Input layers to get information on.
Ergebnis	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specification of the output layer with information. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ergebnis	`OUTPUT`	[Vektor: Polygon]	Polygon vector layer showing extent of input layers and associated information in attributes.

Pythoncode

Algorithm ID: `native:exportlayersinformation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export to spreadsheet

Exports the attributes of a selection of vector layers into a spreadsheet document or optionally appends them to an existing spreadsheet as additional sheets.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`LAYERS`	[vector: any] [list]	Input vector layers. The output spreadsheet will consist of a sheet, for each layer, that contains the attributes of this layer.
Use field aliases as column headings	`USE_ALIAS`	[Boolean] Standard: falsch	Use the field aliases from the attribute table for the spreadsheet.

Fortsetzung auf der nächsten Seite

Tab. 24.162 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Export formatted values instead of raw values	`FORMATTED_VALU`	[Boolean] Standard: falsch	If `True`, exports the formatted, human readable values (e.g., from a *value map or value relation*) to the spreadsheet.
Overwrite existing spreadsheet	`OVERWRITE`	[Boolean] Standard: True	If the specified spreadsheet exists, setting this option to `True` will overwrite the existing spreadsheet. If this option is `False` and the spreadsheet exists, the layers will be appended as additional sheets.
Destination spreadsheet	`OUTPUT`	[Datei] Standard: `[In temporäre Datei speichern]`	Output spreadsheet with a sheet for every layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Destination spreadsheet	`OUTPUT`	[Datei]	Spreadsheet with a sheet for every layer.
Layers within spreadsheet	`OUTPUT_LAYERS`	[list]	The list of sheets added to the spreadsheet.

Pythoncode

Algorithm ID: `native:exporttospreadsheet`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract layer extent

Generates a vector layer with the minimum bounding box (rectangle with N-S orientation) that covers all the input features.

The output layer contains a single bounding box for the whole input layer.

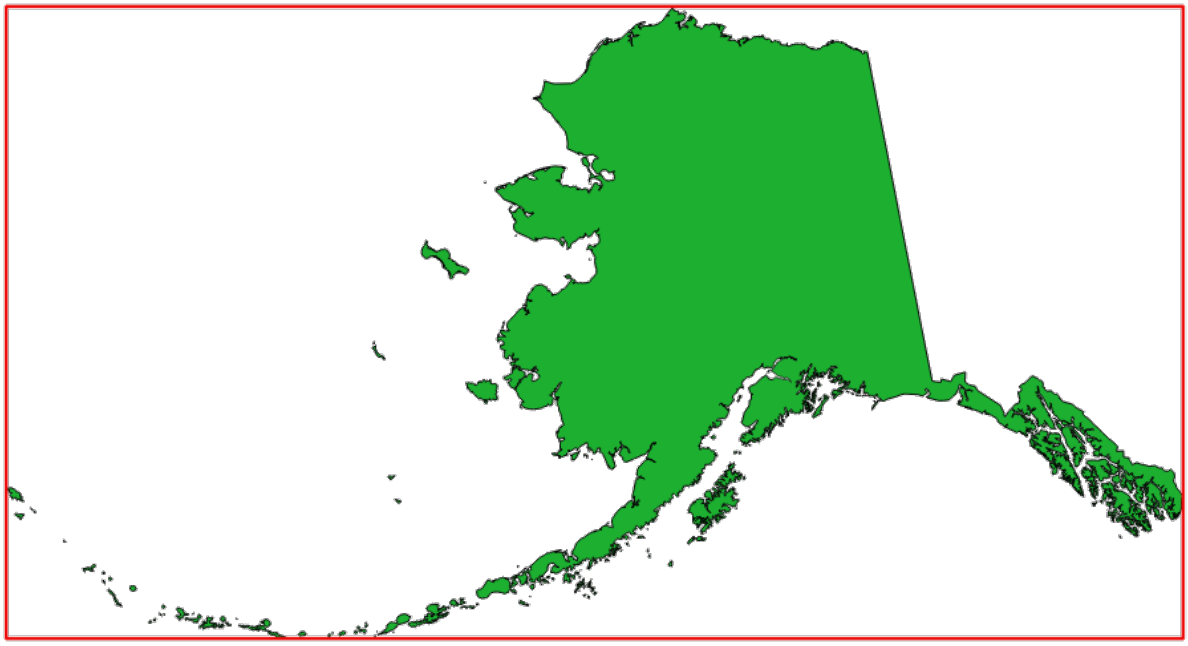


Abb. 24.29: In red the bounding box of the source layer

Default menu: *Vector* ► *Research Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Layer	`INPUT`	[layer]	Input layer
Extent	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the polygon vector layer for the output extent. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extent	`OUTPUT`	[Vektor: Polygon]	Output (polygon) vector layer with the extent (minimum bounding box)

Pythoncode

Algorithm ID: `native:polygonfromlayerextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.10 Netz

Export contours

Creates contours as a vector layer from a mesh scalar dataset.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input mesh layer	`INPUT`	[mesh]	The mesh layer to export data from
Dataset groups	`DATASET_GROUPS`	[layer] [list]	The dataset groups
Dataset time	`DATASET_TIME`	[datetime]	The time range to take into account • 0 — Current canvas time • 1 — Defined date/time • 2 — Dataset group time step
Increment between contour levels Optional	`INCREMENT`	[numerisch: Double] Standard: Nicht gesetzt	Interval between generated levels.
Minimum contour level Optional	`MINIMUM`	[numerisch: Double] Standard: Nicht gesetzt	Starting level values of contours. No generated levels will be lower than this value.
Maximum contour level Optional	`MAXIMUM`	[numerisch: Double] Standard: Nicht gesetzt	Maximum values of contours, i.e. no generated levels will be greater than this value.

Fortsetzung auf der nächsten Seite

Tab. 24.166 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
List of contours level Optional	`CONTOUR_LEVEL_`	[Text] Standard: Nicht gesetzt	List of desired levels of contours (separated by commas). If filled, the increment, minimum, and maximum fields will not be considered.
Output coordinate system Optional	`CRS_OUTPUT`	[KBS]	Coordinate Reference System to assign to the output
Exported contour lines	`OUTPUT_LINES`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line layer representing the contours of the mesh layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Exported contour polygons	`OUTPUT_POLYGON`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output polygon layer representing the contours of the mesh layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Exported contour lines	`OUTPUT_LINES`	[vector: line]	Line layer representing the contours of the mesh layer.
Exported contour polygons	`OUTPUT_POLYGON`	[Vektor: Polygon]	Polygon layer representing the contours of the mesh layer.

Pythoncode

Algorithm ID: `native:meshcontours`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export cross section dataset values on lines from mesh

Extracts a mesh dataset's values from lines contained in a vector layer.

Each line is discretized with a resolution distance parameter for extraction of values on its vertices.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input mesh layer	`INPUT`	[mesh]	The mesh layer to export data from
Dataset groups	`DATASET_GROUPS`	[layer] [list]	The dataset groups
Dataset time	`DATASET_TIME`	[datetime]	The time range to take into account • 0 — Current canvas time • 1 — Defined date/time • 2 — Dataset group time step
Lines for data export	`INPUT_LINES`	[vector: line]	Lines where the data will be extracted from the dataset mesh
Line segmentation resolution	`RESOLUTION`	[numerisch: Double] Default: 10.0	The distance between points on the lines where the data will be extracted from the dataset mesh. Units can be selected.
Digits count for coordinates	`COORDINATES_DI`	[numerisch: Ganzzahl] Default: 2	Number of digits for rounding the coordinate values
Digits count for dataset value	`DATASET_DIGITS`	[numerisch: Ganzzahl] Default: 2	Number of digits for rounding the dataset values
Exported data CSV file	`OUTPUT`	[Datei] Standard: `[In temporäre Datei speichern]`	Specification of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Exported data CSV file	`OUTPUT`	[Datei]	

Pythoncode

Algorithm ID: `native:meshexportcrosssection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export mesh edges

Exports a mesh layer's edges to a line vector layer, with the dataset values on edges as attribute values.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input mesh layer	`INPUT`	[mesh]	The mesh layer to export data from
Dataset groups	`DATASET_GROUPS`	[layer] [list]	The dataset groups
Dataset time	`DATASET_TIME`	[datetime]	The time range to take into account • 0 — Current canvas time • 1 — Defined date/time • 2 — Dataset group time step
Output coordinate system Optional	`CRS_OUTPUT`	[KBS]	Coordinate Reference System to assign to the output
Export vector option	`VECTOR_OPTION`	[Aufzählung]	Coordinate type of vector value exportation. • 0 — Cartesian (x,y) • 1 — Polar (magnitude, degree) • 2 — Cartesian and polar
Output vector layer	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specification of the output file. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output vector layer	`OUTPUT`	[vector: line]	Output vector line layer containing the edges of the input mesh layer with associated dataset values

Pythoncode

Algorithm ID: `native:exportmeshedges`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export mesh faces

Exports a mesh layer's faces to a polygon vector layer, with the dataset values on faces as attribute values.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input mesh layer	`INPUT`	[mesh]	The mesh layer to export data from
Dataset groups	`DATASET_GROUPS`	[layer] [list]	The dataset groups
Dataset time	`DATASET_TIME`	[datetime]	The time range to take into account • 0 — Current canvas time • 1 — Defined date/time • 2 — Dataset group time step
Output coordinate system Optional	`CRS_OUTPUT`	[KBS]	Coordinate Reference System to assign to the output
Export vector option	`VECTOR_OPTION`	[Aufzählung]	Coordinate type of vector value exportation. • 0 — Cartesian (x,y) • 1 — Polar (magnitude, degree) • 2 — Cartesian and polar
Output vector layer	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specification of the output file. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output vector layer	`OUTPUT`	[Vektor: Polygon]	Output vector polygon layer containing the faces of the input mesh layer with associated dataset values

Pythoncode

Algorithm ID: `native:exportmeshfaces`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export mesh on grid

Exports a mesh layer's dataset values to a gridded point vector layer, with the dataset values on this point as attribute values.

For data on volume (3D stacked dataset values), the exported dataset values are averaged on faces using the method defined in *the mesh layer properties* (default is Multi level averaging method). 1D meshes are not supported.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input mesh layer	`INPUT`	[mesh]	The mesh layer to export data from
Dataset groups	`DATASET_GROUPS`	[layer] [list]	The dataset groups
Dataset time	`DATASET_TIME`	[datetime]	The time range to take into account • 0 — Current canvas time • 1 — Defined date/time • 2 — Dataset group time step
Extent Optional	`EXTENT`	[Ausdehnung]	Specify the spatial extent on which to process the data. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Grid spacing Optional	`GRID_SPACING`	[numerisch: Double] Default: 10.0	Spacing between the sample points to use
Output coordinate system Optional	`CRS_OUTPUT`	[KBS]	Coordinate Reference System to assign to the output
Export vector option	`VECTOR_OPTION`	[Aufzählung]	Coordinate type of vector value exportation. • 0 — Cartesian (x,y) • 1 — Polar (magnitude, degree) • 2 — Cartesian and polar

Fortsetzung auf der nächsten Seite

Tab. 24.174 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output vector layer	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specification of the output file. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output vector layer	`OUTPUT`	[Vektor: Punkt]	Output vector point layer with dataset values computed from the overlaid face.

Pythoncode

Algorithm ID: `native:exportmeshongrid`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export mesh vertices

Exports a mesh layer's vertices to a point vector layer, with the dataset values on vertices as attribute values.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input mesh layer	`INPUT`	[mesh]	The mesh layer to export data from
Dataset groups	`DATASET_GROUPS`	[layer] [list]	The dataset groups
Dataset time	`DATASET_TIME`	[datetime]	The time range to take into account • 0 — Current canvas time • 1 — Defined date/time • 2 — Dataset group time step
Output coordinate system Optional	`CRS_OUTPUT`	[KBS]	Coordinate Reference System to assign to the output

Fortsetzung auf der nächsten Seite

Tab. 24.176 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Export vector option	`VECTOR_OPTION`	[Aufzählung]	Coordinate type of vector value exportation. • 0 — Cartesian (x,y) • 1 — Polar (magnitude, degree) • 2 — Cartesian and polar
Output vector layer	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specification of the output file. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output vector layer	`OUTPUT`	[Vektor: Punkt]	Output vector point layer containing the vertices of the input mesh layer with associated dataset values

Pythoncode

Algorithm ID: `native:exportmeshvertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export time series values from points of a mesh dataset

Extracts a mesh dataset's time series values from points contained in a vector layer.

If the time step is kept to its default value (0 hours), the time step used is the one of the two first datasets of the first selected dataset group.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input mesh layer	`INPUT`	[mesh]	The mesh layer to extract data from
Dataset groups	`DATASET_GROUPS`	[layer] [list]	The dataset groups
Starting time	`STARTING_TIME`	[datetime]	The start of the time range to take into account • 0 — Current canvas time • 1 — Defined date/time • 2 — Dataset group time step
Finishing time	`FINISHING_TIME`	[datetime]	The end of the time range to take into account • 0 — Current canvas time • 1 — Defined date/time • 2 — Dataset group time step
Time step (hours) Optional	`TIME_STEP`	[numerisch: Double] Standard: 0	Time between two consecutive steps to extract. Keep `0` to use time step of the first selected dataset group.
Points for data export	`INPUT_POINTS`	[Vektor: Punkt]	Vector layer containing points where the data will be extracted from the dataset mesh
Digits count for coordinates	`COORDINATES_DI`	[numerisch: Ganzzahl] Default: 2	Number of digits for rounding coordinate values
Digits count for dataset value	`DATASET_DIGITS`	[numerisch: Ganzzahl] Default: 2	Number of digits for rounding dataset values
Exported data CSV file	`OUTPUT`	[Datei] Standard: `[In temporäre Datei speichern]`	Specification of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Exported data CSV file	`OUTPUT`	[Datei]	`.CSV` file containing the mesh dataset time series values at the overlaying point features

Pythoncode

Algorithm ID: `native:meshexporttimeseries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Rasterize mesh dataset

Creates a raster layer from a mesh dataset.

For data on volume (3D stacked dataset values), the exported dataset values are averaged on faces using the method defined in *the mesh layer properties* (default is Multi level averaging method). 1D meshes are not supported.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input mesh layer	`INPUT`	[mesh]	The mesh layer to export data from
Dataset groups	`DATASET_GROUPS`	[layer] [list]	The dataset groups
Dataset time	`DATASET_TIME`	[datetime]	The time range to take into account • 0 — Current canvas time • 1 — Defined date/time • 2 — Dataset group time step
Extent Optional	`EXTENT`	[Ausdehnung]	Specify the spatial extent on which to process the data. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Fortsetzung auf der nächsten Seite

Tab. 24.180 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Standard: 1,0	Pixel size of the output raster layer.
Output coordinate system Optional	`CRS_OUTPUT`	[KBS]	Coordinate Reference System to assign to the output
Output raster layer	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster layer	`OUTPUT`	[Raster]	Output raster layer with dataset values computed from the mesh layer.

Pythoncode

Algorithm ID: `native:meshrasterize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

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.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input mesh layer	`INPUT`	[mesh]	The input mesh layer whose boundary will be exported as a polygon
Output coordinate system Optional	`OUTPUT_CRS`	[KBS]	The coordinate reference system (CRS) for the output polygon layer. If not specified, the CRS of the input mesh layer will be used.
Output vector layer	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specification of the output file. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output vector layer	`OUTPUT`	[Vektor: Polygon]	The resulting polygon layer containing the boundary of the mesh layer. The polygon may contain holes and may be multi-part.

Pythoncode

Algorithm ID: `native:surfacetopolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

TIN mesh creation

Creates a TIN mesh layer from vector layers. The TIN mesh is created using a Delaunay triangulation.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`SOURCE_DATA`	[vector: any] [list]	Vector layers to combine to generate the mesh layer
Vector layer	GUI ONLY	[vector: any] [list]	A selector for the vector layers to combine to generate the mesh layer
Value on vertex	GUI ONLY	[tablefield: any]	A selector of the field to use from the selected layer. Each vertex is assigned the corresponding value of its original feature.
Use Z-coordinate for value on vertex	GUI ONLY	[Boolean] Standard: falsch	If checked, the Z value of vector layer points or polygons/lines vertices will be used to assign the Z value of the vertex mesh layer. Only available if the input layers are 3D.
Output format	`MESH_FORMAT`	[Aufzählung] Default: 2DM	Output format of the generated layer • 0 — 2DM • 1 — SELAFIN • 2 — PLY • 3 — Ugrid • 4 — Mike21
Output coordinate system Optional	`CRS_OUTPUT`	[KBS]	Coordinate Reference System to assign to the output
Output file	`OUTPUT_MESH`	[mesh] Standard: `[In temporäre Datei speichern]`	Specification of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output file	`OUTPUT_MESH`	[mesh]	Output mesh layer with dataset values computed from the vector layers.

Pythoncode

Algorithm ID: `native:tinmeshcreation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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
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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* 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 *Verarbeitungs-Algorithmen von der Konsole aus verwenden* for details on how to run processing algorithms from the Python console.

24.1.12 Modellierungswerkzeuge

Warnung

Diese Werkzeuge sind nur in der Grafischen Modellierung verfügbar. Sie sind nicht in der Verarbeitungs-Toolbox verfügbar.

Calculate expression

It calculates the result of a QGIS expression and eliminates the need to use the same expression multiple times throughout a model when the same result is required more than once. Additionally, it enables use cases that would otherwise not be possible. For instance, you can generate a timestamp value once and use it multiple times within the model, if the timestamp were recalculated every time, the values would vary during the model's runtime.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input	`INPUT`	[Ausdruck]	Expression to calculate

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Value	`OUTPUT`	[Result Value]	Calculated result value, the data type of the output will vary based on the specific expression used in the algorithm.

Pythoncode

Algorithm ID: `native:calculateexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Bedingte Verzweigung

Fügt einem Modell eine bedingte Verzweigung hinzu, so dass Teile des Modells auf der Grundlage des Ergebnisses eines Ausdrucks ausgeführt werden können. Meistens durch die Verwendung von Werkzeugabhängigkeiten, um den Ablauf des Modells zu steuern.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Field	`BRANCH`	[Text]	Name of the condition
Field	`CONDITION`	[Ausdruck]	Expression to evaluate

Ausgaben

None.

Pythoncode

Algorithm ID: `native:condition`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create directory

Creates a new directory on a file system. Directories will be created recursively, creating all required parent directories in order to construct the full specified directory path. No errors will be raised if the directory already exists.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Directory path	`PATH`	[Text]	Folder path to create

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ergebnis	`OUTPUT`	[Verzeichnis]	Created folder

Pythoncode

Algorithm ID: `native:createdirectory`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Feature filter

Filters features from the input layer and redirects them to one or several outputs. If you do not know about any attribute names that are common to all possible input layers, filtering is only possible on the feature geometry and general record mechanisms, such as `$id` and `uuid`.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	The input layer.
Outputs and filters (one or more)	`OUTPUT_<name of the filter>`	[wie Eingabe]	The output layers with filters (as many as there are filters).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ergebnis (one or more)	`native:filter_ of filter>`	[wie Eingabe]	The output layers with filtered features (as many as there are filters).

Pythoncode

Algorithm ID: `native:filter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Filter layers by type

Filters layers by their type. Incoming layers will be directed to different outputs based on whether they are a vector or raster layer.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[layer]	Generic Map Layer

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Vector features Optional	`VECTOR`	[Vektor]	A Vector Layer of the input, if compatible
Raster layer Optional	`RASTER`	[Raster]	A Raster Layer of the input, if compatible

Pythoncode

Algorithm ID: `native:filterlayersbytype`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Load layer into project

Loads a layer to the current project.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Layer	INPUT	[layer]	Layer to load in the legend
Loaded layer name	NAME	[Text]	Name of the loaded layer

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Layer	OUTPUT	[wie Eingabe]	The (renamed) loaded layer

Pythoncode

Algorithm ID: `native:loadlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raise exception

Raises an exception and cancels a model's execution. The exception message can be customized, and optionally an expression based condition can be specified. If an expression condition is used, then the exception will only be raised if the expression result is true. A false result indicates that no exception will be raised, and the model execution can continue uninterrupted.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Message	MESSAGE	[Text]	Message to display
Condition Optional	CONDITION	[Ausdruck]	Expression to evaluate if true

Ausgaben

A message in the log panel.

Pythoncode

Algorithm ID: `native:raiseexception`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raise message

Raises an information message in the log. The message can be customized, and optionally an expression based condition can be specified. If an expression condition is used, then the message will only be logged if the expression result is true. A false result indicates that no message will be logged.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Information message	`MESSAGE`	[Text]	Message to display
Condition Optional	`CONDITION`	[Ausdruck]	Expression to evaluate if true

Ausgaben

A message in the log panel.

Pythoncode

Algorithm ID: `native:raisemessage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raise warning

Raises a warning message in the log. The warning message can be customized, and optionally an expression based condition can be specified. If an expression condition is used, then the warning will only be logged if the expression result is true. A false result indicates that no warning will be logged.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Message	`MESSAGE`	[Text]	Message to display
Condition Optional	`CONDITION`	[Ausdruck]	Expression to evaluate if true

Ausgaben

A message in the log panel.

Pythoncode

Algorithm ID: `native:raisewarning`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Rename layer

Renames a layer.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Layer	`INPUT`	[layer]	Layer to rename
New name	`NAME`	[Text]	The new name of the layer

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Layer	`OUTPUT`	[wie Eingabe]	The (renamed) output layer

Pythoncode

Algorithm ID: `native:renamelayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Save log to file

Saves the model's execution log to a file. Optionally, the log can be saved in a HTML formatted version.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Use HTML	`USE_HTML`	[Boolean] Standard: falsch	Use HTML formatting

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
File	`OUTPUT`	[Text]	Destination of the log

Pythoncode

Algorithm ID: `native:savelog`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Set project variable

Sets an expression variable for the current project.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Variable name	NAME	[Text]	Name of the variable
Variable value	VALUE	[Text]	Value to be stored

Ausgaben

None.

Pythoncode

Algorithm ID: native:setprojectvariable

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

String concatenation

Concatenates two strings into a single one in the Processing Modeler.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input 1	INPUT_1	[Text]	First string
Input 2	INPUT_2	[Text]	Second string

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Concatenation	CONCATENATION	[Text]	The concatenated string

Pythoncode

Algorithm ID: `native:stringconcatenation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Variable distance buffer

Warnung

This algorithm is deprecated and can be removed anytime. Prefer using *Puffer* algorithm instead.

Computes a buffer area for all the features in an input layer.

The size of the buffer for a given feature is defined by an attribute, so it allows different features to have different buffer sizes.

Siehe auch

Puffer

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer
Distance field	`DISTANCE`	[tablefield: numeric]	Attribute for the distance radius of the buffer
Segmente	`SEGMENTS`	[numerisch: Ganzzahl] Default: *5*	Steuert die Anzahl der Liniensegmente, die zur Annäherung an einen Viertelkreis beim Erstellen von abgerundeten Puffern verwendet werden sollen.
Ergebnis auflösen	`DISSOLVE`	[Boolean] Default: *False*	Choose to dissolve the final buffer, resulting in a single feature covering all input features.

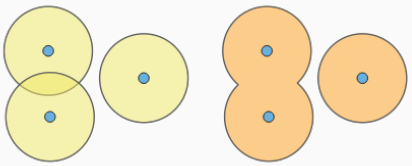


Abb. 24.30: Normal and dissolved buffer

Fortsetzung auf der nächsten Seite

Tab. 24.220 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Endstil	`END_CAP_STYLE`	[Aufzählung] Default: Round	Controls how line endings are handled in the buffer. Abb. 24.31: Runder, flacher und quadratischer Endstil
Verbindungsstil	`JOIN_STYLE`	[Aufzählung] Default: Round	Specifies whether round, miter or beveled joins should be used when offsetting corners in a line.
Eckgrenze	`MITER_LIMIT`	[numerisch: Double] Default: 2.0	Only applicable for mitered join styles, and controls the maximum distance from the offset curve to use when creating a mitered join.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Buffer	`OUTPUT`	[Vektor: Polygon]	Buffer polygon vector layer.

Pythoncode

Algorithm ID: `qgis:variabledistancebuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.13 Network analysis

Service area (from layer)

Returns all the edges or parts of edges of a network that can be reached within a distance or a time, starting from a point layer. This allows evaluation of accessibility within a network, e.g., what are the places I can navigate to on a road network without spending cost greater than a given value (the cost can be distance or time).

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Vector layer representing network	`INPUT`	[vector: line]	Line vector layer representing the network to be covered
Vector layer with start points	`START_POINTS`	[Vektor: Punkt]	Point vector layer whose features are used as start points to generate the service areas
Path type to calculate	`STRATEGY`	[Aufzählung] Standard: 0	The type of path to calculate. One of: • 0 — Shortest • 1 — Fastest
Travel cost (distance for „Shortest“, time for „Fastest“)	`TRAVEL_COST`	[numerisch: Double] Default: 0.0	The value is estimated as a distance (in the network layer units) when looking for the *Shortest* path and as time (in hours) for the *Fastest* path.
Service area (lines)	`OUTPUT_LINES`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line layer for the service area. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Service area (boundary nodes)	`OUTPUT`	[Vektor: Punkt] Standard: `[Ausgabe überspringen]`	Specify the output point layer for the service area boundary nodes. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Non-routable features	`OUTPUT_NON_ROU`	[Vektor: Punkt] Standard: `[Ausgabe überspringen]`	Specify the output which will be used to store any input features which could not be routed (e.g., those which are too far from the network layer). *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Direction field Optional	DIRECTION_FIEL	[tablefield: any]	The field used to specify directions for the network edges. The values used in this field are specified with the three parameters `Value for forward direction`, `Value for backward direction` and `Value for both directions`. Forward and reverse directions correspond to a one-way edge, „both directions“ indicates a two-way edge. If a feature does not have a value in this field, or no field is set then the default direction setting (provided with the `Default direction` parameter) is used.
Value for forward direction Optional	VALUE_FORWARD	[Text] Default: ‚ (empty string)	Value set in the direction field to identify edges with a forward direction
Value for backward direction Optional	VALUE_BACKWARD	[Text] Default: ‚ (empty string)	Value set in the direction field to identify edges with a backward direction
Value for both directions Optional	VALUE_BOTH	[Text] Default: ‚ (empty string)	Value set in the direction field to identify bidirectional edges
Default direction	DEFAULT_DIRECT	[Aufzählung] Default: 2	If a feature has no value set in the direction field or if no direction field is set, then this direction value is used. One of: • 0 — Forward direction • 1 — Backward direction • 2 — Both directions
Speed field Optional	SPEED_FIELD	[tablefield: numeric]	Field providing the speed value (in `km/h`) for the edges of the network when looking for the fastest path. If a feature does not have a value in this field, or no field is set then the default speed value (provided with the `Default speed` parameter) is used.
Default speed (km/h)	DEFAULT_SPEED	[numerisch: Double] Default: 50.0	Value to use to calculate the travel time if no speed value is provided for an edge in the specified field
Topology tolerance	TOLERANCE	[numerisch: Double] Default: 0.0	Two lines with nodes closer than the specified tolerance are considered connected

Include upper/lower bound points	INCLUDE_BOUNDS	[Boolean] Standard: falsch	Creates a point layer output with two points for each edge at the boundaries of the service area. One point is the start of that edge, the other is the end.
Maximum point distance from network Optional	POINT_TOLERANC	[numerisch: Double] Standard: Nicht gesetzt	Specifies an optional limit on the distance from the points to the network layer. If a point is further from the network than this distance it will be treated as non-routable. If not set, endpoints will be snapped to the nearest point on the network layer, regardless of how far away from the network they actually are.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Service area (boundary nodes)	OUTPUT	[Vektor: Punkt]	The output point layer with the service area boundary nodes.
Service area (lines)	OUTPUT_LINES	[vector: line]	Line layer representing the parts of the network that can be serviced by the start points, for the given cost.
Non routable features	OUTPUT_NON_ROU	[Vektor: Punkt]	An optional output which will be used to store any input features which could not be routed (e.g., those which are too far from the network layer).

Python-Code

Algorithm ID: `native:serviceareafromlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Service area (from point)

Returns all the edges or parts of edges of a network that can be reached within a given distance or time, starting from a point feature. This allows the evaluation of accessibility within a network, e.g., what are the places I can navigate to on a road network without spending a cost greater than a given value (the cost can be distance or time).

> **Bemerkung**
>
> This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Vector layer representing the network	`INPUT`	[vector: line]	Line vector layer representing the network to be covered
Path type to calculate	`STRATEGY`	[Aufzählung] Standard: 0	The type of path to calculate. One of: • 0 — Shortest • 1 — Fastest
Start point	`START_POINT`	[coordinates]	Coordinate of the point to calculate the service area around. Press the … button next to the option and click on the canvas to fill the parameter with the clicked point coordinate.
Travel cost (distance for „Shortest“, time for „Fastest“)	`TRAVEL_COST`	[numerisch: Double] Default: 0.0	The value is estimated as a distance (in the network layer units) when looking for the *Shortest* path and as time (in hours) for the *Fastest* path.
Service area (lines) Optional	`OUTPUT_LINES`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line layer for the service area. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Service area (boundary nodes) Optional	`OUTPUT`	[Vektor: Punkt] Standard: `[Ausgabe überspringen]`	Specify the output point layer for the service area boundary nodes. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Direction field Optional	DIRECTION_FIEL	[tablefield: any]	The field used to specify directions for the network edges. The values used in this field are specified with the three parameters `Value for forward direction`, `Value for backward direction` and `Value for both directions`. Forward and reverse directions correspond to a one-way edge, „both directions“ indicates a two-way edge. If a feature does not have a value in this field, or no field is set then the default direction setting (provided with the `Default direction` parameter) is used.
Value for forward direction Optional	VALUE_FORWARD	[Text] Default: ‚ (empty string)	Value set in the direction field to identify edges with a forward direction
Value for backward direction Optional	VALUE_BACKWARD	[Text] Default: ‚ (empty string)	Value set in the direction field to identify edges with a backward direction
Value for both directions Optional	VALUE_BOTH	[Text] Default: ‚ (empty string)	Value set in the direction field to identify bidirectional edges
Default direction	DEFAULT_DIRECT	[Aufzählung] Default: 2	If a feature has no value set in the direction field or if no direction field is set, then this direction value is used. One of: • 0 — Forward direction • 1 — Backward direction • 2 — Both directions
Speed field Optional	SPEED_FIELD	[tablefield: numeric]	Field providing the speed value (in `km/h`) for the edges of the network when looking for the fastest path. If a feature does not have a value in this field, or no field is set then the default speed value (provided with the `Default speed` parameter) is used.
Default speed (km/h)	DEFAULT_SPEED	[numerisch: Double] Default: 50.0	Value to use to calculate the travel time if no speed value is provided for an edge in the specified field
Topology tolerance	TOLERANCE	[numerisch: Double] Default: 0.0	Two lines with nodes closer than the specified tolerance are considered connected

Maximum point distance from network Optional	POINT_TOLERANC	[numerisch: Double] Default: 0.0	Specifies an optional limit on the distance from the start point to the network layer. If the point is further from the network than this distance an error will be raised. If not set, the point will be snapped to the nearest point on the network layer, regardless of how far away from the network it actually is.
Include upper/lower bound points	INCLUDE_BOUNDS	[Boolean] Standard: falsch	Creates a point layer output with two points for each edge at the boundaries of the service area. One point is the start of that edge, the other is the end.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Service area (boundary nodes)	OUTPUT	[Vektor: Punkt]	The output point layer with the service area boundary nodes.
Service area (lines)	OUTPUT_LINES	[vector: line]	Line layer representing the parts of the network that can be serviced by the start point, for the given cost.

Python-Code

Algorithm ID: `native:serviceareafrompoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Shortest path (layer to point)

Computes the optimal (shortest or fastest) routes from multiple start points defined by a vector layer and a given end point.

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Vector layer representing network	`INPUT`	[vector: line]	Line vector layer representing the network to be covered
Path type to calculate	`STRATEGY`	[Aufzählung] Standard: 0	The type of path to calculate. One of: • 0 — Shortest • 1 — Fastest
Vector layer with start points	`START_POINTS`	[Vektor: Punkt]	Point vector layer whose features are used as start points of the routes
End point (x, y)	`END_POINT`	[coordinates]	Point feature representing the end point of the routes Press the … button next to the option and click on the canvas to fill the parameter with the clicked point coordinate.
Shortest path	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line layer for the shortest paths. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Non-routable features Optional	`OUTPUT_NON_ROU`	[Vektor: Punkt] Standard: `[Ausgabe überspringen]`	Specify the output which will be used to store any input features which could not be routed (e.g., those which are too far from the network layer). *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Direction field Optional	DIRECTION_FIEL	[tablefield: any]	The field used to specify directions for the network edges. The values used in this field are specified with the three parameters `Value for forward direction`, `Value for backward direction` and `Value for both directions`. Forward and reverse directions correspond to a one-way edge, „both directions“ indicates a two-way edge. If a feature does not have a value in this field, or no field is set then the default direction setting (provided with the `Default direction` parameter) is used.
Value for forward direction Optional	VALUE_FORWARD	[Text] Default: ‚ (empty string)	Value set in the direction field to identify edges with a forward direction
Value for backward direction Optional	VALUE_BACKWARD	[Text] Default: ‚ (empty string)	Value set in the direction field to identify edges with a backward direction
Value for both directions Optional	VALUE_BOTH	[Text] Default: ‚ (empty string)	Value set in the direction field to identify bidirectional edges
Default direction	DEFAULT_DIRECT	[Aufzählung] Default: 2	If a feature has no value set in the direction field or if no direction field is set, then this direction value is used. One of: • 0 — Forward direction • 1 — Backward direction • 2 — Both directions
Speed field Optional	SPEED_FIELD	[tablefield: numeric]	Field providing the speed value (in `km/h`) for the edges of the network when looking for the fastest path. If a feature does not have a value in this field, or no field is set then the default speed value (provided with the `Default speed` parameter) is used.
Default speed (km/h)	DEFAULT_SPEED	[numerisch: Double] Default: 50.0	Value to use to calculate the travel time if no speed value is provided for an edge in the specified field
Topology tolerance	TOLERANCE	[numerisch: Double] Default: 0.0	Two lines with nodes closer than the specified tolerance are considered connected

Maximum point distance from network Optional	POINT_TOLERANC	[numerisch: Double] Standard: Nicht gesetzt	Specifies an optional limit on the distance from the start and end points to the network layer. If a start feature is further from the network than this distance it will be treated as non-routable. If the end point is further from the network than this distance an error will be raised. If not set, points will be snapped to the nearest point on the network layer, regardless of how far away from the network they actually are.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Shortest path	`OUTPUT`	[vector: line]	Line layer of the shortest or fastest path from each of the start points to the end point
Non routable features	`OUTPUT_NON_ROU`	[Vektor: Punkt]	An optional output layer which will be used to store any input features which could not be routed (e.g., those which are too far from the network layer).

Python-Code

Algorithm ID: `native:shortestpathlayertopoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Shortest path (point to layer)

Computes the optimal (shortest or fastest) routes between a given start point and multiple end points defined by a point vector layer.

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Vector layer representing network	`INPUT`	[vector: line]	Line vector layer representing the network to be covered
Path type to calculate	`STRATEGY`	[Aufzählung] Standard: 0	The type of path to calculate. One of: • 0 — Shortest • 1 — Fastest

Fortsetzung auf der nächsten Seite

Tab. 24.231 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Start point	`START_POINT`	[coordinates]	Point feature representing the start point of the routes Press the … button next to the option and click on the canvas to fill the parameter with the clicked point coordinate.
Vector layer with end points	`END_POINTS`	[Vektor: Punkt]	Point vector layer whose features are used as end points of the routes
Shortest path	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line layer for the shortest paths. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Non-routable features Optional	`OUTPUT_NON_ROU`	[Vektor: Punkt] Standard: `[Ausgabe überspringen]`	Specify the output which will be used to store any input features which could not be routed (e.g., those which are too far from the network layer). *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Direction field Optional	`DIRECTION_FIEL`	[tablefield: any]	The field used to specify directions for the network edges. The values used in this field are specified with the three parameters `Value for forward direction`, `Value for backward direction` and `Value for both directions`. Forward and reverse directions correspond to a one-way edge, „both directions“ indicates a two-way edge. If a feature does not have a value in this field, or no field is set then the default direction setting (provided with the `Default direction` parameter) is used.
Value for forward direction Optional	`VALUE_FORWARD`	[Text] Default: ‚‘ (empty string)	Value set in the direction field to identify edges with a forward direction

Fortsetzung auf der nächsten Seite

Tab. 24.232 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Value for backward direction Optional	VALUE_BACKWARD	[Text] Default: ‚' (empty string)	Value set in the direction field to identify edges with a backward direction
Value for both directions Optional	VALUE_BOTH	[Text] Default: ‚' (empty string)	Value set in the direction field to identify bidirectional edges
Default direction	DEFAULT_DIRECT	[Aufzählung] Default: 2	If a feature has no value set in the direction field or if no direction field is set, then this direction value is used. One of: • 0 — Forward direction • 1 — Backward direction • 2 — Both directions
Speed field Optional	SPEED_FIELD	[tablefield: numeric]	Field providing the speed value (in km/h) for the edges of the network when looking for the fastest path. If a feature does not have a value in this field, or no field is set then the default speed value (provided with the Default speed parameter) is used.
Default speed (km/h)	DEFAULT_SPEED	[numerisch: Double] Default: 50.0	Value to use to calculate the travel time if no speed value is provided for an edge in the specified field
Topology tolerance	TOLERANCE	[numerisch: Double] Default: 0.0	Two lines with nodes closer than the specified tolerance are considered connected
Maximum point distance from network Optional	POINT_TOLERANC	[numerisch: Double] Standard: Nicht gesetzt	Specifies an optional limit on the distance from the start and end points to the network layer. If the start point is further from the network than this distance an error will be raised. If an end feature is further from the network than this distance it will be treated as non-routable. If not set, points will be snapped to the nearest point on the network layer, regardless of how far away from the network they actually are.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Shortest path	OUTPUT	[vector: line]	Line layer of the shortest or fastest path from each of the start points to the end point
Non routable features	OUTPUT_NON_ROU	[Vektor: Punkt]	An optional output layer which will be used to store any input features which could not be routed (e.g., those which are too far from the network layer).

Python-Code

Algorithm ID: `native:shortestpathpointtolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Shortest path (point to point)

Computes the optimal (shortest or fastest) route between a given start point and a given end point.

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Vector layer representing network	`INPUT`	[vector: line]	Line vector layer representing the network to be covered
Path type to calculate	`STRATEGY`	[Aufzählung] Standard: 0	The type of path to calculate. One of: • 0 — Shortest • 1 — Fastest
Start point (x, y)	`START_POINT`	[coordinates]	Point feature representing the start point of the routes. Press the … button next to the option and click on the canvas to fill the parameter with the clicked point coordinate.
End point (x, y)	`END_POINT`	[coordinates]	Point feature representing the end point of the routes. Press the … button next to the option and click on the canvas to fill the parameter with the clicked point coordinate.
Shortest path	`OUTPUT`	[vector: line]	Specify the output line layer for the shortest paths. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Direction field Optional	`DIRECTION_FIEL`	[tablefield: any]	The field used to specify directions for the network edges. The values used in this field are specified with the three parameters `Value for forward direction`, `Value for backward direction` and `Value for both directions`. Forward and reverse directions correspond to a one-way edge, „both directions“ indicates a two-way edge. If a feature does not have a value in this field, or no field is set then the default direction setting (provided with the `Default direction` parameter) is used.
Value for forward direction Optional	`VALUE_FORWARD`	[Text] Default: ‚ (empty string)	Value set in the direction field to identify edges with a forward direction
Value for backward direction Optional	`VALUE_BACKWARD`	[Text] Default: ‚ (empty string)	Value set in the direction field to identify edges with a backward direction
Value for both directions Optional	`VALUE_BOTH`	[Text] Default: ‚ (empty string)	Value set in the direction field to identify bidirectional edges
Default direction	`DEFAULT_DIRECT`	[Aufzählung] Default: 2	If a feature has no value set in the direction field or if no direction field is set, then this direction value is used. One of: • 0 — Forward direction • 1 — Backward direction • 2 — Both directions
Speed field Optional	`SPEED_FIELD`	[tablefield: numeric]	Field providing the speed value (in `km/h`) for the edges of the network when looking for the fastest path. If a feature does not have a value in this field, or no field is set then the default speed value (provided with the `Default speed` parameter) is used.
Default speed (km/h)	`DEFAULT_SPEED`	[numerisch: Double] Default: 50.0	Value to use to calculate the travel time if no speed value is provided for an edge in the specified field
Topology tolerance	`TOLERANCE`	[numerisch: Double] Default: 0.0	Two lines with nodes closer than the specified tolerance are considered connected

Maximum point distance from network Optional	POINT_TOLERANC	[numerisch: Double] Standard: Nicht gesetzt	Specifies an optional limit on the distance from the start and end points to the network layer. If either point is further from the network than this distance an error will be raised. If not set, points will be snapped to the nearest point on the network layer, regardless of how far away from the network they actually are.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Shortest path	OUTPUT	[vector: line]	Line layer of the shortest or fastest path from each of the start point to the end point

Python-Code

Algorithm ID: `native:shortestpathpointtopoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.14 Plots

Bar plot

Creates a bar plot from a category and a layer field.

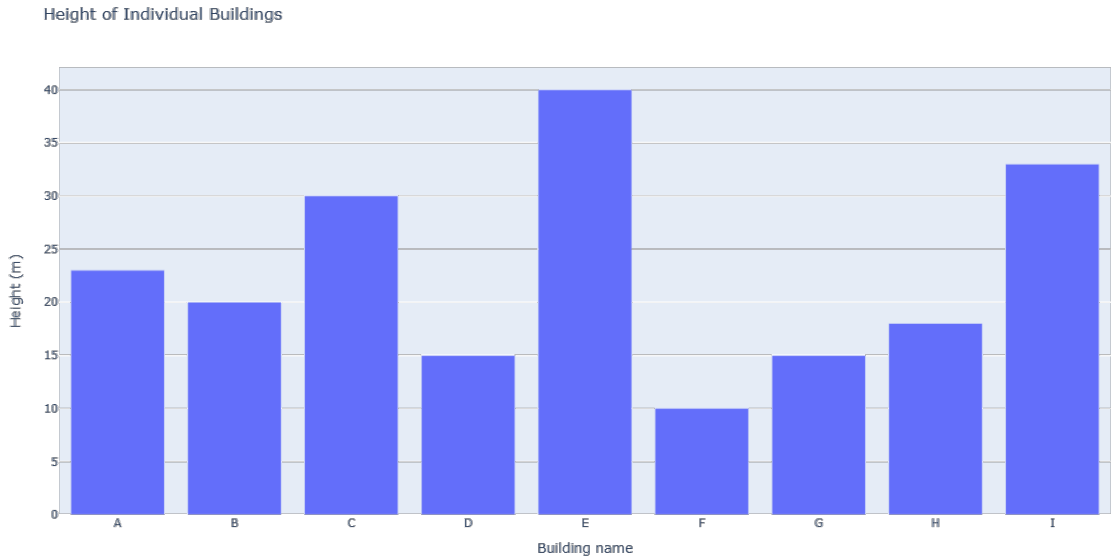


Abb. 24.32: Bar plot example

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Category field name	`NAME_FIELD`	[tablefield: any]	Categorical field to use for grouping the bars (X axis)
Value field	`VALUE_FIELD`	[tablefield: numeric]	Value to use for the plot (Y axis).
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.
Bar plot	`OUTPUT`	[html] Standard: `[In temporäre Datei speichern]`	Specify the HTML file for the plot. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Bar plot	`OUTPUT`	[html]	HTML file with the plot. Available in the *Processing* ► *Result Viewer*.

Python-Code

Algorithm ID: `qgis:barplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Box plot

Creates a box plot from a category field and a numerical layer field.

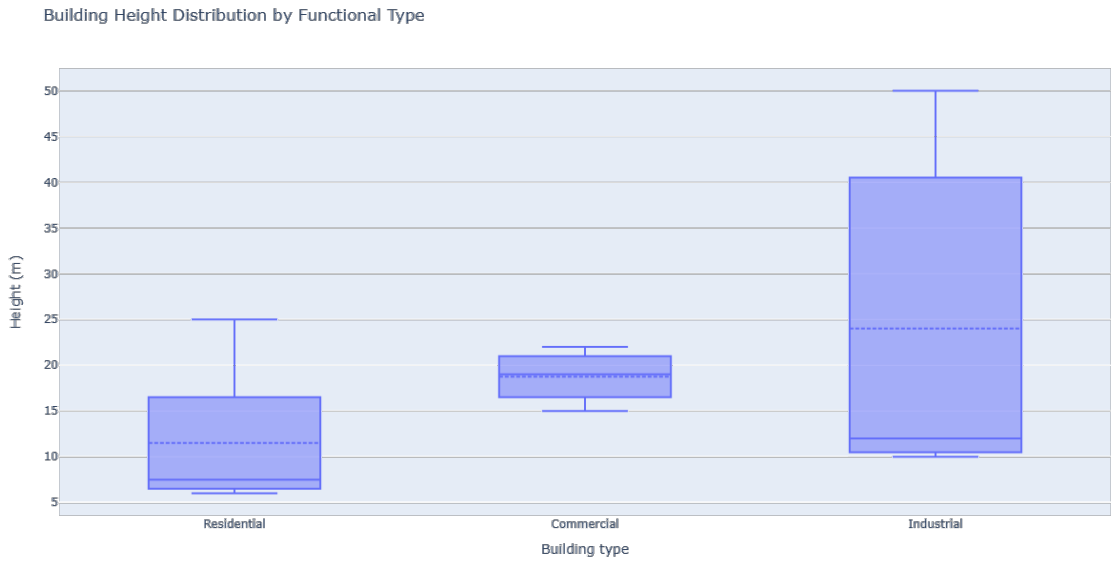


Abb. 24.33: Example of box plot

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Category name field	`NAME_FIELD`	[tablefield: any]	Categorical field to use for grouping the boxes (X axis)
Value field	`VALUE_FIELD`	[tablefield: numeric]	Value to use for the plot (Y axis).
Additional statistic lines	`MSD`	[Aufzählung] Standard: 0	Additional statistics information to add to the plot. One of: • 0 — Show Mean • 1 — Show Standard Deviation • 2 — Don't show mean and standard deviation
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 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.
Box plot	`OUTPUT`	[html] Standard: `[In temporäre Datei speichern]`	Specify the HTML file for the plot. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Box plot	`OUTPUT`	[html]	HTML file with the plot. Available in the *Processing* ► *Result Viewer*.

Python-Code

Algorithm ID: `qgis:boxplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Generate elevation profile image

Creates an elevation profile image from a list of map layers and an optional terrain layer.

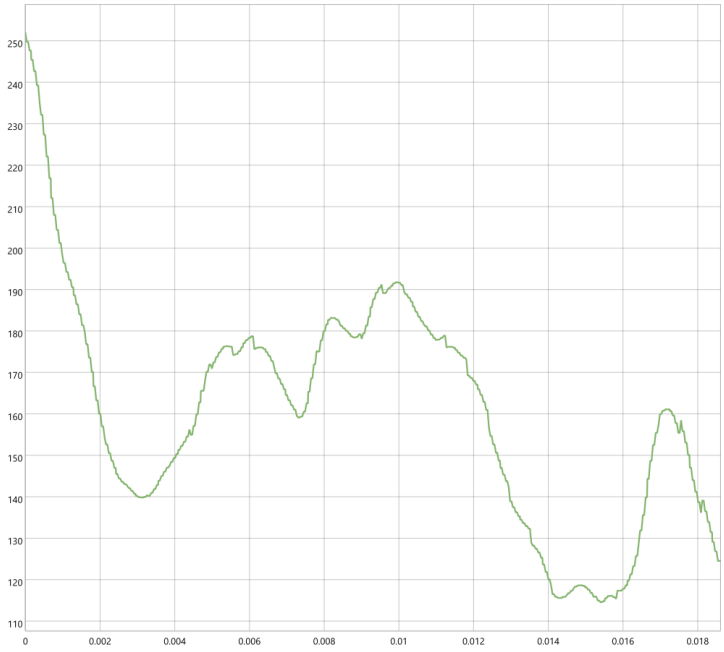


Abb. 24.34: Example of elevation profile

Parameter

Basic parameters

Bezeichnung	Name	Typ	Beschreibung
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`	[numerisch: Ganzzahl] Default: 400 Minimum value: 0	The width of the output chart in pixels.
Chart height (in pixels)	`HEIGHT`	[numerisch: Ganzzahl] 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

Bezeichnung	Name	Typ	Beschreibung
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	[numerisch: Ganzzahl] Default: 96 Minimum value: 0	The resolution of the output image in dots per inch (DPI).
Output image	OUTPUT	[file] Standard: [In temporäre Datei speichern]	Specify the image file for the plot. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output image	OUTPUT	[file]	The generated elevation profile image in the specified format.

Python-Code

Algorithm ID: `native:generateelevationprofileimage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Mean and standard deviation plot

Creates a box plot with mean and standard deviation values.

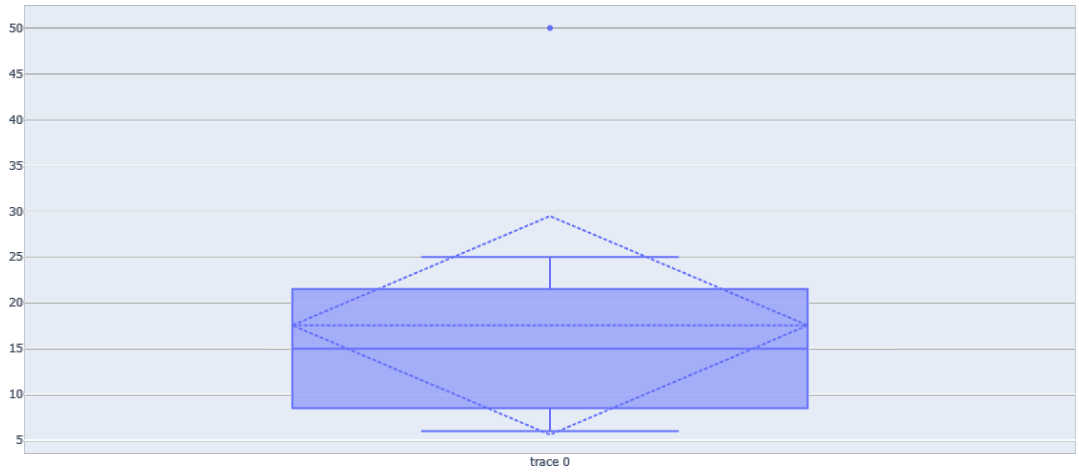


Abb. 24.35: Example of mean and standard deviation plot

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input table	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Category name field	`NAME_FIELD`	[tablefield: any]	Categorical field to use for grouping the boxes (X axis)
Value field	`VALUE_FIELD`	[tablefield: numeric]	Value to use for the plot (Y axis).
Plot	`OUTPUT`	[html] Standard: `[In temporäre Datei speichern]`	Specify the HTML file for the plot. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Plot	`OUTPUT`	[html]	HTML file with the plot. Available in the *Processing* ► *Result Viewer*.

Python-Code

Algorithm ID: `qgis:meanandstandarddeviationplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Polar plot

Generates a polar plot based on the value of an input vector layer.

Two fields must be entered as parameters: one that defines the category each feature (to group features) and another one with the variable to plot (this has to be a numeric one).

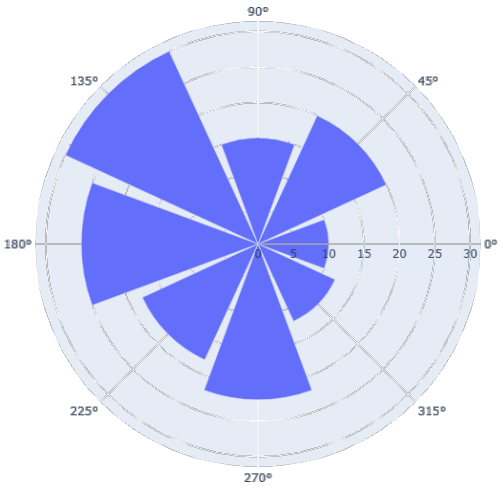


Abb. 24.36: Example of polar plot

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Category name field	`NAME_FIELD`	[tablefield: any]	Categorical field to use for grouping the features (X axis)
Value field	`VALUE_FIELD`	[tablefield: numeric]	Value to use for the plot (Y axis).
Polar plot	`OUTPUT`	[html] Standard: `[In temporäre Datei speichern]`	Specify the HTML file for the plot. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Polar plot	`OUTPUT`	[html]	HTML file with the plot. Available in the *Processing* ► *Result Viewer*.

Python-Code

Algorithm ID: `qgis:polarplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster layer histogram

Generates a histogram with the values of a raster layer.

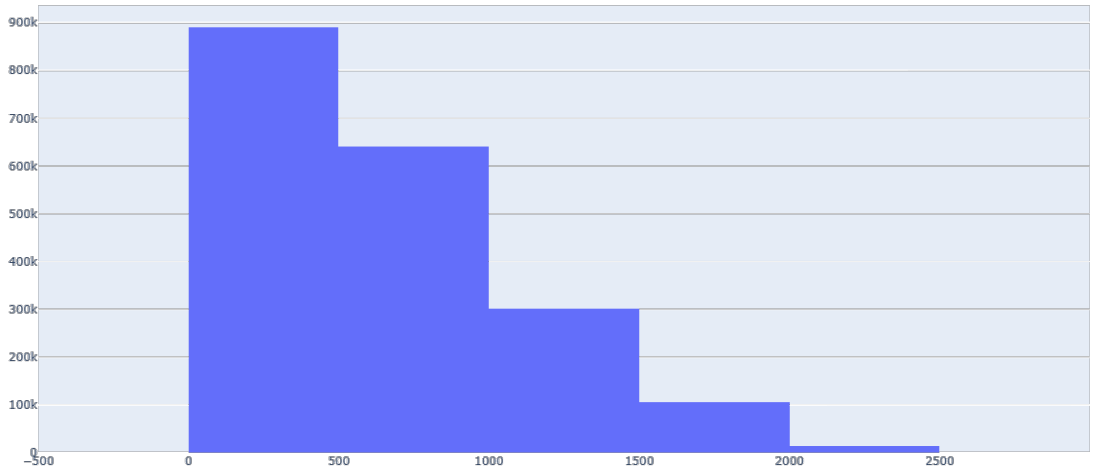


Abb. 24.37: Raster layer histogram example

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer
Band number	`BAND`	[raster band]	Raster band to use for the histogram
number of bins	`BINS`	[numerisch: Ganzzahl] Default: 10	The number of bins to use in the histogram (X axis). Minimum 2.
Histogram	`OUTPUT`	[html] Standard: `[In temporäre Datei speichern]`	Specify the HTML file for the plot. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Histogram	`OUTPUT`	[html]	HTML file with the plot. Available in the *Processing* ► *Result Viewer*.

Python-Code

Algorithm ID: `qgis:rasterlayerhistogram`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Vector layer histogram

Generates a histogram with the values of the attribute of a vector layer.

The attribute to use for computing the histogram must be numeric.

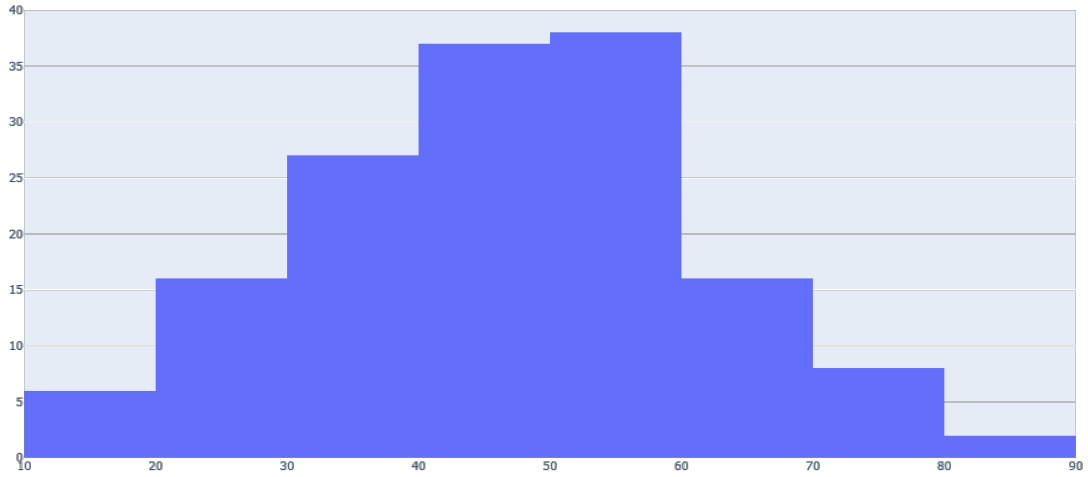


Abb. 24.38: Example of vector layer histogram

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer
Attribut	`FIELD`	[tablefield: numeric]	Value to use for the plot (Y axis).
number of bins	`BINS`	[numerisch: Ganzzahl] Default: 10	The number of bins to use in the histogram (X axis). Minimum 2.
Histogram	`OUTPUT`	[html] Standard: `[In temporäre Datei speichern]`	Specify the HTML file for the plot. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Histogram	`OUTPUT`	[html]	HTML file with the plot. Available in the *Processing* ► *Result Viewer*.

Python-Code

Algorithm ID: `qgis:vectorlayerhistogram`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Vector layer scatterplot

Creates a simple `X` - `Y` scatter plot for a vector layer.

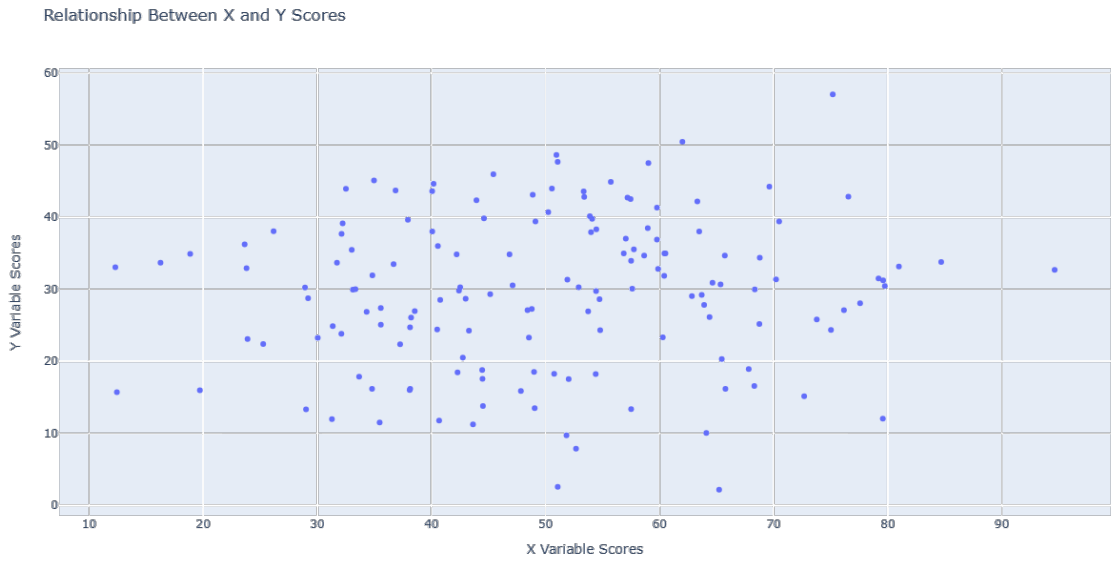


Abb. 24.39: Example of vector layer scatterplot

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
X attribute	`XFIELD`	[tablefield: numeric]	Field to use for the X axis
Y attribute	`YFIELD`	[tablefield: numeric]	Field to use for the Y axis
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
Scatterplot	`OUTPUT`	[html] Standard: `[In temporäre Datei speichern]`	Specify the HTML file for the plot. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Scatterplot	`OUTPUT`	[html]	HTML file with the plot. Available in the *Processing* ► *Result Viewer*.

Python-Code

Algorithm ID: `qgis:vectorlayerscatterplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Vector layer scatterplot 3D

Creates a 3D scatter plot for a vector layer.

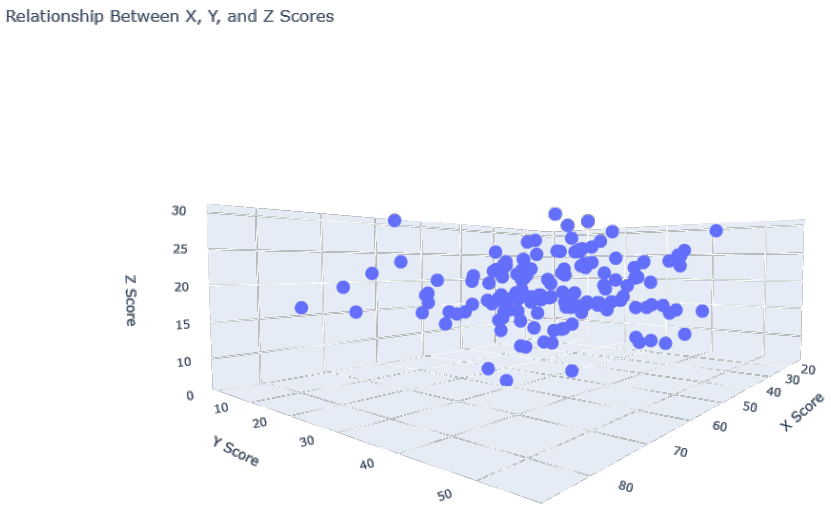


Abb. 24.40: Example of 3D scatter plot

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
X attribute	`XFIELD`	[tablefield: numeric]	Field to use for the X axis
Y attribute	`YFIELD`	[tablefield: numeric]	Field to use for the Y axis
Z attribute	`ZFIELD`	[tablefield: numeric]	Field to use for the Z axis
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] Standard: `[In temporäre Datei speichern]`	Specify the HTML file for the plot. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Scatterplot 3D	`OUTPUT`	[html]	HTML file with the plot. Available in the *Processing* ► *Result Viewer*.

Python-Code

Algorithm ID: `qgis:scatter3dplot`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.15 Punktwolkenkonvertierung

Achtung

Running these algorithms requires QGIS installed with PDAL >= 2.5.0 (see *Help* ► *About* menu).

Format konvertieren

Konvertiert eine Punktwolke in ein anderes Dateiformat, z.B. erstellt eine komprimierte `.LAZ`.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Punktwolke-Eingabelayer, der konvertiert werden soll
Umgewandelt	`OUTPUT`	[Punktwolke] Standard: `[In temporäre Datei speichern]`	Specify the point cloud file to use as output. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Umgewandelt	`OUTPUT`	[Punktwolke]	Ausgabe des Punktwolken-Layers in einem modifizierten Dateiformat. Derzeit unterstützte Formate sind `.LAS`, `.LAZ`, `.COPC.LAZ` und `.VPC`.

Python-Code

Algorithmus-Kennung: `pdal:convertformat`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster exportieren

Exportiert Punktwolkendaten in ein 2D-Rastergitter mit einer Zellengröße der gegebenen Auflösung und schreibt Werte aus dem angegebenen Attribut.

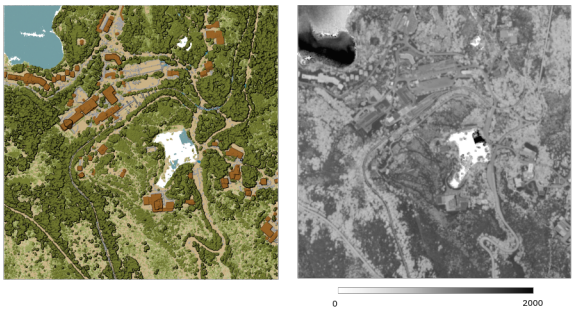


Abb. 24.41: Rasterausgabe unter Verwendung des Attributs „Intensität“ der Punkte

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Layer der zu exportierenden Punktwolke
Attribut	`ATTRIBUTE`	[Feld] [Aufzählung]	Ein Feld des Punktwolken-Layers, aus dem die Werte extrahiert werden sollen
Auflösung des Rasters	`RESOLUTION`	[numerisch: Double] Standard: 1,0	Zellengröße des Ausgaberasters

Fortsetzung auf der nächsten Seite

Tab. 24.258 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Kachelgröße für parallele Durchläufe	`TILE_SIZE`	[numerisch: Ganzzahl] Standard: 1000	
Exportiert	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the raster file to export the data to. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Filterausdruck Optional	`FILTER_EXPRESS`	[Ausdruck]	Ein *PDAL-Ausdruck* zur Auswahl einer Untergruppe von Objekten in den Punktwolkendaten
Begrenzungsausmaß Optional	`FILTER_EXTENT`	[Ausdehnung]	Ein Kartenausschnitt zur Auswahl einer Teilmenge von Objekten in den Punktwolkendaten Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Aktuelles Kartenansichtsausmaß verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
X-Ursprung einer Kachel für parallele Läufe Optional	`ORIGIN_X`	[numerisch: Double]	
Y-Ursprung einer Kachel für parallele Läufe Optional	`ORIGIN_Y`	[numerisch: Double]	

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Exportiert	`OUTPUT`	[Raster]	Rasterlayer, in den die Daten der Punktwolke exportiert werden. Das derzeit unterstützte Format ist `.TIF`.

Python-Code

Algorithmus-Kennung: `pdal:exportraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Als Raster exportieren (mit Triangulation)

Exportiert Punktwolkendaten in ein 2D-Rastergitter unter Verwendung einer Triangulation von Punkten und anschließender Interpolation von Zellenwerten aus Dreiecken.

Bemerkung

Dieser Algorithmus kann langsamer sein, wenn Sie mit einem großen Datensatz arbeiten. Wenn Ihre Punktwolke sehr dicht ist, können Sie Ihre Bodenpunkte mit dem Algorithmus *Raster exportieren* als Raster exportieren.

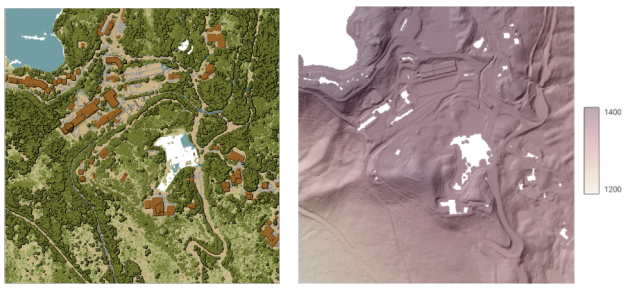


Abb. 24.42: Durch Triangulation von Punktwolken erzeugte Geländerasterausgabe

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Layer der zu exportierenden Punktwolke

Fortsetzung auf der nächsten Seite

Tab. 24.261 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Auflösung des Rasters	`RESOLUTION`	[numerisch: Double] Standard: 1,0	Zellengröße des Ausgaberasters
Kachelgröße für parallele Durchläufe	`TILE_SIZE`	[numerisch: Ganzzahl] Standard: 1000	
Exportiert	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the raster file to export the data to. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Filterausdruck Optional	`FILTER_EXPRESS`	[Ausdruck]	Ein *PDAL-Ausdruck* zur Auswahl einer Untergruppe von Objekten in den Punktwolkendaten
Begrenzungsausmaß Optional	`FILTER_EXTENT`	[Ausdehnung]	Ein Kartenausschnitt zur Auswahl einer Teilmenge von Objekten in den Punktwolkendaten Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Aktuelles Kartenansichtsausmaß verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
X-Ursprung einer Kachel für parallele Läufe Optional	`ORIGIN_X`	[numerisch: Double]	
Y-Ursprung einer Kachel für parallele Läufe Optional	`ORIGIN_Y`	[numerisch: Double]	

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Exportiert (mit Triangulation)	`OUTPUT`	[Raster]	Rasterlayer, in den die Daten der Punktwolke exportiert werden. Das derzeit unterstützte Format ist `.TIF`.

Python-Code

Algorithmus-Kennung: `pdal:exportrastertin`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Als Vektor exportieren

Exportiert Punktwolkendaten in einen Vektorlayer mit 3D Punkten (ein GeoPackage), optional mit zusätzlichen Attributen.

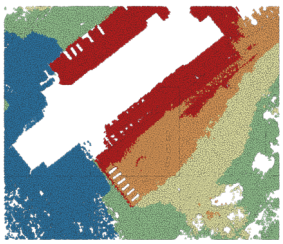


Abb. 24.43: Exportieren der Punktwolke (Bodenpunkte) in einen Vektorlayer, der auf der Grundlage der Erhöhung erstellt wurde

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Layer der zu exportierenden Punktwolke
Attribut Optional	`ATTRIBUTE`	[Feld] [Liste]	Ein oder mehrere Felder des Layers der Punktwolke, der mit den Punkten exportiert werden soll.
Exportiert	`OUTPUT`	[Vektor] Standard: `[In temporäre Datei speichern]`	Specify the vector file to export the data to. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Filterausdruck Optional	`FILTER_EXPRESS`	[Ausdruck]	Ein *PDAL-Ausdruck* zur Auswahl einer Untergruppe von Objekten in den Punktwolkendaten
Begrenzungsausmaß Optional	`FILTER_EXTENT`	[Ausdehnung]	Ein Kartenausschnitt zur Auswahl einer Teilmenge von Objekten in den Punktwolkendaten Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Aktuelles Kartenansichtsausmaß verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Exportiert	`OUTPUT`	[Vektor]	Vektorlayer, in den die Daten der Punktwolke exportiert werden. Das derzeit unterstützte Format ist `.GPKG`.

Python-Code

Algorithmus-Kennung: `pdal:exportvector`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.16 Punktwolkendatenverwaltung

Achtung

Running these algorithms requires QGIS installed with PDAL >= 2.5.0 (see *Help* ► *About* menu).

Projektion zuweisen

Weist einem Layer der Punktwolke ein Koordinatenreferenzsystem zu, wenn es fehlt oder falsch ist. Ein neuer Layer wird erstellt.

Siehe auch

Reprojizieren

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Punktwolke-Eingabelayer, dem ein KBS zugewiesen werden soll
Gewünschte KBS	`CRS`	[KBS]	Das KBS, das auf den Layer angewendet werden soll
Ausgabelayer	`OUTPUT`	[Punktwolke] Standard: `[In temporäre Datei speichern]`	Specify the point cloud file to use as output. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Konvertiert	`OUTPUT`	[Punktwolke]	Punktwolken-Ausgabelayer mit dem neuen KBS. Derzeit unterstützte Formate sind `.LAS`, `.LAZ`, `.COPC.LAZ` und `.VPC`.

Pythoncode

Algorithmus-Kennung: `pdal:assignprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Virtuelle Punktwolke aufbauen (VPC)

Erzeugt eine *virtuelle Punktwolke (VPC)* aus den eingegebenen Punktwolkendaten.

Wenn Sie die optionalen Parameter nicht aktivieren, wird die VPC-Datei sehr schnell erstellt, da der Algorithmus nur die Metadaten der Eingabedateien liest. Wenn einer der optionalen Parameter gesetzt ist, liest der Algorithmus alle Punkte, was einige Zeit dauern kann.

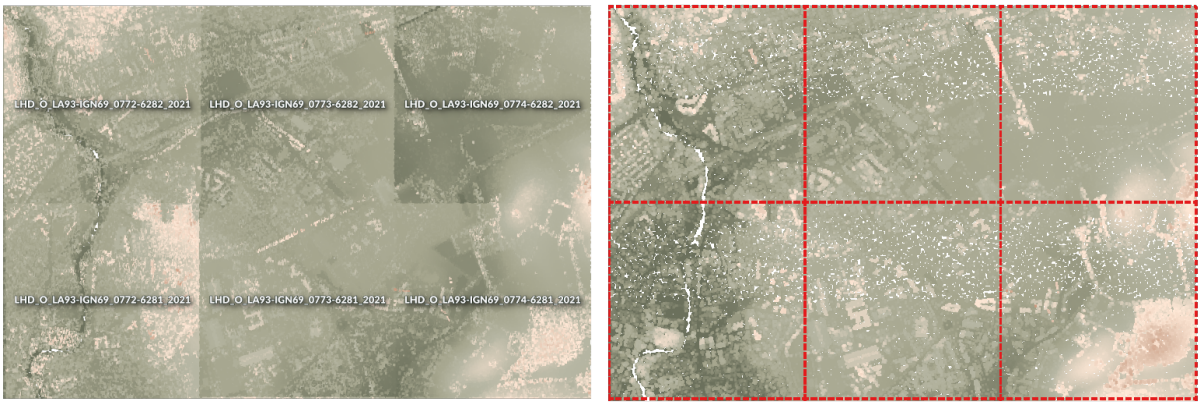


Abb. 24.44: Erzeugung einer virtuellen Punktwolke mit Übersicht aus einem Satz von Punktwolkenkacheln

Siehe auch

Vereinigen

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`LAYERS`	[Punktwolke] [Liste]	Eingabe von einem oder mehreren Punktwolken-Layern zur Kombination in einem virtuellen Punktwolken-Layer
Umgrenzungspolygc berechnen	`BOUNDARY`	[Boolean] Standard: falsch	Auf „wahr“ setzen, um die genauen Grenzen der Daten anzuzeigen (und nicht nur die rechteckige Ausdehnung)
Statistik berechnen	`STATISTICS`	[Boolean] Standard: falsch	Auf „wahr“ setzen, um die Wertebereiche der verschiedenen Attribute zu ermitteln
Punktwolkenübersic aufbauen	`OVERVIEW`	[Boolean] Standard: falsch	Erzeugt eine einzelne „ausgedünnte“ Punktwolke aus allen Eingabedaten (wobei nur jeder 1000ste Punkt der Originaldaten verwendet wird). Die Punktwolkenübersicht wird neben der VPC-Datei erstellt - für `mydata.vpc` würde die Punktwolkenübersicht zum Beispiel `mydata-overview.copc.laz` heißen.
Virtuelle Punktwolke	`OUTPUT`	[Punktwolke] Standard: `[In temporäre Datei speichern]`	Specify the point cloud file to build the data into. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Virtuelle Punktwolke	`OUTPUT`	[Raster]	Punktwolken-Ausgabelayer, der alle Eingabedaten in einer virtuellen Datei kombiniert

Pythoncode

Algorithmus-Kennung: `pdal:virtualpointcloud`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Zuschneiden

Schneidet einen Punktwolken-Layer durch einen Polygon Layer so aus, dass die resultierende Punktwolke nur Punkte innerhalb der Polygone enthält.

Abb. 24.45: Beschneiden eines Punktwolken-Eingabelayers mit einer Polygonabdeckung

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Punktwolke-Eingabelayer, der ausgeschnitten werden soll
Begrenzungspolygor	`OVERLAY`	[Vektor: Polygon]	Polygon-Vektor Layer, der als Abdeckung für das Ausschneiden der Punkte verwendet wird
Zugeschnitten	`OUTPUT`	[Punktwolke] Standard: `[In temporäre Datei speichern]`	Specify the point cloud file to export the clipped points to. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Filterausdruck Optional	`FILTER_EXPRESS`	[Ausdruck]	Ein *PDAL-Ausdruck* zur Auswahl einer Untergruppe von Objekten in den Punktwolkendaten
Begrenzungsausmal Optional	`FILTER_EXTENT`	[Ausdehnung]	Ein Kartenausschnitt zur Auswahl einer Teilmenge von Objekten in den Punktwolkendaten Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Zugeschnitten	`OUTPUT`	[Punktwolke]	Punktwolke-Ausgabelayer, deren Objekte die Punkte innerhalb des Polygon-Abdeckungslayers sind.

Pythoncode

Algorithmus-Kennung: `pdal:clip`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

COPC erzeugen

Erzeugt einen Index (COPC-Datei) für alle eingegebenen Punktwolkendateien im Stapelverarbeitungs-Modus.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`LAYERS`	[Punktwolke] [Liste]	Ein oder mehrere Punktwolke-Eingabelayer, für die ein Index erstellt werden soll
Ausgabeverzeichnis Optional	`OUTPUT`	[Verzeichnis] Standard: `[Ausgabe überspringen]`	Specify the folder to create the new files in. *One of*: • Ausgabe übersrpingen • In einem temporären Verzeichnis speichern • In Verzeichnis speichern

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabeverzeichnis	`OUTPUT`	[Verzeichnis]	Ausgabeordner mit Punktwolken-Layern und zugehörigen COPC-Indexdateien.

Pythoncode

Algorithmus-Kennung: `pdal:createcopc`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Information

Gibt grundlegende Metadaten aus einer Punktwolken-Eingabedatei aus.

Beispiel für die Ausgabe von Informationen:

```
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
```

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Punktwolke-Eingabelayer, aus dem die Metadateninformationen ausgelesen werden sollen
Layer Information	`OUTPUT`	[Datei] Standard: `[In temporäre Datei speichern]`	Specify the file to store the metadata information. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Layer Information	`OUTPUT`	[html]	`HTML`-Datei zum Speichern der Metadaten-Informationen.

Pythoncode

Algorithmus-Kennung: `pdal:info`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Vereinigen

Führt mehrere Punktwolkendateien zu einer einzigen zusammen.

Siehe auch

Virtuelle Punktwolke aufbauen (VPC)

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`LAYERS`	[Punktwolke] [Liste]	Punktwolke-Eingabelayer, die zu einer einzigen zusammengeführt werden sollen
Zusammengeführt	`OUTPUT`	[Punktwolke] Standard: `[In temporäre Datei speichern]`	Specify the output point cloud merging input files. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Filterausdruck Optional	`FILTER_EXPRESS`	[Ausdruck]	Ein *PDAL-Ausdruck* zur Auswahl einer Untergruppe von Objekten in den Punktwolkendaten
Begrenzungsausmaß Optional	`FILTER_EXTENT`	[Ausdehnung]	Ein Kartenausschnitt zur Auswahl einer Teilmenge von Objekten in den Punktwolkendaten Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Zusammengeführt	`OUTPUT`	[Punktwolke]	Punktwolke-Ausgabelayer, der alle Eingabedateien zusammenfasst.

Pythoncode

Algorithmus-Kennung: `pdal:merge`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Reprojizieren

Reprojiziert eine Punktwolke auf ein anderes Koordinatenreferenzsystem (KBS).

Siehe auch

Projektion zuweisen

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Punktwolke-Eingabelayer, der in ein anderes KBS neu projizieren werden soll
Ziel KBS	`CRS`	[KBS]	Das KBS, das auf den Layer angewendet werden soll
Reprojiziert	`OUTPUT`	[Punktwolke] Standard: `[In temporäre Datei speichern]`	Specify the reprojected point cloud file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Koordinatenoperati Optional	`OPERATION`	[Datum]	Die *Datumstransformation*, die zur Reprojektion der Daten zwischen dem Ursprungs- und dem Zielsystem verwendet werden soll.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Reprojiziert	`OUTPUT`	[Punktwolke]	Punktwolke-Ausgabelayer im Ziel-KBS.

Pythoncode

Algorithmus-Kennung: `pdal:reproject`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Dünn (durch Abtastradius)

Erzeugt eine ausgedünnte Version der Punktwolke, indem eine Stichprobe nach Abstandspunkten durchgeführt wird (reduziert die Anzahl der Punkte innerhalb eines bestimmten Radius).

Siehe auch

Dünn (durch Auslassen von Punkten)

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Punktwolke-Eingabelayer, von dem eine ausgedünnte Version erstellt werden soll
Abtastradius (in Karteneinheiten)	`SAMPLING_RADIU`	[numerisch: Double] Standard: 1,0	Abstand, innerhalb dessen Punkte zu einem einzelnen Punkt abgetastet werden
Ausgedünnt (durch Radius)	`OUTPUT`	[Punktwolke] Standard: `[In temporäre Datei speichern]`	Specify the output point cloud with reduced points. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Filterausdruck Optional	`FILTER_EXPRESS`	[Ausdruck]	Ein *PDAL-Ausdruck* zur Auswahl einer Untergruppe von Objekten in den Punktwolkendaten

Fortsetzung auf der nächsten Seite

Tab. 24.285 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Begrenzungsausmal Optional	`FILTER_EXTENT`	[Ausdehnung]	Ein Kartenausschnitt zur Auswahl einer Teilmenge von Objekten in den Punktwolkendaten Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgedünnt (durch Radius)	`OUTPUT`	[Punktwolke]	Punktwolke-Ausgabedatei mit reduzierten Punkten.

Pythoncode

Algorithmus-Kennung: `pdal:thinbyradius`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Dünn (durch Auslassen von Punkten)

Erzeugt eine ausgedünnte Version der Punktwolke, indem nur jeder N-te Punkt beibehalten wird (reduziert die Anzahl der Punkte, indem nahe gelegene Punkte ausgelassen werden).

Siehe auch

Dünn (durch Abtastradius)

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Punktwolke-Eingabelayer, von dem eine ausgedünnte Version erstellt werden soll
Anzahl der zu überspringenden Punkte	`POINTS_NUMBER`	[numerisch: Ganzzahl] Standard: 1	Behält nur jeden N-ten Punkt aus dem Eingabelayer
Ausgedünnt (durch Dezimierung)	`OUTPUT`	[Punktwolke] Standard: `[In temporäre Datei speichern]`	Specify the output point cloud with reduced points. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Filterausdruck Optional	`FILTER_EXPRESS`	[Ausdruck]	Ein *PDAL-Ausdruck* zur Auswahl einer Untergruppe von Objekten in den Punktwolkendaten

Fortsetzung auf der nächsten Seite

Tab. 24.288 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Begrenzungsausmaß Optional	`FILTER_EXTENT`	[Ausdehnung]	Ein Kartenausschnitt zur Auswahl einer Teilmenge von Objekten in den Punktwolkendaten Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgedünnt (durch Dezimierung)	`OUTPUT`	[Punktwolke]	Punktwolke-Ausgabedatei mit reduzierten Punkten.

Pythoncode

Algorithmus-Kennung: `pdal:thinbydecimate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Kacheln

Erzeugt Kacheln aus den eingegebenen Punktwolkendateien; dies ist die Empfehlung für die beste Leistung (bei der Anzeige oder Analyse) mit solchen Datensätzen in QGIS.

Siehe auch

Virtuelle Punktwolke aufbauen (VPC), *COPC erzeugen*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`LAYERS`	[Punktwolke] [Liste]	Ein oder mehrere Punktwolke-Eingabelayer, aus denen Kacheln erstellt werden sollen
Kachellänge	`LENGTH`	[numerisch: Double] Standard: 1000,0	Größe des Randes jeder erzeugten Kachel
Ausgabeverzeichnis	`OUTPUT`	[Verzeichnis] Standard: `[In temporärem Verzeichnis speichern]`	Specify the folder to store the generated tiles. *One of*: • In einem temporären Verzeichnis speichern • In Verzeichnis speichern

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
KBS zuweisen Optional	`CRS`	[KBS]	Das KBS, das auf den Layer angewendet werden soll

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabeverzeichnis	`OUTPUT`	[Verzeichnis]	Ausgabeordner, der die aus den Eingabedateien erzeugten Kacheln enthält.

Pythoncode

Algorithmus-Kennung: `pdal:tile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.17 Point Cloud Extraction

Achtung

Running these algorithms requires QGIS installed with PDAL >= 2.5.0 (see *Help* ► *About* menu).

Boundary

Exports a polygon file containing point cloud layer boundary. It may contain holes and it may be a multi-part polygon.

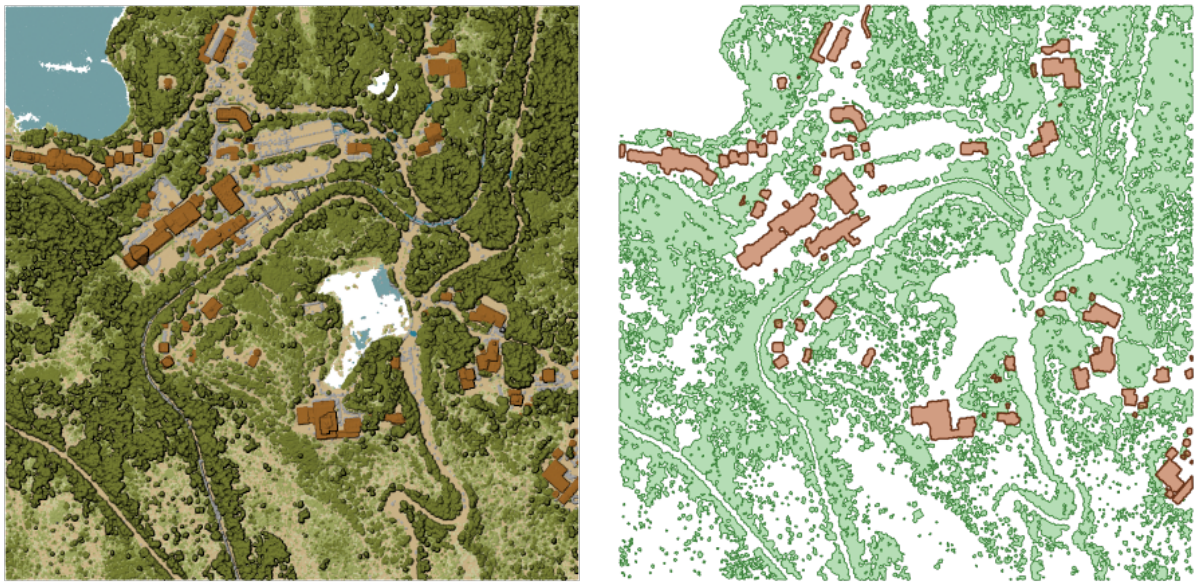


Abb. 24.46: Extracting high vegetation and building polygons from an input point cloud layer

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Input point cloud layer to calculate boundary for
Resolution of cells used to calculate boundary Optional	`RESOLUTION`	[numerisch: Double]	Resolution of cells used to calculate boundary
Minimal number of points in a cell to consider cell occupied Optional	`THRESHOLD`	[numerisch: Ganzzahl]	Minimal number of points in a cell to consider cell occupied
Boundary	`OUTPUT`	[Vektor: Polygon] Standard: `[In temporäre Datei speichern]`	Specify the point cloud file to use as output. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Filterausdruck Optional	`FILTER_EXPRESS`	[Ausdruck]	Ein *PDAL-Ausdruck* zur Auswahl einer Untergruppe von Objekten in den Punktwolkendaten
Begrenzungsausmal Optional	`FILTER_EXTENT`	[Ausdehnung]	Ein Kartenausschnitt zur Auswahl einer Teilmenge von Objekten in den Punktwolkendaten Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Boundary	`OUTPUT`	[Vektor: Polygon]	Output polygon vector layer with the point cloud boundary. Currently supported format is `.GPKG`.

Pythoncode

Algorithm ID: `pdal:boundary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Density

Exports a raster file based on the number of points within each raster cell - useful for quality checking of point cloud datasets.

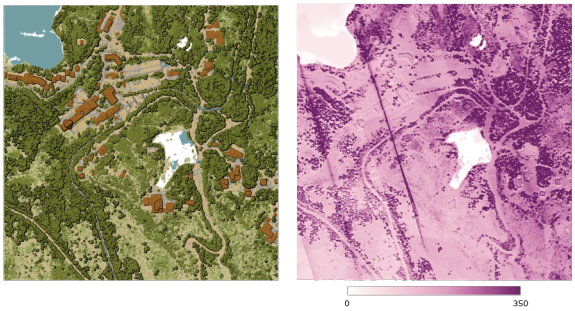


Abb. 24.47: Point density (number of points per 2x2 m) as a raster

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Layer der zu exportierenden Punktwolke
Auflösung des Rasters	`RESOLUTION`	[numerisch: Double] Standard: 1,0	Zellengröße des Ausgaberasters
Kachelgröße für parallele Durchläufe	`TILE_SIZE`	[numerisch: Ganzzahl] Standard: 1000	Size of the tiles to split the data into for parallel runs

Fortsetzung auf der nächsten Seite

Tab. 24.296 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Density	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the raster file to export the data to. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Filterausdruck Optional	`FILTER_EXPRESS`	[Ausdruck]	Ein *PDAL-Ausdruck* zur Auswahl einer Untergruppe von Objekten in den Punktwolkendaten
Begrenzungsausmal Optional	`FILTER_EXTENT`	[Ausdehnung]	Ein Kartenausschnitt zur Auswahl einer Teilmenge von Objekten in den Punktwolkendaten Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
X-Ursprung einer Kachel für parallele Läufe Optional	`ORIGIN_X`	[numerisch: Double]	
Y-Ursprung einer Kachel für parallele Läufe Optional	`ORIGIN_Y`	[numerisch: Double]	

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Density	`OUTPUT`	[Raster]	Output raster layer with number of points within each cell. Currently supported format is `.TIF`.

Pythoncode

Algorithm ID: `pdal:density`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Filter

Extracts point from the input point cloud which match PDAL expression and/or are inside of a cropping rectangle.

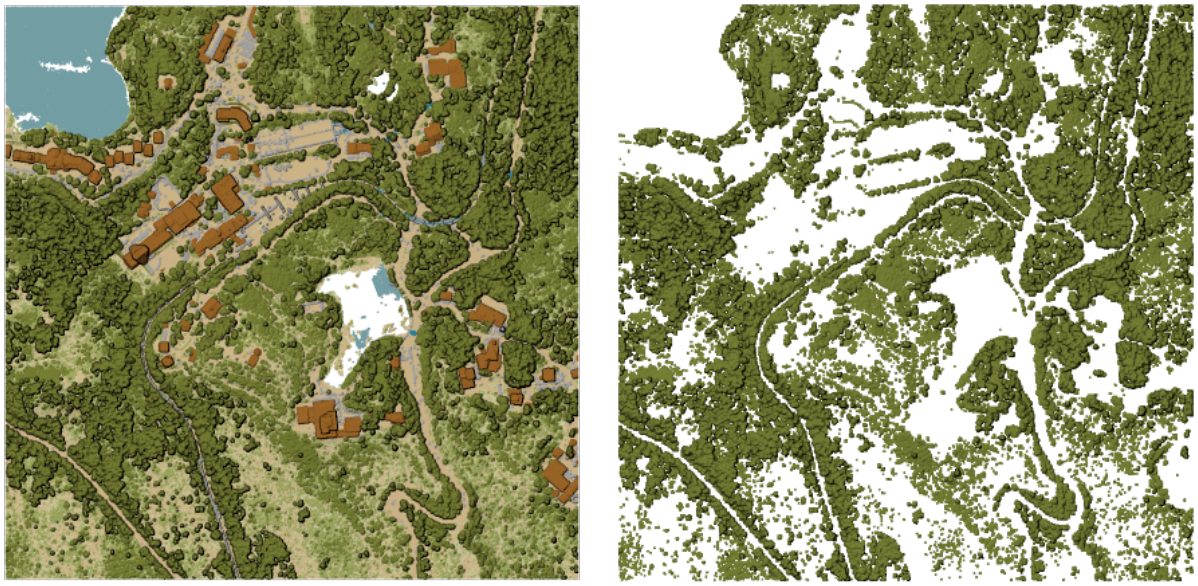


Abb. 24.48: Filtering of high vegetation class from an input point cloud layer

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Punktwolke]	Layer der zu exportierenden Punktwolke
Filterausdruck Optional	`FILTER_EXPRESS`	[Ausdruck]	Ein *PDAL-Ausdruck* zur Auswahl einer Untergruppe von Objekten in den Punktwolkendaten
Begrenzungsausmal Optional	`FILTER_EXTENT`	[Ausdehnung]	Ein Kartenausschnitt zur Auswahl einer Teilmenge von Objekten in den Punktwolkendaten Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Filtered	`OUTPUT`	[Punktwolke] Standard: `[In temporäre Datei speichern]`	Specify the point cloud file to export the data to. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Filtered	`OUTPUT`	[Punktwolke]	Output point cloud layer with the filtered features of the input point cloud layer. Currently supported formats are `.LAS`, `.LAZ`, `.COPC.LAZ` and `.VPC`.

Pythoncode

Algorithm ID: `pdal:filter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.18 Raster analysis

Cell stack percent rank from value

Calculates the cell-wise percentrank value of a stack of rasters based on a single input value and writes them to an output raster.

At each cell location, the specified value is ranked among the respective values in the stack of all overlaid and sorted cell values from the input rasters. For values outside of the stack value distribution, the algorithm returns NoData because the value cannot be ranked among the cell values.

There are two methods for percentile calculation:

- Inclusive linear interpolation (PERCENTRANK.INC)
- Exclusive linear interpolation (PERCENTRANK.EXC)

The linear interpolation methods return the unique percent rank for different values. Both interpolation methods follow their counterpart methods implemented by LibreOffice or Microsoft Excel.

The output raster's extent and resolution is defined by a reference raster. Input raster layers that do not match the cell size of the reference raster layer will be resampled using nearest neighbor resampling. NoData values in any of the input layers will result in a NoData cell output if the „Ignore NoData values“ parameter is not set. The output raster data type will always be `Float32`.

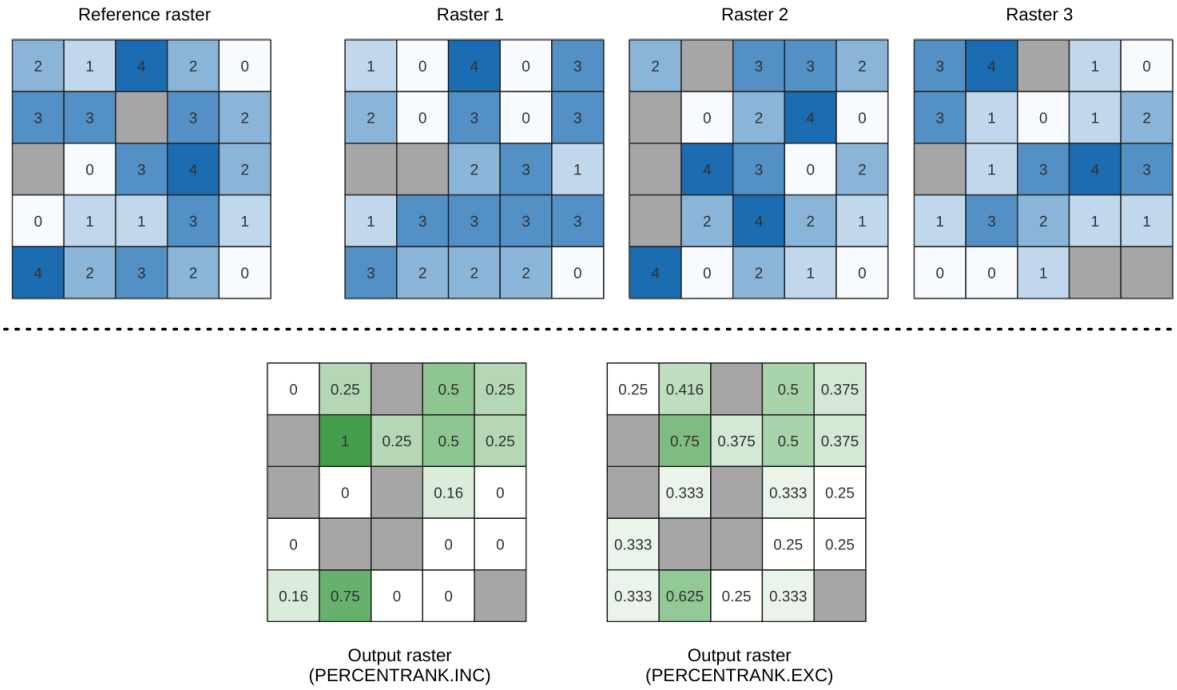


Abb. 24.49: Percent ranking Value = 1. `NoData` cells (grey) are ignored.

Siehe auch

Cell stack percentile, *Cell stack percentrank from raster layer*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[raster] [list]	Raster layers to evaluate. If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters
Methode	`METHOD`	[Aufzählung] Standard: 0	Method for percentile calculation: • 0 — Inclusive linear interpolation (PERCENTRANK.INC) • 1 — Exclusive linear interpolation (PERCENTRANK.EXC)
Value	`VALUE`	[numerisch: Double] Default: 10.0	Value to rank among the respective values in the stack of all overlaid and sorted cell values from the input rasters
Ignore NoData values	`IGNORE_NODATA`	[Boolean] Standard: True	If unchecked, any NoData cells in the input layers will result in a NoData cell in the output raster
Referenz-Layer	`REFERENCE_LAYE`	[Raster]	The reference layer for the output layer creation (extent, CRS, pixel dimensions)

Fortsetzung auf der nächsten Seite

Tab. 24.301 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value	`OUTPUT_NODATA_`	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Raster]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:cellstackpercentrankfromvalue`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Cell stack percentile

Calculates the cell-wise percentile value of a stack of rasters and writes the results to an output raster. The percentile to return is determined by the percentile input value (ranges between 0 and 1). At each cell location, the specified percentile is obtained using the respective value from the stack of all overlaid and sorted cell values of the input rasters.

There are three methods for percentile calculation:

- Nearest rank: returns the value that is nearest to the specified percentile
- Inclusive linear interpolation (PERCENTRANK.INC)
- Exclusive linear interpolation (PERCENTRANK.EXC)

The linear interpolation methods return the unique values for different percentiles. Both interpolation methods follow their counterpart methods implemented by LibreOffice or Microsoft Excel.

The output raster's extent and resolution is defined by a reference raster. Input raster layers that do not match the cell size of the reference raster layer will be resampled using nearest neighbor resampling. NoData values in any of the input layers will result in a NoData cell output if the „Ignore NoData values“ parameter is not set. The output raster data type will always be `Float32`.

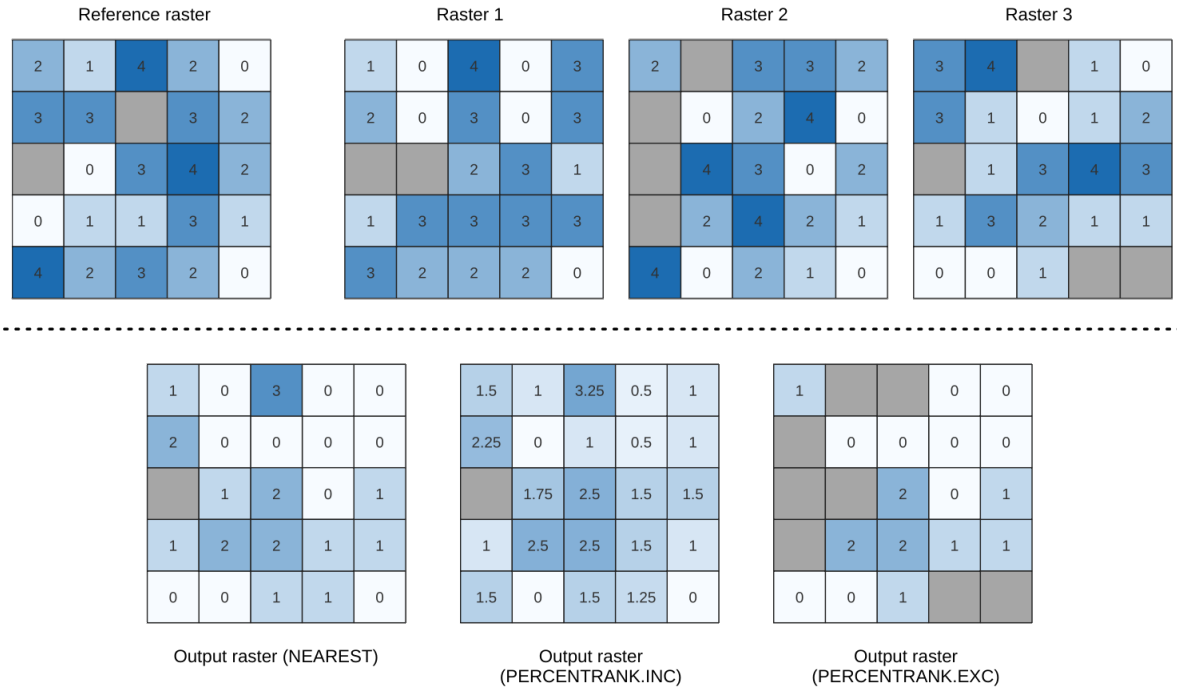


Abb. 24.50: Percentile = 0.25. `NoData` cells (grey) are ignored.

Siehe auch

Cell stack percentile, *Cell stack percentrank from raster layer*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[raster] [list]	Raster layers to evaluate. If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters
Methode	`METHOD`	[Aufzählung] Standard: 0	Method for percentile calculation: • 0 — Nearest rank: returns the value that is nearest to the specified percentile • 1 — Inclusive linear interpolation (PERCENTILE.INC) • 2 — Exclusive linear interpolation (PERCENTILE.EXC)
Percentile	`VALUE`	[numerisch: Double] Default: 0.25	Value to rank among the respective values in the stack of all overlaid and sorted cell values from the input rasters. Between 0 and 1.
Ignore NoData values	`IGNORE_NODATA`	[Boolean] Standard: True	If unchecked, any NoData cells in the input layers will result in a NoData cell in the output raster
Referenz-Layer	`REFERENCE_LAYE`	[Raster]	The reference layer for the output layer creation (extent, CRS, pixel dimensions)
Ausgabelayer	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value	`OUTPUT_NODATA_`	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ,‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Raster]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:cellstackpercentile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Cell stack percentrank from raster layer

Calculates the cell-wise percentrank value of a stack of rasters based on an input value raster and writes them to an output raster.

At each cell location, the current value of the value raster is ranked among the respective values in the stack of all overlaid and sorted cell values of the input rasters. For values outside of the the stack value distribution, the algorithm returns NoData because the value cannot be ranked among the cell values.

There are two methods for percentile calculation:

- Inclusive linear interpolation (PERCENTRANK.INC)
- Exclusive linear interpolation (PERCENTRANK.EXC)

The linear interpolation methods return the unique values for different percentiles. Both interpolation methods follow their counterpart methods implemented by LibreOffice or Microsoft Excel.

The output raster's extent and resolution is defined by a reference raster. Input raster layers that do not match the cell size of the reference raster layer will be resampled using nearest neighbor resampling. NoData values in any of the input layers will result in a NoData cell output if the „Ignore NoData values“ parameter is not set. The output raster data type will always be `Float32`.

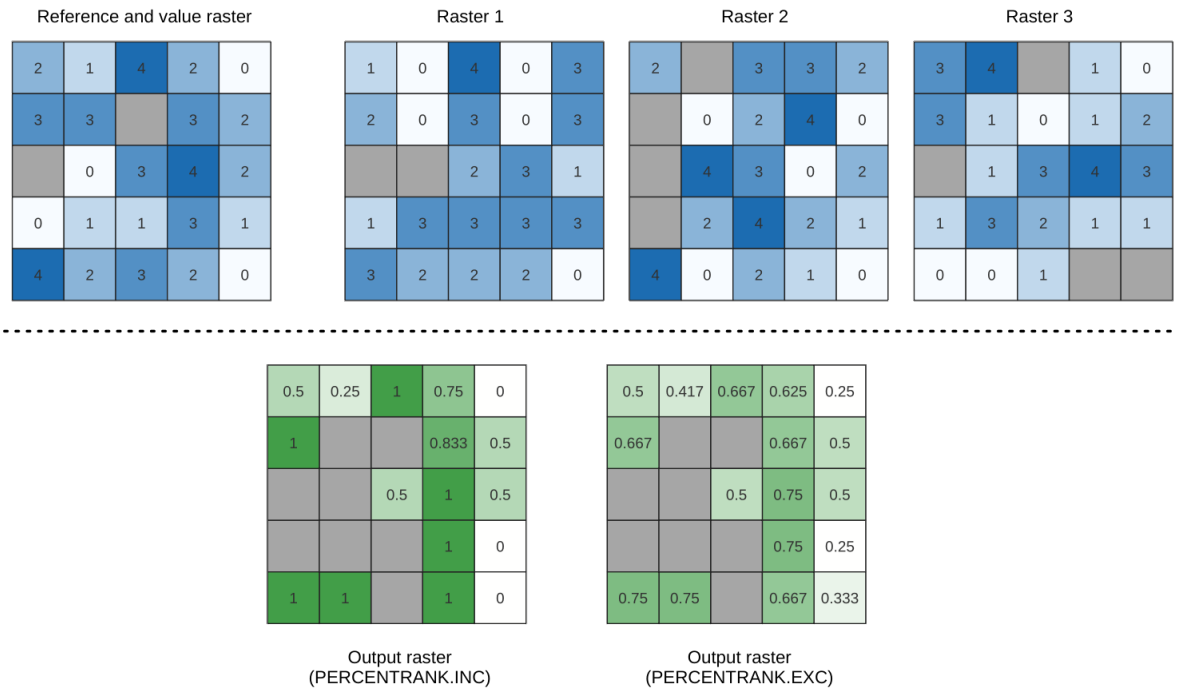


Abb. 24.51: Ranking the value raster layer cells. `NoData` cells (grey) are ignored.

Siehe auch

Cell stack percentile, *Cell stack percent rank from value*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[raster] [list]	Raster layers to evaluate. If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters
Value raster layer	`INPUT_VALUE_RA`	[Raster]	The layer to rank the values among the stack of all overlaid layers
Value raster band	`VALUE_RASTER_B`	[numerisch: Ganzzahl] Standard: 1	Band of the „value raster layer“ to compare to
Methode	`METHOD`	[Aufzählung] Standard: 0	Method for percentile calculation: • 0 — Inclusive linear interpolation (PERCENTRANK.INC) • 1 — Exclusive linear interpolation (PERCENTRANK.EXC)
Ignore NoData values	`IGNORE_NODATA`	[Boolean] Standard: True	If unchecked, any NoData cells in the input layers will result in a NoData cell in the output raster

Fortsetzung auf der nächsten Seite

Tab. 24.307 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Referenz-Layer	`REFERENCE_LAYE`	[Raster]	The reference layer for the output layer creation (extent, CRS, pixel dimensions)
Ausgabelayer	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value	`OUTPUT_NODATA_`	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Raster]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:cellstackpercentrankfromrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Cell statistics

Computes per-cell statistics based on input raster layers and for each cell writes the resulting statistics to an output raster. At each cell location, the output value is defined as a function of all overlaid cell values of the input rasters.

By default, a NoData cell in ANY of the input layers will result in a NoData cell in the output raster. If the *Ignore NoData values* option is checked, then NoData inputs will be ignored in the statistic calculation. This may result in NoData output for locations where all cells are NoData.

The *Reference layer* parameter specifies an existing raster layer to use as a reference when creating the output raster. The output raster will have the same extent, CRS, and pixel dimensions as this layer.

Calculation details: Input raster layers that do not match the cell size of the reference raster layer will be resampled using `nearest neighbor resampling`. The output raster data type will be set to the most complex data type present in the input datasets except when using the functions `Mean`, `Standard deviation` and `Variance` (data type is always `Float32` or `Float64` depending on input float type) or `Count` and `Variety` (data type is always `Int32`).

- `Count`: The count statistic will always result in the number of cells without NoData values at the current cell location.
- `Median`: If the number of input layers is even, the median will be calculated as the arithmetic mean of the two middle values of the ordered cell input values.
- `Minority/Majority`: If no unique minority or majority could be found, the result is NoData, except all input cell values are equal.

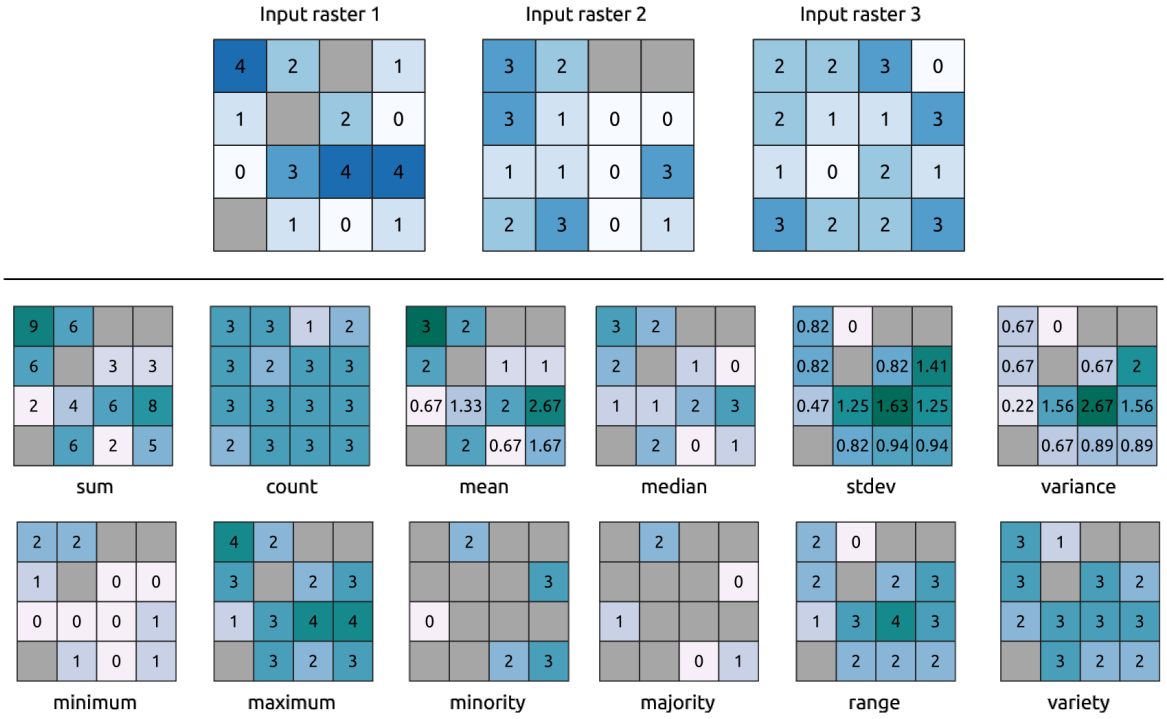


Abb. 24.52: Example with all the statistic functions. `NoData` cells (grey) are taken into account.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[raster] [list]	Input raster layers
Statistic	`STATISTIC`	[Aufzählung] Standard: 0	Available statistics. Options: • 0 — Sum • 1 — Count • 2 — Mean • 3 — Median • 4 — Standard deviation • 5 — Variance • 6 — Minimum • 7 — Maximum • 8 — Minority (least common value) • 9 — Majority (most common value) • 10 — Range (max - min) • 11 — Variety (unique value count)
Ignore NoData values	`IGNORE_NODATA`	[Boolean] Standard: True	Calculate statistics also for all cells stacks, ignoring NoData occurrence.
Referenz-Layer	`REF_LAYER`	[Raster]	The reference layer to create the output layer from (extent, CRS, pixel dimensions)
Ausgabelayer	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value Optional	`OUTPUT_NO_DATA`	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
CRS authority identifier	`CRS_AUTHID`	[KBS]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Output raster	`OUTPUT`	[Raster]	Output raster layer containing the result
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer

Pythoncode

Algorithm ID: `native:cellstatistics`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Equal to frequency

Evaluates on a cell-by-cell basis the frequency (number of times) the values of an input stack of rasters are equal to the value of a value layer. The output raster extent and resolution are defined by the input raster layer and is always of `Int32` type.

If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters - use GDAL to use other bands in the analysis. The output NoData value can be set manually.

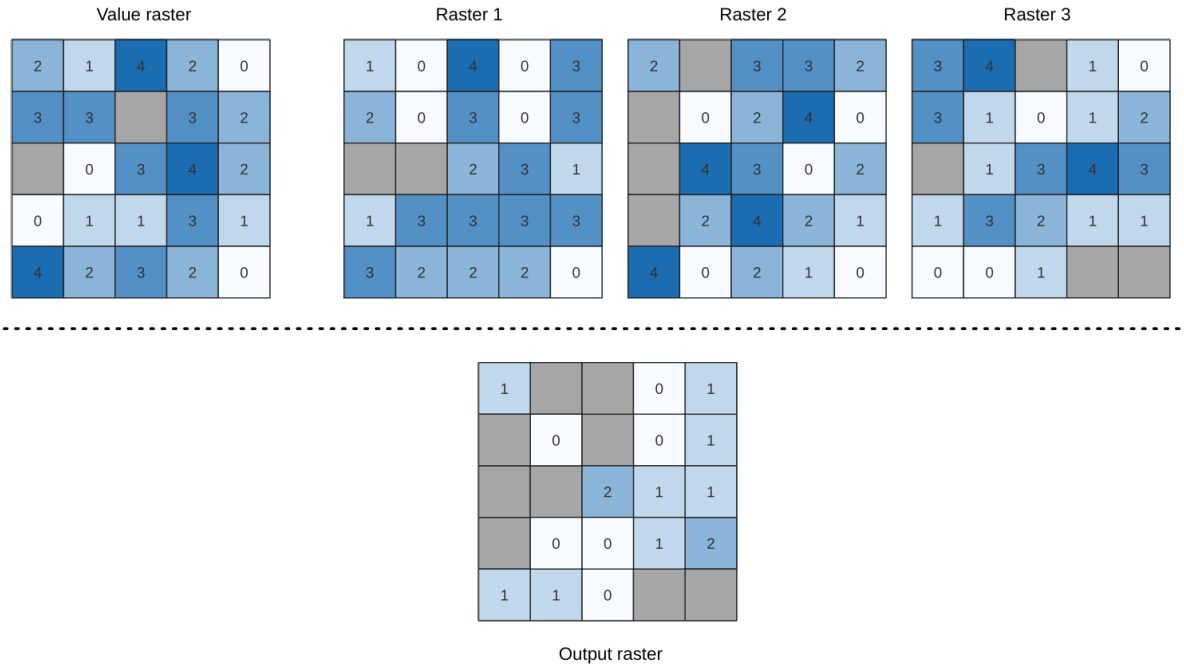


Abb. 24.53: For each cell in the output raster, the value represents the number of times that the corresponding cells in the list of rasters are the same as the value raster. `NoData` cells (grey) are taken into account.

Siehe auch

Greater than frequency, *Less than frequency*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input value raster	`INPUT_VALUE_RA`	[Raster]	The input value layer serves as reference layer for the sample layers
Value raster band	`INPUT_VALUE_RA`	[raster band] Default: The first band of the raster layer	Select the band you want to use as sample
Input raster layers	`INPUT_RASTERS`	[raster] [list]	Raster layers to evaluate. If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters
Ignore NoData values	`IGNORE_NODATA`	[Boolean] Standard: falsch	If unchecked, any NoData cells in the value raster or the data layer stack will result in a NoData cell in the output raster
Ausgabelayer	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value Optional	`OUTPUT_NO_DATA`	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Raster]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Count of cells with equal value occurrences	`FOUND_LOCATION`	[numerisch: Ganzzahl]	
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer
Mean frequency at valid cell locations	`MEAN_FREQUENCY`	[numerisch: Double]	
Count of value occurrences	`OCCURRENCE_COU`	[numerisch: Ganzzahl]	
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer

Pythoncode

Algorithm ID: `native:equaltofrequency`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Fuzzify raster (gaussian membership)

Transforms an input raster to a fuzzified raster by assigning a membership value to each pixel, using a Gaussian membership function. Membership values range from 0 to 1. In the fuzzified raster, a value of 0 implies no membership of the defined fuzzy set, whereas a value of 1 means full membership. The gaussian membership function is defined as $\mu(x) = e^{-f1*(x-f2)^2}$, where *f1* is the spread and *f2* the midpoint.

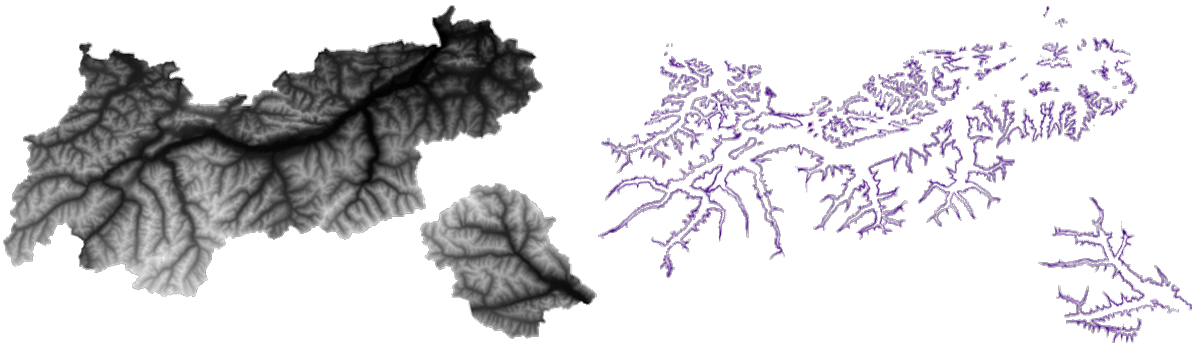


Abb. 24.54: Fuzzify raster example. Input raster source: Land Tirol - data.tirol.gv.at.

Siehe auch

Fuzzify raster (large membership), *Fuzzify raster (linear membership)*, *Fuzzify raster (near membership)*, *Fuzzify raster (power membership)*, *Fuzzify raster (small membership)*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Raster	`INPUT`	[Raster]	Input raster layer
Band Number	`BAND`	[raster band] Default: The first band of the raster layer	If the raster is multiband, choose the band that you want to fuzzify.
Function midpoint	`FUZZYMIDPOINT`	[numerisch: Double] Default: 10.0	Midpoint of the gaussian function
Function spread	`FUZZYSPREAD`	[numerisch: Double] Default: 0.01	Spread of the gaussian function
Fuzzified raster	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	CREATION_OPTIO (for QGIS <= 3.42, this was CREATE_OPTIONS	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Fuzzified raster	OUTPUT	[wie Eingabe]	Output raster layer containing the result
CRS authority identifier	CRS_AUTHID	[KBS]	The coordinate reference system of the output raster layer
Extent	EXTENT	[Text]	The spatial extent of the output raster layer
Width in pixels	WIDTH_IN_PIXEL	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	HEIGHT_IN_PIXE	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	TOTAL_PIXEL_CO	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:fuzzifyrastergaussianmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Fuzzify raster (large membership)

Transforms an input raster to a fuzzified raster by assigning a membership value to each pixel, using a Large membership function. Membership values range from 0 to 1. In the fuzzified raster, a value of 0 implies no membership of the defined fuzzy set, whereas a value of 1 means full membership. The large membership function is defined as $\mu(x) = \frac{1}{1 + \left(\frac{x}{f2}\right)^{-f1}}$, where *f1* is the spread and *f2* the midpoint.

Siehe auch

Fuzzify raster (gaussian membership), *Fuzzify raster (linear membership)*, *Fuzzify raster (near membership)*, *Fuzzify raster (power membership)*, *Fuzzify raster (small membership)*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Raster	`INPUT`	[Raster]	Input raster layer
Band Number	`BAND`	[raster band] Default: The first band of the raster layer	If the raster is multiband, choose the band that you want to fuzzify.
Function midpoint	`FUZZYMIDPOINT`	[numerisch: Double] Default: 50.0	Midpoint of the large function
Function spread	`FUZZYSPREAD`	[numerisch: Double] Default: 5.0	Spread of the large function
Fuzzified raster	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Fuzzified raster	`OUTPUT`	[wie Eingabe]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[KBS]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer

Fortsetzung auf der nächsten Seite

Tab. 24.321 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:fuzzifyrasterlargemembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Fuzzify raster (linear membership)

Transforms an input raster to a fuzzified raster by assigning a membership value to each pixel, using a Linear membership function. Membership values range from 0 to 1. In the fuzzified raster, a value of 0 implies no membership of the defined fuzzy set, whereas a value of 1 means full membership. The linear function is defined as $\mu(X)\begin{cases}0 & x \leq a \\ \frac{x-a}{b-a} & a < x < b \\ 1 & x \geq b\end{cases}$, where a is the low bound and b the high bound. This equation assigns membership values using a linear transformation for pixel values between the low and high bounds. Pixels values smaller than the low bound are given 0 membership whereas pixel values greater than the high bound are given 1 membership.

Siehe auch

Fuzzify raster (gaussian membership), *Fuzzify raster (large membership)*, *Fuzzify raster (near membership)*, *Fuzzify raster (power membership)*, *Fuzzify raster (small membership)*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Raster	`INPUT`	[Raster]	Input raster layer
Band Number	`BAND`	[raster band] Default: The first band of the raster layer	If the raster is multiband, choose the band that you want to fuzzify.
Low fuzzy membership bound	`FUZZYLOWBOUND`	[numerisch: Double] Default: 0.0	Low bound of the linear function
High fuzzy membership bound	`FUZZYHIGHBOUND`	[numerisch: Double] Standard: 1,0	High bound of the linear function

Fortsetzung auf der nächsten Seite

Tab. 24.322 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Fuzzified raster	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Fuzzified raster	`OUTPUT`	[wie Eingabe]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[KBS]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:fuzzifyrasterlinearmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Fuzzify raster (near membership)

Transforms an input raster to a fuzzified raster by assigning a membership value to each pixel, using a Near membership function. Membership values range from 0 to 1. In the fuzzified raster, a value of 0 implies no membership of the defined fuzzy set, whereas a value of 1 means full membership. The near membership function is defined as $\mu(x) = \frac{1}{1 + f1 * (x - f2)^2}$, where *f1* is the spread and *f2* the midpoint.

Siehe auch

Fuzzify raster (gaussian membership), *Fuzzify raster (large membership)*, *Fuzzify raster (linear membership)*, *Fuzzify raster (power membership)*, *Fuzzify raster (small membership)*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Raster	`INPUT`	[Raster]	Input raster layer
Band Number	`BAND`	[raster band] Default: The first band of the raster layer	If the raster is multiband, choose the band that you want to fuzzify.
Function midpoint	`FUZZYMIDPOINT`	[numerisch: Double] Default: 50.0	Midpoint of the near function
Function spread	`FUZZYSPREAD`	[numerisch: Double] Default: 0.01	Spread of the near function
Fuzzified raster	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Fuzzified raster	`OUTPUT`	[wie Eingabe]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[KBS]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:fuzzifyrasternearmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Fuzzify raster (power membership)

Transforms an input raster to a fuzzified raster by assigning a membership value to each pixel, using a Power membership function. Membership values range from 0 to 1. In the fuzzified raster, a value of 0 implies no membership of the defined fuzzy set, whereas a value of 1 means full membership. The power function is defined as $\mu(x)\begin{cases}0 & x \le a \\ \left(\frac{x-a}{b-a}\right)^{f1} & a < x < b \\ 1 & x \ge b\end{cases}$, where a is the low bound, b is the high bound, and $f1$ the exponent. This equation assigns membership values using the power transformation for pixel values between the low and high bounds. Pixels values smaller than the low bound are given 0 membership whereas pixel values greater than the high bound are given 1 membership.

Siehe auch

Fuzzify raster (gaussian membership), *Fuzzify raster (large membership)*, *Fuzzify raster (linear membership)*, *Fuzzify raster (near membership)*, *Fuzzify raster (small membership)*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Raster	`INPUT`	[Raster]	Input raster layer
Band Number	`BAND`	[raster band] Default: The first band of the raster layer	If the raster is multiband, choose the band that you want to fuzzify.
Low fuzzy membership bound	`FUZZYLOWBOUND`	[numerisch: Double] Default: 0.0	Low bound of the power function
High fuzzy membership bound	`FUZZYHIGHBOUND`	[numerisch: Double] Standard: 1,0	High bound of the power function
High fuzzy membership bound	`FUZZYEXPONENT`	[numerisch: Double] Default: 2.0	Exponent of the power function
Fuzzified raster	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Fuzzified raster	`OUTPUT`	[wie Eingabe]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[KBS]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer

Fortsetzung auf der nächsten Seite

Tab. 24.330 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:fuzzifyrasterpowermembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Fuzzify raster (small membership)

Transforms an input raster to a fuzzified raster by assigning a membership value to each pixel, using a Small membership function. Membership values range from 0 to 1. In the fuzzified raster, a value of 0 implies no membership of the defined fuzzy set, whereas a value of 1 means full membership. The small membership function is defined as $\mu(x) = \frac{1}{1 + \left(\frac{x}{f2}\right)^{f1}}$, where *f1* is the spread and *f2* the midpoint.

Siehe auch

Fuzzify raster (gaussian membership), *Fuzzify raster (large membership)* *Fuzzify raster (linear membership)*, *Fuzzify raster (near membership)*, *Fuzzify raster (power membership)*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Raster	`INPUT`	[Raster]	Input raster layer
Band Number	`BAND`	[raster band] Default: The first band of the raster layer	If the raster is multiband, choose the band that you want to fuzzify.
Function midpoint	`FUZZYMIDPOINT`	[numerisch: Double] Default: 50.0	Midpoint of the small function
Function spread	`FUZZYSPREAD`	[numerisch: Double] Default: 5.0	Spread of the small function

Fortsetzung auf der nächsten Seite

Tab. 24.331 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Fuzzified raster	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Fuzzified raster	`OUTPUT`	[wie Eingabe]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[KBS]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:fuzzifyrastersmallmembership`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Greater than frequency

Evaluates on a cell-by-cell basis the frequency (number of times) the values of an input stack of rasters are equal to the value of a value raster. The output raster extent and resolution is defined by the input raster layer and is always of `Int32` type.

If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters - use GDAL to use other bands in the analysis. The output NoData value can be set manually.

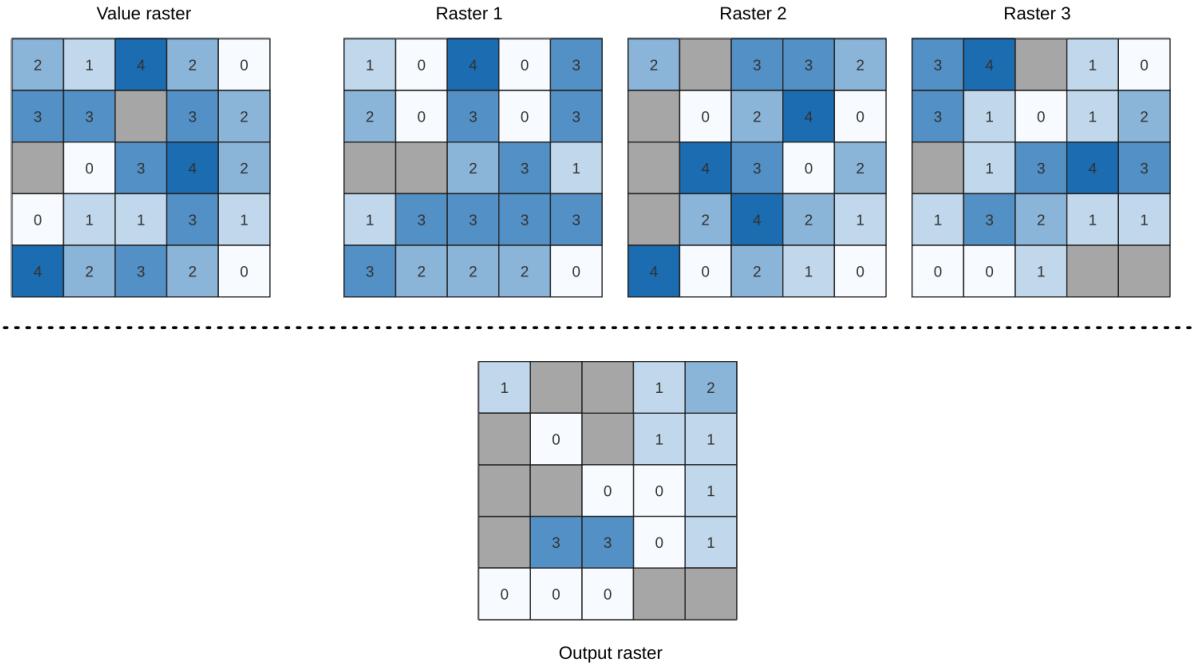


Abb. 24.55: For each cell in the output raster, the value represents the number of times that the corresponding cells in the list of rasters are greater than the value raster. `NoData` cells (grey) are taken into account.

Siehe auch

Equal to frequency, *Less than frequency*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input value raster	`INPUT_VALUE_RA`	[Raster]	The input value layer serves as reference layer for the sample layers
Value raster band	`INPUT_VALUE_RA`	[raster band] Default: The first band of the raster layer	Select the band you want to use as sample
Input raster layers	`INPUT_RASTERS`	[raster] [list]	Raster layers to evaluate. If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters

Fortsetzung auf der nächsten Seite

Tab. 24.334 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Ignore NoData values	`IGNORE_NODATA`	[Boolean] Standard: falsch	If unchecked, any NoData cells in the value raster or the data layer stack will result in a NoData cell in the output raster
Ausgabelayer	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value Optional	`OUTPUT_NO_DATA`	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Raster]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Count of cells with equal value occurrences	`FOUND_LOCATION`	[numerisch: Ganzzahl]	
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer
Mean frequency at valid cell locations	`MEAN_FREQUENCY`	[numerisch: Double]	
Count of value occurrences	`OCCURRENCE_COU`	[numerisch: Ganzzahl]	
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer

Pythoncode

Algorithm ID: `native:greaterthanfrequency`

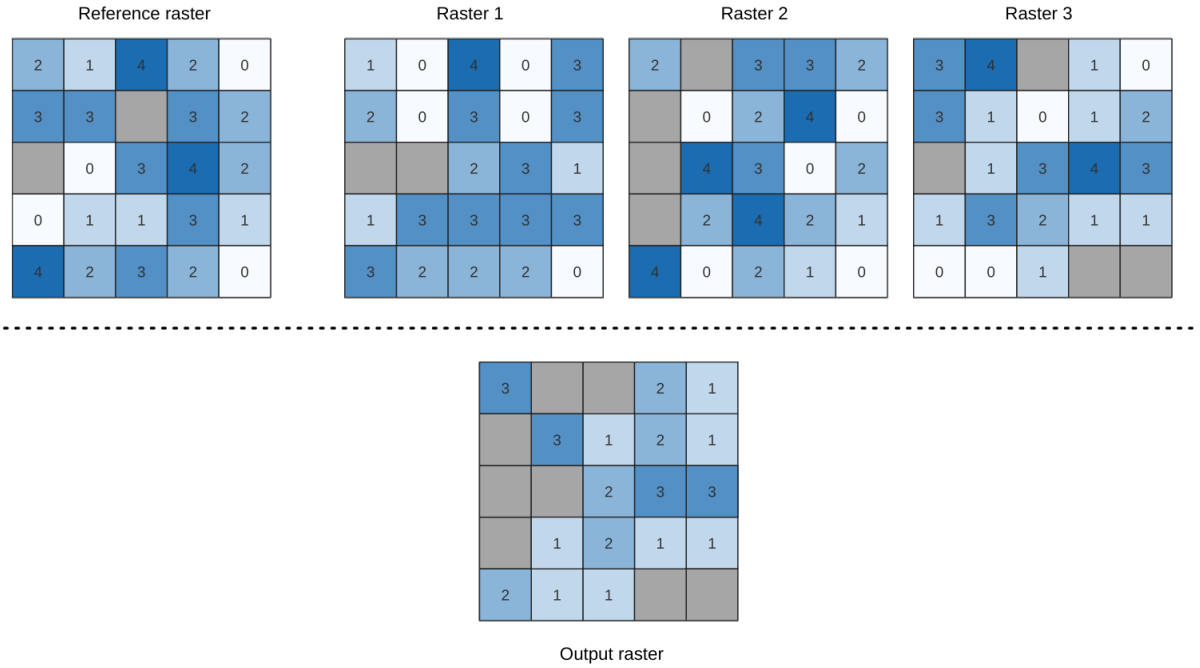
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Highest position in raster stack

Evaluates on a cell-by-cell basis the position of the raster with the highest value in a stack of rasters. Position counts start with 1 and range to the total number of input rasters. The order of the input rasters is relevant for the algorithm. If multiple rasters feature the highest value, the first raster will be used for the position value.

If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters - use GDAL to use other bands in the analysis. Any NoData cells in the raster layer stack will result in a NoData cell in the output raster unless the „ignore NoData“ parameter is checked. The output NoData value can be set manually. The output rasters extent and resolution is defined by a reference raster layer and is always of `Int32` type.



Siehe auch

Lowest position in raster stack

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input raster layers	`INPUT_RASTERS`	[raster] [list]	List of raster layers to compare with
Referenz-Layer	`REFERENCE_LAYE`	[Raster]	The reference layer for the output layer creation (extent, CRS, pixel dimensions)
Ignore NoData values	`IGNORE_NODATA`	[Boolean] Standard: falsch	If unchecked, any NoData cells in the data layer stack will result in a NoData cell in the output raster
Ausgabelayer	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster containing the result. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value	`OUTPUT_NODATA_`	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Raster]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:highestpositioninrasterstack`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Less than frequency

Evaluates on a cell-by-cell basis the frequency (number of times) the values of an input stack of rasters are less than the value of a value raster. The output raster extent and resolution is defined by the input raster layer and is always of `Int32` type.

If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters - use GDAL to use other bands in the analysis. The output NoData value can be set manually.

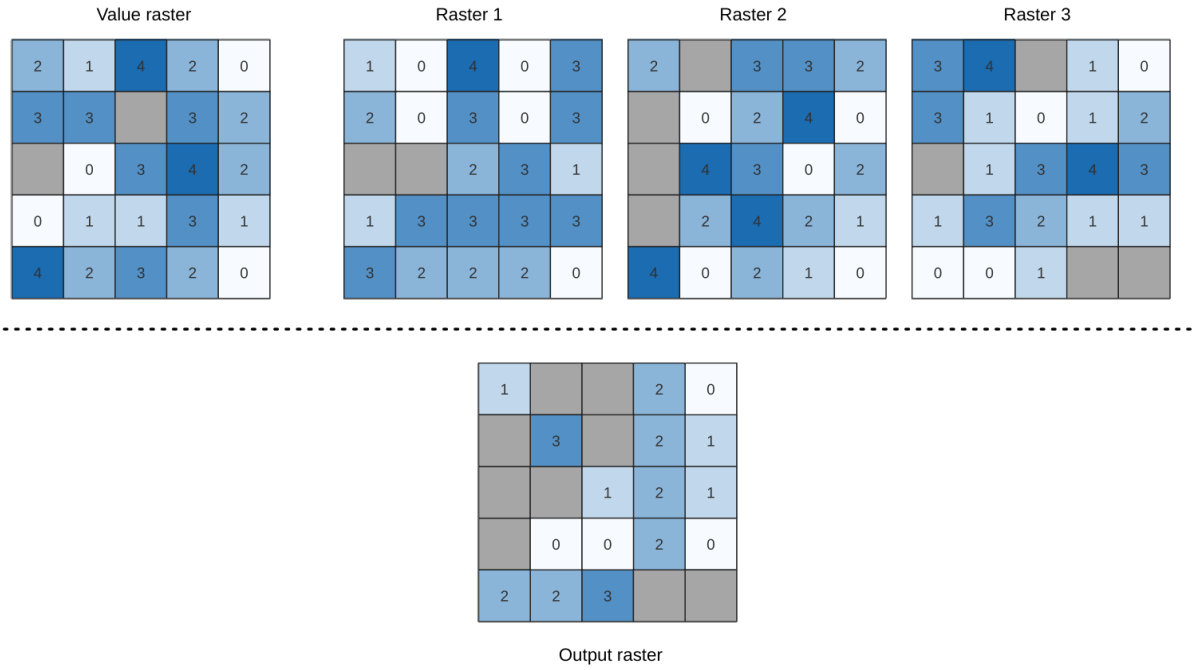


Abb. 24.56: For each cell in the output raster, the value represents the number of times that the corresponding cells in the list of rasters are less than the value raster. `NoData` cells (grey) are taken into account.

Siehe auch

Equal to frequency, *Greater than frequency*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input value raster	`INPUT_VALUE_RA`	[Raster]	The input value layer serves as reference layer for the sample layers
Value raster band	`INPUT_VALUE_RA`	[raster band] Default: The first band of the raster layer	Select the band you want to use as sample
Input raster layers	`INPUT_RASTERS`	[raster] [list]	Raster layers to evaluate. If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters
Ignore NoData values	`IGNORE_NODATA`	[Boolean] Standard: falsch	If unchecked, any NoData cells in the value raster or the data layer stack will result in a NoData cell in the output raster
Ausgabelayer	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value Optional	`OUTPUT_NO_DATA`	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Raster]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Count of cells with equal value occurrences	`FOUND_LOCATION`	[numerisch: Ganzzahl]	

Fortsetzung auf der nächsten Seite

Tab. 24.342 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer
Mean frequency at valid cell locations	`MEAN_FREQUENCY`	[numerisch: Double]	
Count of value occurrences	`OCCURRENCE_COU`	[numerisch: Ganzzahl]	
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer

Pythoncode

Algorithm ID: `native:lessthanfrequency`

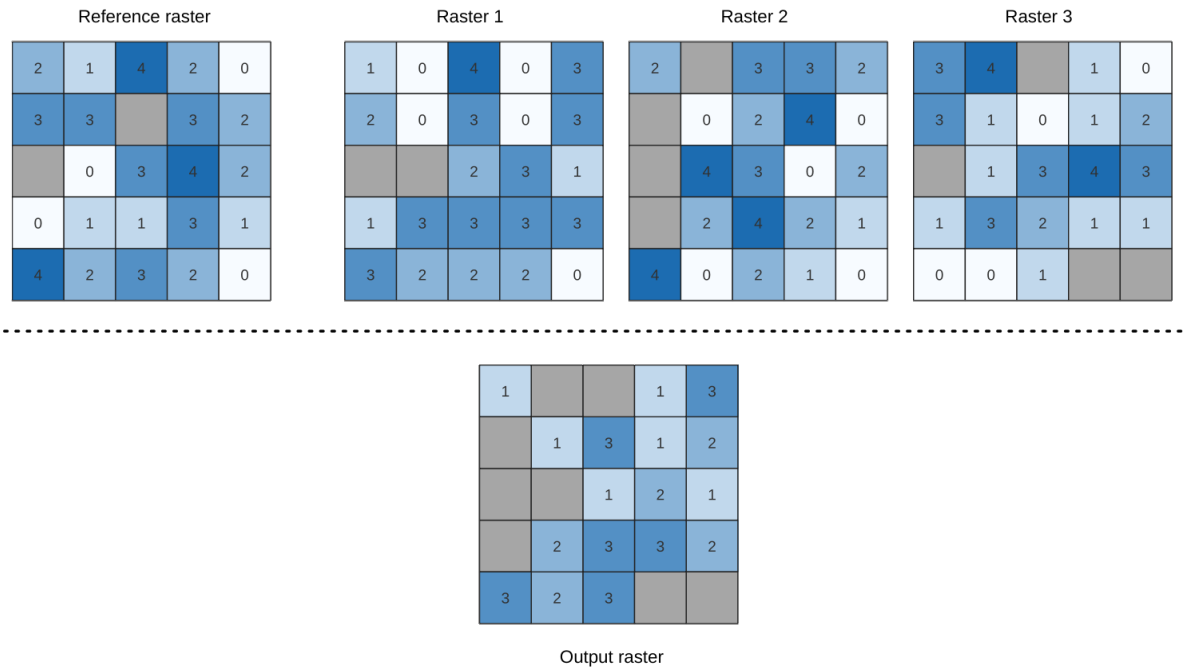
```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Lowest position in raster stack

Evaluates on a cell-by-cell basis the position of the raster with the lowest value in a stack of rasters. Position counts start with 1 and range to the total number of input rasters. The order of the input rasters is relevant for the algorithm. If multiple rasters feature the lowest value, the first raster will be used for the position value.

If multiband rasters are used in the data raster stack, the algorithm will always perform the analysis on the first band of the rasters - use GDAL to use other bands in the analysis. Any NoData cells in the raster layer stack will result in a NoData cell in the output raster unless the „ignore NoData“ parameter is checked. The output NoData value can be set manually. The output rasters extent and resolution is defined by a reference raster layer and is always of `Int32` type.



Siehe auch

Highest position in raster stack

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input raster layers	`INPUT_RASTERS`	[raster] [list]	List of raster layers to compare with
Referenz-Layer	`REFERENCE_LAYE`	[Raster]	The reference layer for the output layer creation (extent, CRS, pixel dimensions)
Ignore NoData values	`IGNORE_NODATA`	[Boolean] Standard: falsch	If unchecked, any NoData cells in the data layer stack will result in a NoData cell in the output raster
Ausgabelayer	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster containing the result. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value	`OUTPUT_NODATA_`	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Raster]	Output raster layer containing the result
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Pythoncode

Algorithm ID: `native:lowestpositioninrasterstack`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster boolean AND

Calculates the boolean `AND` for a set of input rasters. If all of the input rasters have a non-zero value for a pixel, that pixel will be set to `1` in the output raster. If any of the input rasters have `0` values for the pixel it will be set to `0` in the output raster.

The reference layer parameter specifies an existing raster layer to use as a reference when creating the output raster. The output raster will have the same extent, CRS, and pixel dimensions as this layer.

By default, a NoData pixel in ANY of the input layers will result in a NoData pixel in the output raster. If the *Treat NoData values as false* option is checked, then NoData inputs will be treated the same as a `0` input value.

Siehe auch

Raster boolean OR

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[raster] [list]	List of input raster layers
Referenz-Layer	`REF_LAYER`	[Raster]	The reference layer to create the output layer from (extent, CRS, pixel dimensions)
Treat NoData values as false	`NODATA_AS_FALS`	[Boolean] Standard: falsch	Treat NoData values in the input files as 0 when performing the operation
Ausgabelayer	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster containing the result. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value	`NO_DATA`	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer

Fortsetzung auf der nächsten Seite

Tab. 24.347 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output data type	DATA_TYPE	[Aufzählung] Default: 5	Output raster data type. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 4 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).
Creation options Optional	CREATION_OPTIO (for QGIS <= 3.42, this was CREATE_OPTIONS	[Text] Default: ‚	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extent	EXTENT	[Text]	The spatial extent of the output raster layer
CRS authority identifier	CRS_AUTHID	[KBS]	The coordinate reference system of the output raster layer
Width in pixels	WIDTH_IN_PIXEL	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	HEIGHT_IN_PIXE	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	TOTAL_PIXEL_CO	[numerisch: Ganzzahl]	The count of pixels in the output raster layer
NoData pixel count	NODATA_PIXEL_C	[numerisch: Ganzzahl]	The count of NoData pixels in the output raster layer
True pixel count	TRUE_PIXEL_COU	[numerisch: Ganzzahl]	The count of True pixels (value = 1) in the output raster layer
False pixel count	FALSE_PIXEL_CO	[numerisch: Ganzzahl]	The count of False pixels (value = 0) in the output raster layer

Fortsetzung auf der nächsten Seite

Tab. 24.348 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	OUTPUT	[Raster]	Output raster layer containing the result

Pythoncode

Algorithm ID: native:rasterbooleanand

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster boolean OR

Calculates the boolean OR for a set of input rasters. If all of the input rasters have a zero value for a pixel, that pixel will be set to 0 in the output raster. If any of the input rasters have 1 values for the pixel it will be set to 1 in the output raster.

The reference layer parameter specifies an existing raster layer to use as a reference when creating the output raster. The output raster will have the same extent, CRS, and pixel dimensions as this layer.

By default, a NoData pixel in ANY of the input layers will result in a NoData pixel in the output raster. If the *Treat NoData values as false* option is checked, then NoData inputs will be treated the same as a 0 input value.

Siehe auch

Raster boolean AND

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	INPUT	[raster] [list]	List of input raster layers
Referenz-Layer	REF_LAYER	[Raster]	The reference layer to create the output layer from (extent, CRS, pixel dimensions)
Treat NoData values as false	NODATA_AS_FALS	[Boolean] Standard: falsch	Treat NoData values in the input files as 0 when performing the operation
Ausgabelayer	OUTPUT	[Raster] Standard: [In temporäre Datei speichern]	Specification of the output raster containing the result. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value	NO_DATA	[numerisch: Double] Default: -9999.0	Value to use for NoData in the output layer
Output data type	DATA_TYPE	[Aufzählung] Default: 5	Output raster data type. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 4 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).
Creation options Optional	CREATION_OPTIO (for QGIS <= 3.42, this was CREATE_OPTIONS	[Text] Default: ,‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extent	EXTENT	[Text]	The spatial extent of the output raster layer
CRS authority identifier	CRS_AUTHID	[KBS]	The coordinate reference system of the output raster layer
Width in pixels	WIDTH_IN_PIXEL	[numerisch: Ganzzahl]	The number of columns in the output raster layer
Height in pixels	HEIGHT_IN_PIXE	[numerisch: Ganzzahl]	The number of rows in the output raster layer
Total pixel count	TOTAL_PIXEL_CO	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Fortsetzung auf der nächsten Seite

Tab. 24.351 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
NoData pixel count	NODATA_PIXEL_C	[numerisch: Ganzzahl]	The count of NoData pixels in the output raster layer
True pixel count	TRUE_PIXEL_COU	[numerisch: Ganzzahl]	The count of True pixels (value = 1) in the output raster layer
False pixel count	FALSE_PIXEL_CO	[numerisch: Ganzzahl]	The count of False pixels (value = 0) in the output raster layer
Ausgabelayer	OUTPUT	[Raster]	Output raster layer containing the result

Pythoncode

Algorithm ID: native:rasterbooleanor

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster calculator

Performs algebraic operations using raster layers.

The resulting layer will have its values computed according to an expression. The expression can contain numerical values, operators and references to any of the layers in the current project.

Siehe auch

Raster calculator (virtual), *Raster calculator*, *Rasterrechner*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	INPUT	[raster] [list]	List of input raster layers
Expression	EXPRESSION	[Ausdruck]	*Raster-based expression* that will be used to calculate the output raster layer.

Fortsetzung auf der nächsten Seite

Tab. 24.352 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output extent Optional	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. If the extent is not specified, the minimum extent that covers all the selected reference layers will be used. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Output cell size (leave empty to set automatically) Optional	`CELLSIZE`	[numerisch: Double]	Cell size of the output raster layer. If the cell size is not specified, the minimum cell size of the selected reference layer(s) will be used. The cell size will be the same for the X and Y axes.
Output CRS Optional	`CRS`	[KBS]	CRS of the output raster layer. If the output CRS is not specified, the CRS of the first reference layer will be used.
Calculated	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Calculated	`OUTPUT`	[Raster]	Output raster file with the calculated values.

Pythoncode

Algorithm ID: `native:rastercalc`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster calculator (virtual)

Performs algebraic operations using raster layers and generates in-memory result.

The resulting layer will have its values computed according to an expression. The expression can contain numerical values, operators and references to any of the layers in the current project.

A virtual raster layer is a raster layer defined by its URI and whose pixels are calculated on-the-fly. It's not a new file on disk; the virtual layer is still connected to the rasters used in the calculation meaning that deleting or moving these rasters would break it. A *Layer name* can be provided, otherwise the calculation expression is used as such. Removing the virtual layer from the project deletes it, and it can be made persistent in file using the layer *Export* ► *Save as…* contextual menu.

Siehe auch

Raster calculator, *Raster calculator*, *Rasterrechner*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[raster] [list]	List of input raster layers
Expression	`EXPRESSION`	[Ausdruck]	*Raster-based expression* that will be used to calculate the output raster layer.

Fortsetzung auf der nächsten Seite

Tab. 24.355 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output extent Optional	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. If the extent is not specified, the minimum extent that covers all the selected reference layers will be used. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Output cell size (leave empty to set automatically) Optional	`CELLSIZE`	[numerisch: Double]	Cell size of the output raster layer. If the cell size is not specified, the minimum cell size of the selected reference layer(s) will be used. The cell size will be the same for the X and Y axes.
Output CRS Optional	`CRS`	[KBS]	CRS of the output raster layer. If the output CRS is not specified, the CRS of the first reference layer will be used.
Output layer name Optional	`LAYER_NAME`	[Text]	The name to assign to the generated layer. If not set, the text of the calculation expression is used.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Calculated	`OUTPUT`	[Raster]	Output virtual raster layer with the calculated values.

Pythoncode

Algorithm ID: `native:virtualrastercalc`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster layer properties

Returns basic properties of the given raster layer, including the extent, size in pixels and dimensions of pixels (in map units), number of bands, and NoData value.

This algorithm is intended for use as a means of extracting these useful properties to use as the input values to other algorithms in a model - e.g. to allow to pass an existing raster's pixel sizes over to a GDAL raster algorithm.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer
Band number Optional	`BAND`	[raster band] Standard: Nicht gesetzt	Whether to also return properties of a specific band. If a band is specified, the noData value for the selected band is also returned.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Number of bands in raster	`BAND_COUNT`	[numerisch: Ganzzahl]	The number of bands in the raster
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The raster layer extent in the CRS
Band has a NoData value set	`HAS_NODATA_VAL`	[Boolean]	Indicates whether the raster layer has a value set for NoData pixels in the selected band
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of columns in the raster layer
Band NoData value	`NODATA_VALUE`	[numerisch: Double]	The value (if set) of the NoData pixels in the selected band
Pixel size (height) in map units	`PIXEL_HEIGHT`	[numerisch: Ganzzahl]	Vertical size in map units of the pixel
Pixel size (width) in map units	`PIXEL_WIDTH`	[numerisch: Ganzzahl]	Horizontal size in map units of the pixel
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of rows in the raster layer

Fortsetzung auf der nächsten Seite

Tab. 24.358 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Maximum x-coordinate	`X_MAX`	[numerisch: Double]	
Minimum x-coordinate	`X_MIN`	[numerisch: Double]	
Maximum y-coordinate	`Y_MAX`	[numerisch: Double]	
Minimum y-coordinate	`Y_MIN`	[numerisch: Double]	

Pythoncode

Algorithm ID: `native:rasterlayerproperties`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster layer statistics

Calculates basic statistics from the values in a given band of the raster layer. The output is loaded in the *Processing* ► *Results viewer* menu.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer
Band number	`BAND`	[raster band] Default: The first band of the input layer	If the raster is multiband, choose the band you want to get statistics for.
Statistics	`OUTPUT_HTML_FI`	[html] Standard: `[In temporäre Datei speichern]`	Specification of the output file: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Maximum value	`MAX`	[numerisch: Double]	
Mean value	`MEAN`	[numerisch: Double]	
Minimum value	`MIN`	[numerisch: Double]	
Statistics	`OUTPUT_HTML_FI`	[html]	The output file contains the following information: • Analyzed file: path of the raster layer • Minimum value: minimum value of the raster • Maximum value: maximum value of the raster • Range: difference between the maximum and minimum values • Sum: total sum of the values • Mean value: mean of the values • Standard deviation: standard deviation of the values • Sum of the squares: sum of the squared differences of each observation from the overall mean
Range	`RANGE`	[numerisch: Double]	
Standard deviation	`STD_DEV`	[numerisch: Double]	
Sum	`SUM`	[numerisch: Double]	
Sum of the squares	`SUM_OF_SQUARES`	[numerisch: Double]	

Pythoncode

Algorithm ID: `native:rasterlayerstatistics`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster layer unique values report

Returns the count and area of each unique value in a given raster layer. The calculation of the area is done in the area unit of the layer's CRS.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer
Band number	`BAND`	[raster band] Default: The first band of the input layer	If the raster is multiband, choose the band you want to get statistics for.
Unique values report	`OUTPUT_HTML_FI`	[Datei] Standard: `[In temporäre Datei speichern]`	Specification of the output file: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…
Unique values table	`OUTPUT_TABLE`	[vector: table] Standard: `[Ausgabe überspringen]`	Specification of the table for unique values: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
NoData pixel count	`NODATA_PIXEL_C`	[numerisch: Ganzzahl]	The number of NoData pixels in the output raster layer
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer

Fortsetzung auf der nächsten Seite

Tab. 24.362 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Unique values report	`OUTPUT_HTML_FI`	[html]	The output HTML file contains the following information: • Analyzed file: the path of the raster layer • Extent: xmin, ymin, xmax, ymax coordinates of the extent • Projection: projection of the layer • Width in pixels: number of columns and pixel width size • Height in pixels: number of rows and pixel width size • Total pixel count: count of all the pixels • NoData pixel count: count of pixels with NoData value
Unique values table	`OUTPUT_TABLE`	[vector: table]	A table with three columns: • *value*: pixel value • *count*: count of pixels with this value • *m2* or *deg2* or *ft2* or … : total area of pixels with this value. The column name depends on the area unit of the layer's CRS and the calculation is done in that unit.
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer

Pythoncode

Algorithm ID: `native:rasterlayeruniquevaluesreport`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster layer zonal statistics

Calculates statistics for a raster layer's values, categorized by zones defined in another raster layer.

Siehe auch

Zonal statistics

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Raster]	Input raster layer
Band number	`BAND`	[raster band] Default: The first band of the raster layer	If the raster is multiband choose the band for which you want to calculate the statistics.
Zones layer	`ZONES`	[Raster]	Raster layer defining zones. Zones are given by contiguous pixels having the same pixel value.
Zones band number	`ZONES_BAND`	[raster band] Default: The first band of the raster layer	If the raster is multiband, choose the band that defines the zones
Statistics	`OUTPUT_TABLE`	[vector: table] Standard: `[Create temporary layer]`	Specification of the output report. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Referenz-Layer Optional	`REF_LAYER`	[Aufzählung] Standard: 0	Raster layer used to calculate the centroids that will be used as reference when determining the zones in the output layer. One of: • 0 — Input layer: zones are determined by sampling the zone raster layer value at the centroid of each pixel from the source raster layer • 1 — Zones layer: the input raster layer will be sampled at the centroid of each pixel from the zones raster layer

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
CRS authority identifier	`CRS_AUTHID`	[Text]	The coordinate reference system of the output raster layer
Extent	`EXTENT`	[Text]	The spatial extent of the output raster layer
Height in pixels	`HEIGHT_IN_PIXE`	[numerisch: Ganzzahl]	The number of rows in the output raster layer
NoData pixel count	`NODATA_PIXEL_C`	[numerisch: Ganzzahl]	The number of NoData pixels in the output raster layer
Statistics	`OUTPUT_TABLE`	[vector: table]	The output layer contains the following information **for each zone**: • Area: the area in square raster units in the zone; • Sum: the total sum of the pixel values in the zone; • Count: the number of pixels in the zone; • Min: the minimum pixel value in the zone; • Max: the maximum pixel value in the zone; • Mean: the mean of the pixel values in the zone;
Total pixel count	`TOTAL_PIXEL_CO`	[numerisch: Ganzzahl]	The count of pixels in the output raster layer
Width in pixels	`WIDTH_IN_PIXEL`	[numerisch: Ganzzahl]	The number of columns in the output raster layer

Pythoncode

Algorithm ID: `native:rasterlayerzonalstats`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

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.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Raster layer from which to extract the minimum and maximum pixel values.
Band number	`BAND`	[raster band] Standard: 1	The band of the raster to analyze. If the raster is multiband, specify the band number (starting from 1).
Extract	`EXTREMA`	[Aufzählung] Default: 0 (Minimum and Maximum)	Choose which extrema to extract: • 0: Minimum and Maximum • 1: Minimum only • 2: Maximum only
Ausgabelayer	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specification of the output layer. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Vektor: Punkt]	Vector layer with point features at the locations of the minimum and/or maximum pixel values.

Pythoncode

Algorithm ID: `native:rasterminmax`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

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.

Siehe auch

Cell stack percent rank from value, *Cell stack percentile*, *Cell stack percentrank from raster layer*, *Cell statistics*, *Raster calculator*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT_RASTERS`	[raster] [list]	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`	[Aufzählung] Standard: 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.
Ranked	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster containing the rank values. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output extent Optional	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. If the extent is not specified, the minimum extent that covers all the selected reference layers will be used. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Output cell size Optional	`CELL_SIZE`	[numerisch: Double]	Cell size of the output raster layer. If the cell size is not specified, the minimum cell size of the selected reference layer(s) will be used. The cell size will be the same for the X and Y axes.
Output CRS Optional	`CRS`	[KBS]	CRS of the output raster layer. If the output CRS is not specified, the CRS of the first reference layer will be used.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ranked	`OUTPUT`	[Raster]	Output raster layer containing the result.

Pythoncode

Algorithm ID: `native:rasterrank`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster surface volume

Calculates the volume under a raster surface relative to a given base level. This is mainly useful for Digital Elevation Models (DEM).

Parameter

Bezeichnung	Name	Typ	Beschreibung
INPUT layer	`INPUT`	[Raster]	Input raster, representing a surface
Band number	`BAND`	[raster band] Default: The first band of the raster layer	If the raster is multiband, choose the band that shall define the surface.
Base level	`LEVEL`	[numerisch: Double] Default: 0.0	Define a base or reference value. This base is used in the volume calculation according to the `Method` parameter (see below).
Methode	`METHOD`	[Aufzählung] Standard: 0	Define the method for the volume calculation given by the difference between the raster pixel value and the `Base level`. Options: • 0 — Count Only Above Base Level: only pixels above the base level will add to the volume. • 1 — Count Only Below Base Level: only pixels below the base level will add to the volume. • 2 — Subtract Volumes Below Base level: pixels above the base level will add to the volume, pixels below the base level will subtract from the volume. • 3 — Add Volumes Below Base level: Add the volume regardless whether the pixel is above or below the base level. This is equivalent to sum the absolute values of the difference between the pixel value and the base level.
Surface volume report	`OUTPUT_HTML_FI`	[html] Standard: `[In temporäre Datei speichern]`	Specification of the output HTML report. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortsetzung auf der nächsten Seite

Tab. 24.371 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Surface volume table	`OUTPUT_TABLE`	[vector: table] Standard: `[Ausgabe überspringen]`	Specification of the output table. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Volume	`VOLUME`	[numerisch: Double]	The calculated volume
Area	`AREA`	[numerisch: Double]	The area in square map units
Pixel_count	`PIXEL_COUNT`	[numerisch: Ganzzahl]	The total number of pixels that have been analyzed
Surface volume report	`OUTPUT_HTML_FI`	[html]	The output report (containing volume, area and pixel count) in HTML format
Surface volume table	`OUTPUT_TABLE`	[vector: table]	The output table (containing volume, area and pixel count)

Pythoncode

Algorithm ID: `native:rastersurfacevolume`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Reclassify by layer

Reclassifies a raster band by assigning new class values based on the ranges specified in a vector table.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Raster layer	`INPUT_RASTER`	[Raster]	Raster layer to reclassify
Band number	`RASTER_BAND`	[raster band] Default: The first band of the raster layer	If the raster is multiband, choose the band you want to reclassify.
Layer containing class breaks	`INPUT_TABLE`	[Vektor: alle]	Vector layer containing the values to use for classification.
Minimum class value field	`MIN_FIELD`	[tablefield: numeric]	Field with the minimum value of the range for the class. Use `-inf` to capture the lowest possible value.
Maximum class value field	`MAX_FIELD`	[tablefield: numeric]	Field with the maximum value of the range for the class. Use `inf` to capture the highest possible value.
Output value field	`VALUE_FIELD`	[tablefield: numeric]	Field with the value that will be assigned to the pixels that fall in the class (between the corresponding min and max values). Use `nan` to set the value of the range to NoData.
Reclassified raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value	`NO_DATA`	[numerisch: Double] Default: -9999.0	Value to apply to NoData values.
Range boundaries	`RANGE_BOUNDARI`	[Aufzählung] Standard: 0	Defines comparison rules for the classification. Options: • 0 — min < value <= max • 1 — min <= value < max • 2 — min <= value <= max • 3 — min < value < max
Use NoData when no range matches value	`NODATA_FOR_MIS`	[Boolean] Standard: falsch	Applies the NoData value to band values that do not fall in any class. If False, the original value is kept.

Fortsetzung auf der nächsten Seite

Tab. 24.374 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the format of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 4 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Reclassified raster	`OUTPUT`	[Raster]	Output raster layer with reclassified band values

Pythoncode

Algorithm ID: `native:reclassifybylayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Reclassify by table

Reclassifies a raster band by assigning new class values based on the ranges specified in a fixed table.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Raster layer	`INPUT_RASTER`	[Raster]	Raster layer to reclassify
Band number	`RASTER_BAND`	[raster band] Standard: 1	Raster band for which you want to recalculate values.
Reclassification table	`TABLE`	[vector: table]	A 3-columns table to fill with the values to set the boundaries of each class (`Minimum` and `Maximum`) and the new `Value` to assign to the band values that fall in the class. The value `-inf` can be used as minimum, `inf` as maximum, and `nan` can be used to set the output value to NoData.
Reclassified raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output NoData value	`NO_DATA`	[numerisch: Double] Default: -9999.0	Value to apply to NoData values.
Range boundaries	`RANGE_BOUNDARI`	[Aufzählung] Standard: 0	Defines comparison rules for the classification. Options: • 0 — min < value <= max • 1 — min <= value < max • 2 — min <= value <= max • 3 — min < value < max
Use NoData when no range matches value	`NODATA_FOR_MIS`	[Boolean] Standard: falsch	Applies the NoData value to band values that do not fall in any class. If False, the original value is kept.

Fortsetzung auf der nächsten Seite

Tab. 24.377 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the format of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 4 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Reclassified raster	`OUTPUT`	[Raster]	Output raster layer with reclassified band values

Pythoncode

Algorithm ID: `native:reclassifybytable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Rescale raster

Rescales raster layer to a new value range, while preserving the shape (distribution) of the raster's histogram (pixel values). Input values are mapped using a linear interpolation from the source raster's minimum and maximum pixel values to the destination minimum and miximum pixel range.

By default the algorithm preserves the original NoData value, but there is an option to override it.

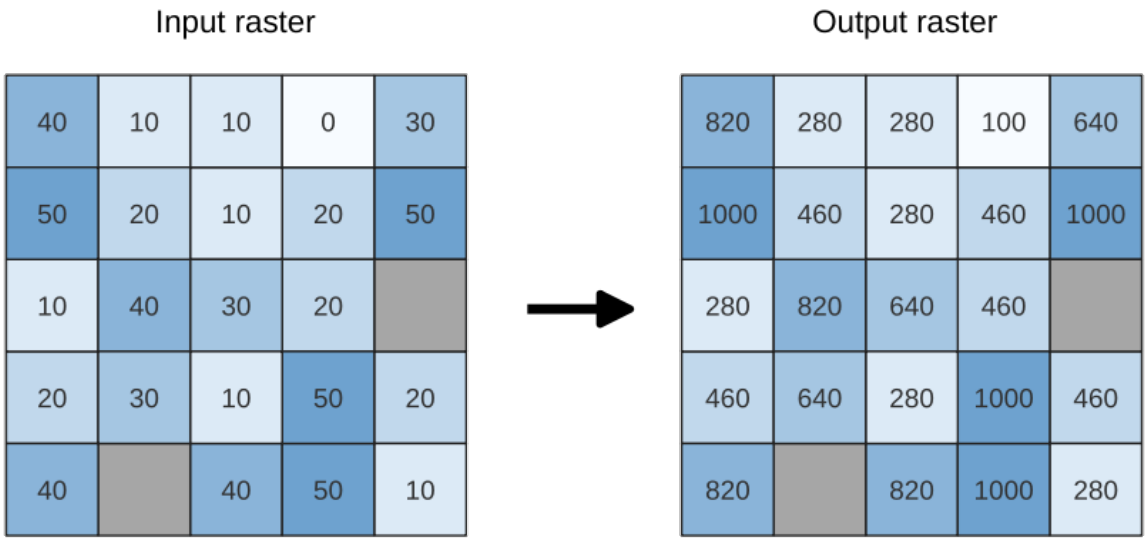


Abb. 24.57: Rescaling values of a raster layer from [0 - 50] to [100 - 1000]

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Raster	`INPUT`	[Raster]	Raster layer to use for rescaling
Band number	`BAND`	[raster band] Default: The first band of the input layer	If the raster is multiband, choose a band.
New minimum value	`MINIMUM`	[numerisch: Double] Default value: 0.0	Minimum pixel value to use in the rescaled layer
New maximum value	`MAXIMUM`	[numerisch: Double] Default value: 255.0	Maximum pixel value to use in the rescaled layer
New NoData value Optional	`NODATA`	[numerisch: Double] Default value: Not set	Value to assign to the NoData pixels. If unset, original NoData values are preserved.
Rescaled	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Rescaled	`OUTPUT`	[Raster]	Output raster layer with rescaled band values

Pythoncode

Algorithm ID: `native:rescaleraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Round raster

Rounds the cell values of a raster dataset according to the specified number of decimals.

Alternatively, a negative number of decimal places may be used to round values to powers of a base n. For example, with a Base value n of 10 and Decimal places of -1, the algorithm rounds cell values to multiples of 10, -2 rounds to multiples of 100, and so on. Arbitrary base values may be chosen, the algorithm applies the same multiplicative principle. Rounding cell values to multiples of a base n may be used to generalize raster layers.

The algorithm preserves the data type of the input raster. Therefore byte/integer rasters can only be rounded to multiples of a base n, otherwise a warning is raised and the raster gets copied as byte/integer raster.

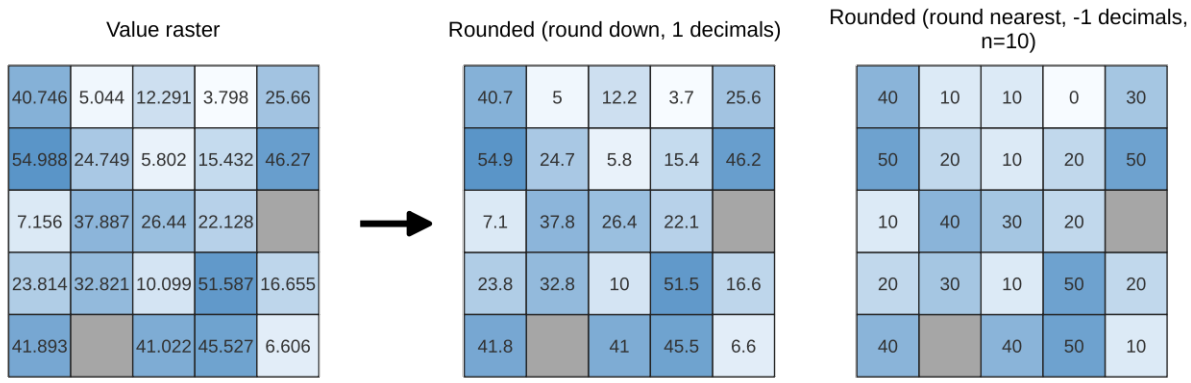


Abb. 24.58: Rounding values of a raster

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input raster	`INPUT`	[Raster]	The raster to process.
Band number	`BAND`	[raster band] Standard: 1	The band of the raster
Rounding direction	`ROUNDING_DIREC`	[list] Standard: 1	How to choose the target rounded value. Options are: • 0 — Round up • 1 — Round to nearest • 2 — Round down
Number of decimals places	`DECIMAL_PLACES`	[numerisch: Ganzzahl] Default: 2	Number of decimals places to round to. Use negative values to round cell values to a multiple of a base n
Output raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Base n for rounding to multiples of n	`BASE_N`	[numerisch: Ganzzahl] Default: 10	When the `DECIMAL_PLACES` parameter is negative, raster values are rounded to multiples of the base n value

Fortsetzung auf der nächsten Seite

Tab. 24.382 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster]	The output raster layer with values rounded for the selected band.

Pythoncode

Algorithm ID: `native:roundrastervalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Sample raster values

Extracts raster values at the point locations. If the raster layer is multiband, each band is sampled.

The attribute table of the resulting layer will have as many new columns as the raster layer band count.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: Punkt]	Point vector layer to use for sampling
Raster Layer	`RASTERCOPY`	[Raster]	Raster layer to sample at the given point locations.
Output column prefix	`COLUMN_PREFIX`	[Text] Default: ‚SAMPLE_‘	Prefix for the names of the added columns.
Sampled Optional	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output layer containing the sampled values. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Sampled	`OUTPUT`	[Vektor: Punkt]	The output layer containing the sampled values.

Pythoncode

Algorithm ID: `native:rastersampling`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Zonal histogram

Appends fields representing counts of each unique value from a raster layer contained within polygon features.

The output layer attribute table will have as many fields as the unique values of the raster layer that intersects the polygon(s).

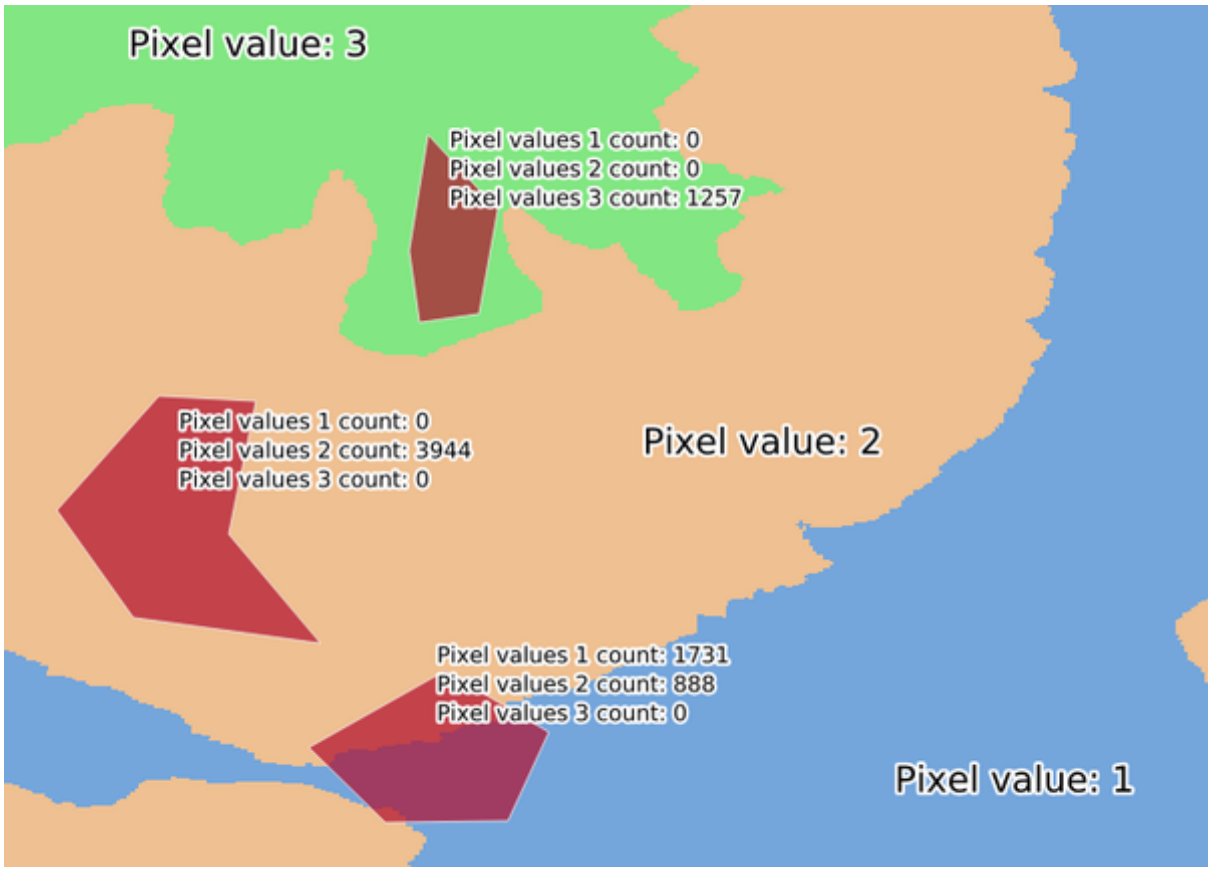


Abb. 24.59: Raster layer histogram example

Parameter

Bezeichnung	Name	Typ	Beschreibung
Raster layer	`INPUT_RASTER`	[Raster]	Input raster layer.
Band number	`RASTER_BAND`	[raster band] Default: The first band of the input layer	If the raster is multiband, choose a band.
Vector layer containing zones	`INPUT_VECTOR`	[Vektor: Polygon]	Vector polygon layer that defines the zones.
Output column prefix	`COLUMN_PREFIX` Optional	[Text] Default: ‚HISTO_‘	Prefix for the output columns names.

Fortsetzung auf der nächsten Seite

Tab. 24.385 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output zones	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector polygon layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output zones	`OUTPUT`	[Vektor: Polygon]	The output vector polygon layer.

Pythoncode

Algorithm ID: `native:zonalhistogram`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

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.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: Polygon]	Polygon layer defining the zones.
Raster Layer	`RASTER`	[Raster]	Raster layer to extract the minimum and maximum values from.
Raster Band	`BAND`	[raster band] Default: The first band of the raster layer	If the raster has multiple bands, select the band to process.
Zonal extrema	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specification of the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Zonal extrema	`OUTPUT`	[Vektor: Punkt]	Output layer containing the minimum and maximum points for each zone.

Pythoncode

Algorithm ID: `native:zonalminmaxpoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Zonal statistics

Calculates statistics of a raster layer for each feature of an overlapping polygon vector layer.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Vector polygon layer that contains the zones.
Raster layer	`INPUT_RASTER`	[Raster]	Input raster layer.
Raster band	`RASTER_BAND`	[raster band] Default: The first band of the input layer	If the raster is multiband, choose a band for the statistics.
Output column prefix	`COLUMN_PREFIX`	[Text] Default: ‚_‘	Prefix for the output columns names.
Statistics to calculate	`STATISTICS`	[Aufzählung] [Liste] Default: [0,1,2]	List of statistical operator for the output. Options: • 0 — Count • 1 — Sum • 2 — Mean • 3 — Median • 4 — St. dev. • 5 — Minimum • 6 — Maximum • 7 — Range • 8 — Minority • 9 — Majority • 10 — Variety • 11 — Variance
Zonal Statistics	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector polygon layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Zonal Statistics	`OUTPUT`	[Vektor: Polygon]	The zone vector layer with added statistics.

Pythoncode

Algorithm ID: `native:zonalstatisticsfb`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.19 Raster Creation

Create constant raster layer

Generates raster layer for given extent and cell size filled with the specified value.

Additionally an output data type can be specified. The algorithm will abort if a value has been entered that cannot be represented by the selected output raster data type.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Desired extent	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Ziel KBS	`TARGET_CRS`	[KBS] Default: Project CRS	CRS for the output raster layer

Fortsetzung auf der nächsten Seite

Tab. 24.389 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Standard: 1,0	Pixel size (X=Y) in map units.
Constant value	`NUMBER`	[numerisch: Double] Standard: 1,0	Constant pixel value for the output raster layer.
Constant	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output raster data type	`OUTPUT_TYPE` Default: 5	[Aufzählung]	Defines the data type of the output raster file. Options: • 0 — Byte • 1 — Integer16 • 2 — Unsigned Integer16 • 3 — Integer32 • 4 — Unsigned Integer32 • 5 — Float32 • 6 — Float64
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ,‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Constant	`OUTPUT`	[Raster]	Raster covering the desired extent with the specified pixel size and value.

Pythoncode

Algorithm ID: `native:createconstantrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create random raster layer (binomial distribution)

Generates a raster layer for given extent and cell size filled with binomially distributed random values.

By default, the values will be chosen given an N of 10 and a probability of 0.5. This can be overridden by using the advanced parameter for N and probability. The raster data type is set to Integer types (Integer16 by default). The binomial distribution random values are defined as positive integer numbers. A floating point raster will represent a cast of integer values to floating point.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Desired extent	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Ziel KBS	`TARGET_CRS`	[KBS] Default: Project CRS	CRS for the output raster layer

Fortsetzung auf der nächsten Seite

Tab. 24.392 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Standard: 1,0	Pixel size (X=Y) in map units.
Output raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output raster data type	`OUTPUT_TYPE` Standard: 0	[Aufzählung]	Defines the data type of the output raster file. Options: • 0 — Integer16 • 1 — Unsigned Integer16 • 2 — Integer32 • 3 — Unsigned Integer32 • 4 — Float32 • 5 — Float64
N	`N`	[numerisch: Ganzzahl] Default: 10	
Probability	`PROBABILITY`	[numerisch: Double] Default: 0.5	
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster]	Raster covering the desired extent with the cell size filled with random values

Pythoncode

Algorithm ID: `native:createrandombinomialrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create random raster layer (exponential distribution)

Generates a raster layer for given extent and cell size filled with exponentially distributed random values.

By default, the values will be chosen given a lambda of 1.0. This can be overridden by using the advanced parameter for lambda. The raster data type is set to Float32 by default as the exponential distribution random values are floating point numbers.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Desired extent	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Ziel KBS	`TARGET_CRS`	[KBS] Default: Project CRS	CRS for the output raster layer
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Standard: 1,0	Pixel size (X=Y) in map units.

Fortsetzung auf der nächsten Seite

Tab. 24.395 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output raster data type	`OUTPUT_TYPE` Standard: 0	[Aufzählung]	Defines the data type of the output raster file. Options: • 0 — Float32 • 1 — Float64
Lambda	`LAMBDA`	[numerisch: Double] Standard: 1,0	
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster]	Raster covering the desired extent with the cell size filled with random values

Pythoncode

Algorithm ID: `native:createrandomexponentialrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create random raster layer (gamma distribution)

Generates a raster layer for given extent and cell size filled with gamma distributed random values.

By default, the values will be chosen given an alpha and beta value of 1.0. This can be overridden by using the advanced parameter for alpha and beta. The raster data type is set to Float32 by default as the gamma distribution random values are floating point numbers.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Desired extent	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Ziel KBS	`TARGET_CRS`	[KBS] Default: Project CRS	CRS for the output raster layer
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Standard: 1,0	Pixel size (X=Y) in map units.
Output raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output raster data type	`OUTPUT_TYPE` Standard: 0	[Aufzählung]	Defines the data type of the output raster file. Options: • 0 — Float32 • 1 — Float64
Alpha	`ALPHA`	[numerisch: Double] Standard: 1,0	
Beta	`BETA`	[numerisch: Double] Standard: 1,0	
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster]	Raster covering the desired extent with the cell size filled with randomly distributed values

Pythoncode

Algorithm ID: `native:createrandomgammarasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create random raster layer (geometric distribution)

Generates a raster layer for given extent and cell size filled with geometrically distributed random values.

By default, the values will be chosen given a probability of 0.5. This can be overridden by using the advanced parameter for mean value. The raster data type is set to Integer types (Integer16 by default). The geometric distribution random values are defined as positive integer numbers. A floating point raster will represent a cast of integer values to floating point.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Desired extent	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Ziel KBS	`TARGET_CRS`	[KBS] Default: Project CRS	CRS for the output raster layer
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Standard: 1,0	Pixel size (X=Y) in map units.
Output raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output raster data type	`OUTPUT_TYPE` Standard: 0	[Aufzählung]	Defines the data type of the output raster file. Options: • 0 — Integer16 • 1 — Unsigned Integer16 • 2 — Integer32 • 3 — Unsigned Integer32 • 4 — Float32 • 5 — Float64
Probability	`PROBABILITY`	[numerisch: Double] Default: 0.5	
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster]	Raster covering the desired extent with the cell size filled with randomly distributed values

Pythoncode

Algorithm ID: `native:createrandomgeometricrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create random raster layer (negative binomial distribution)

Generates a raster layer for given extent and cell size filled with negative binomially distributed random values.

By default, the values will be chosen given a distribution parameter k of 10.0 and a probability of 0.5. This can be overridden by using the advanced parameters for k and probability. The raster data type is set to Integer types (Integer16 by default). The negative binomial distribution random values are defined as positive integer numbers. A floating point raster will represent a cast of integer values to floating point.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Desired extent	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Ziel KBS	`TARGET_CRS`	[KBS] Default: Project CRS	CRS for the output raster layer
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Standard: 1,0	Pixel size (X=Y) in map units.
Output raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output raster data type	`OUTPUT_TYPE` Standard: 0	[Aufzählung]	Defines the data type of the output raster file. Options: • 0 — Integer16 • 1 — Unsigned Integer16 • 2 — Integer32 • 3 — Unsigned Integer32 • 4 — Float32 • 5 — Float64
Distribution parameter k	`K_PARAMETER`	[numerisch: Ganzzahl] Default: 10	
Probability	`PROBABILITY`	[numerisch: Double] Default: 0.5	
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster]	Raster covering the desired extent with the cell size filled with randomly distributed values

Pythoncode

Algorithm ID: `native:createrandomnegativebinomialrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create random raster layer (normal distribution)

Generates a raster layer for given extent and cell size filled with normally distributed random values.

By default, the values will be chosen given a mean of 0.0 and a standard deviation of 1.0. This can be overridden by using the advanced parameters for mean and standard deviation value. The raster data type is set to Float32 by default as the normal distribution random values are floating point numbers.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Desired extent	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Ziel KBS	`TARGET_CRS`	[KBS] Default: Project CRS	CRS for the output raster layer
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Standard: 1,0	Pixel size (X=Y) in map units.
Output raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output raster data type	`OUTPUT_TYPE` Standard: 0	[Aufzählung]	Defines the data type of the output raster file. Options: • 0 — Float32 • 1 — Float64
Mean of normal distribution	`MEAN`	[numerisch: Double] Default: 0.0	
Standard deviation of normal distribution	`STDDEV`	[numerisch: Double] Standard: 1,0	
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster]	Raster covering the desired extent with the cell size filled with randomly distributed values

Pythoncode

Algorithm ID: `native:createrandomnormalrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create random raster layer (poisson distribution)

Generates a raster layer for given extent and cell size filled with poisson distributed random values.

By default, the values will be chosen given a mean of 1.0. This can be overridden by using the advanced parameter for mean value. The raster data type is set to Integer types (Integer16 by default). The poisson distribution random values are positive integer numbers. A floating point raster will represent a cast of integer values to floating point.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Desired extent	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Ziel KBS	`TARGET_CRS`	[KBS] Default: Project CRS	CRS for the output raster layer
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Standard: 1,0	Pixel size (X=Y) in map units.
Output raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output raster data type	`OUTPUT_TYPE` Standard: 0	[Aufzählung]	Defines the data type of the output raster file. Options: • 0 — Integer16 • 1 — Unsigned Integer16 • 2 — Integer32 • 3 — Unsigned Integer32 • 4 — Float32 • 5 — Float64
Mean	`MEAN`	[numerisch: Double] Standard: 1,0	
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster]	Raster covering the desired extent with the cell size filled with randomly distributed values

Pythoncode

Algorithm ID: `native:createrandompoissonrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create random raster layer (uniform distribution)

Generates a raster layer for given extent and cell size filled with random values.

By default, the values will range between the minimum and maximum value of the specified output raster type. This can be overridden by using the advanced parameters for lower and upper bound value. If the bounds have the same value or both are zero (default) the algorithm will create random values in the full value range of the chosen raster data type. Choosing bounds outside the acceptable range of the output raster type will abort the algorithm.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Desired extent	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Ziel KBS	`TARGET_CRS`	[KBS] Default: Project CRS	CRS for the output raster layer
Pixel size	`PIXEL_SIZE`	[numerisch: Double] Standard: 1,0	Pixel size (X=Y) in map units.
Output raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output raster data type	`OUTPUT_TYPE` Default: 5	[Aufzählung]	Defines the data type of the output raster file. Options: • 0 — Byte • 1 — Integer16 • 2 — Unsigned Integer16 • 3 — Integer32 • 4 — Unsigned Integer32 • 5 — Float32 • 6 — Float64
Lower bound for random number range	`LOWER_BOUND`	[numerisch: Double] Default: 0.0	
Upper bound for random number range	`UPPER_BOUND`	[numerisch: Double] Default: 0.0	
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster]	Raster covering the desired extent with the cell size filled with randomly distributed values

Pythoncode

Algorithm ID: `native:createrandomuniformrasterlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.20 Raster terrain analysis

Aspect

Calculates the aspect of the Digital Terrain Model in input. The final aspect raster layer contains values from 0 to 360 that express the slope direction, starting from north (0°) and continuing clockwise.

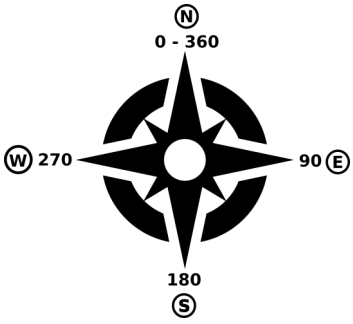


Abb. 24.60: Aspect values

The following picture shows the aspect layer reclassified with a color ramp:

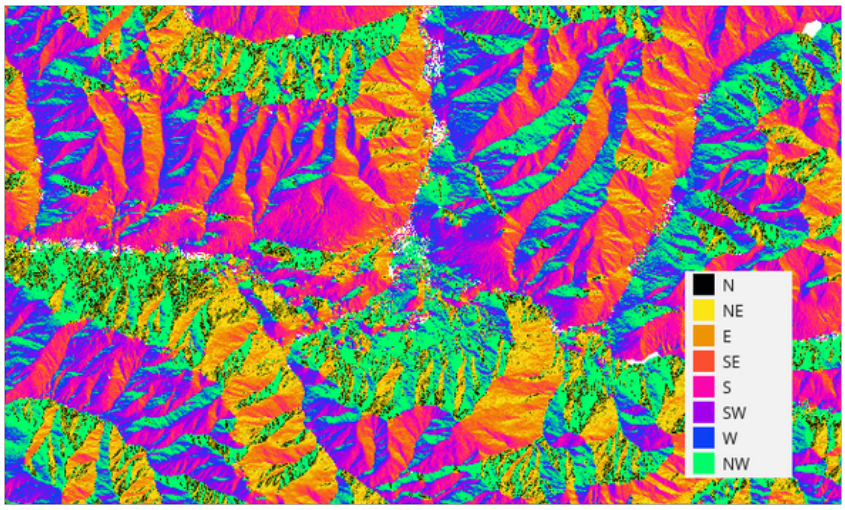


Abb. 24.61: Aspect layer reclassified

Parameter

Bezeichnung	Name	Typ	Beschreibung
Elevation layer	`INPUT`	[Raster]	Digital Terrain Model raster layer
Z factor	`Z_FACTOR`	[numerisch: Double] Standard: 1,0	Vertical exaggeration. This parameter is useful when the Z units differ from the X and Y units, for example feet and meters. You can use this parameter to adjust for this. The default is 1 (no exaggeration).

Fortsetzung auf der nächsten Seite

Tab. 24.416 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Aspect	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output aspect raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Aspect	`OUTPUT`	[Raster]	The output aspect raster layer

Pythoncode

Algorithm ID: `native:aspect`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

DTM filter (slope-based)

Can be used to filter a digital elevation model in order to classify its cells into ground and object (non-ground) cells.

The tool uses concepts as described by Vosselman (2000) and is based on the assumption that a large height difference between two nearby cells is unlikely to be caused by a steep slope in the terrain. The probability that the higher cell might be non-ground increases when the distance between the two cells decreases. Therefore the filter defines a maximum height difference (`dz_max`) between two cells as a function of the distance (`d`) between the cells (`dz_max( d ) = d`). A cell is classified as terrain if there is no cell within the kernel radius to which the height difference is larger than the allowed maximum height difference at the distance between these two cells.

The approximate terrain slope (`s`) parameter is used to modify the filter function to match the overall slope in the study area (`dz_max( d ) = d * s`). A 5 % confidence interval (`ci = 1.65 * sqrt( 2 * stddev )`) may be used to modify the filter function even further by either relaxing (`dz_max( d ) = d * s + ci`) or amplifying (`dz_max( d ) = d * s - ci`) the filter criterium.

References: Vosselman, G. (2000): Slope based filtering of laser altimetry data. IAPRS, Vol. XXXIII, Part B3, Amsterdam, The Netherlands, 935-942

Siehe auch

This tool is a port of the SAGA DTM Filter (slope-based)

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Digital Terrain Model raster layer
Band number	`BAND`	[raster band]	The band of the DEM to consider
Kernel radius (pixels)	`RADIUS`	[numerisch: Ganzzahl] Default: 5	The radius of the filter kernel (in pixels). Must be large enough to reach ground cells next to non-ground objects.
Terrain slope (%, pixel size/vertical units)	`TERRAIN_SLOPE`	[numerisch: Double] Default: 30.0	The approximate terrain slope in `%`. The terrain slope must be adjusted to account for the ratio of height units vs raster pixel dimensions. Used to relax the filter criterium in steeper terrain.
Filter modification	`FILTER_MODIFIC`	[list] Standard: 0	Choose whether to apply the filter kernel without modification or to use a confidence interval to relax or amplify the height criterium. • 0 - None • 1 - Relax filter • 2 - Amplify
Standard deviation	`STANDARD_DEVIA`	[numerisch: Double] Default: 0.1	The standard deviation used to calculate a 5% confidence interval applied to the height threshold.
Output layer (ground) Optional	`OUTPUT_GROUND`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the filtered DEM containing only cells classified as ground. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…
Output layer (non-ground objects) Optional	`OUTPUT_NONGROU`	[Raster] Standard: `[Ausgabe überspringen]`	Specify the non-ground objects removed by the filter. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ,‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output layer (ground)	`OUTPUT_GROUND`	[Raster]	The filtered DEM containing only cells classified as ground.
Output layer (non-ground objects)	`OUTPUT_NONGROU`	[Raster]	The non-ground objects removed by the filter.

Pythoncode

Algorithm ID: `native:dtmslopebasedfilter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

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.

Siehe auch

This tool is a port of the SAGA Fill Sinks (Wang & Liu) tool.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Digital Terrain Model raster layer
Band number	`BAND`	[raster band] Default: 1	The band of the DEM to consider
Minimum slope (degrees)	`MIN_SLOPE`	[numerisch: Double] 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).

Fortsetzung auf der nächsten Seite

Tab. 24.421 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output layer (filled DEM) Optional	OUTPUT_FILLED_	[Raster] Standard: [In temporäre Datei speichern]	Specify the ouput raster corresponding to the depression-free DEM. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…
Output layer (flow directions) Optional	OUTPUT_FLOW_DI	[Raster] Standard: [Ausgabe überspringen]	Specify the output raster with computed flow directions. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…
Output layer (watershed basins) Optional	OUTPUT_WATERSH	[Raster] Standard: [Ausgabe überspringen]	Specify the output raster corresponding to the delineated watershed basins. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	CREATION_OPTIO	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output layer (filled DEM)	OUTPUT_FILLED_	[Raster]	Ouput raster corresponding to the depression-free digital elevation model.
Output layer (flow directions)	OUTPUT_FLOW_DI	[Raster]	Output raster with computed flow directions; 0=N, 1=NE, 2=E, … 7=NW.
Output layer (watershed basins)	OUTPUT_WATERSH	[Raster]	Output raster corresponding to the delineated watershed basins.

Pythoncode

Algorithm ID: `native:fillsinkswangliu`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Schummerung

Calculates the hillshade raster layer given an input Digital Terrain Model.

The shading of the layer is calculated according to the sun position: you have the options to change both the horizontal angle (azimuth) and the vertical angle (sun elevation) of the sun.

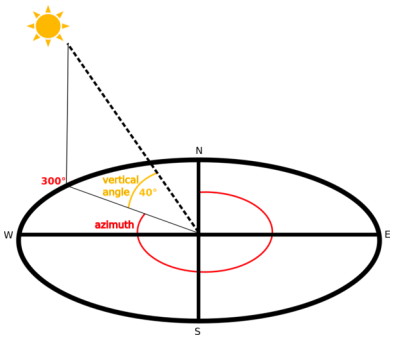


Abb. 24.62: Azimuth and vertical angle

The hillshade layer contains values from 0 (complete shadow) to 255 (complete sun). Hillshade is used usually to better understand the relief of the area.

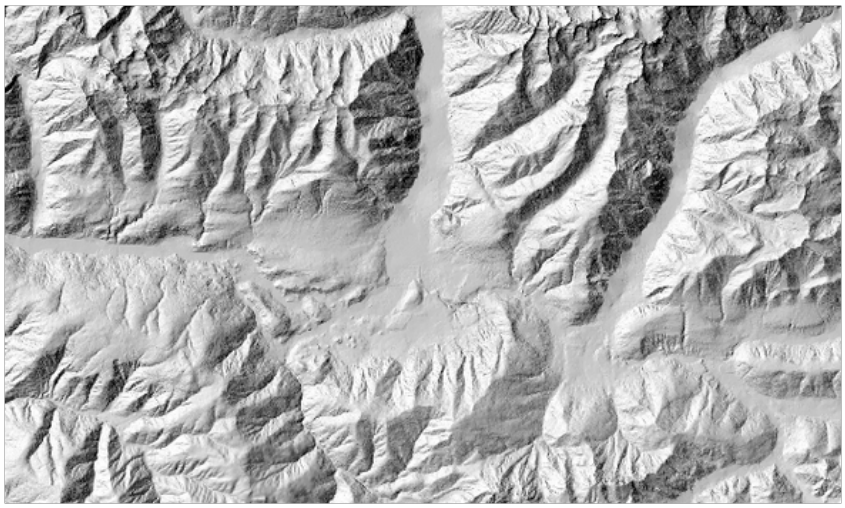


Abb. 24.63: Hillshade layer with azimuth 300 and vertical angle 45

Particularly interesting is to give the hillshade layer a transparency value and overlap it with the elevation raster:

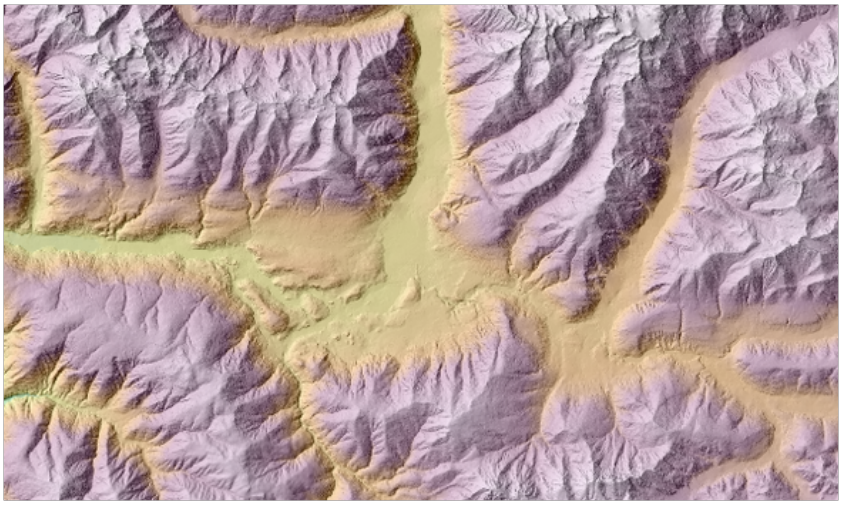


Abb. 24.64: Overlapping the hillshade with the elevation layer

Parameter

Bezeichnung	Name	Typ	Beschreibung
Elevation layer	`INPUT`	[Raster]	Digital Terrain Model raster layer
Z factor	`Z_FACTOR`	[numerisch: Double] Standard: 1,0	Vertical exaggeration. This parameter is useful when the Z units differ from the X and Y units, for example feet and meters. You can use this parameter to adjust for this. Increasing the value of this parameter will exaggerate the final result (making it look more „hilly“). The default is 1 (no exaggeration).
Azimuth (horizontal angle)	`AZIMUTH`	[numerisch: Double] Default: 300.0	Set the horizontal angle (in degrees) of the sun (clockwise direction). Range: 0 to 360. 0 is north.
Vertical angle	`V_ANGLE`	[numerisch: Double] Default: 40.0	Set the vertical angle (in degrees) of the sun, that is the height of the sun. Values can go from 0 (minimum elevation) to 90 (maximum elevation).
Hillshade	`OUTPUT`	[Raster] Default: `Save to temporary file`	Specify the output hillshade raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Hillshade	`OUTPUT`	[Raster]	The output hillshade raster layer

Pythoncode

Algorithm ID: `native:hillshade`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Hypsometric curves

Calculates hypsometric curves for an input Digital Elevation Model. Curves are produced as CSV files in an output folder specified by the user.

A hypsometric curve is a cumulative histogram of elevation values in a geographical area.

You can use hypsometric curves to detect differences in the landscape due to the geomorphology of the territory.

Parameter

Bezeichnung	Name	Typ	Beschreibung
DEM to analyze	`INPUT_DEM`	[Raster]	Digital Terrain Model raster layer to use for calculating altitudes
Boundary layer	`BOUNDARY_LAYER`	[Vektor: Polygon]	Polygon vector layer with boundaries of areas used to calculate hypsometric curves
Step	`STEP`	[numerisch: Double] Default: 100.0	Vertical distance between curves
Use % of area instead of absolute value	`USE_PERCENTAGE`	[Boolean] Standard: falsch	Write area percentage to "Area" field of the CSV file instead of the absolute area
Hypsometric curves	`OUTPUT_DIRECTO`	[Verzeichnis]	Specify the output folder for the hypsometric curves. *One of*: • In einem temporären Verzeichnis speichern • In Verzeichnis speichern

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Hypsometric curves	`OUTPUT_DIRECTO`	[Verzeichnis]	Directory containing the files with the hypsometric curves. For each feature from the input vector layer, a CSV file with area and altitude values will be created. The file names start with `histogram_`, followed by layer name and feature ID.

	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

Pythoncode

Algorithm ID: `qgis:hypsometriccurves`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Relief

Creates a shaded relief layer from digital elevation data. You can specify the relief color manually, or you can let the algorithm choose automatically all the relief classes.

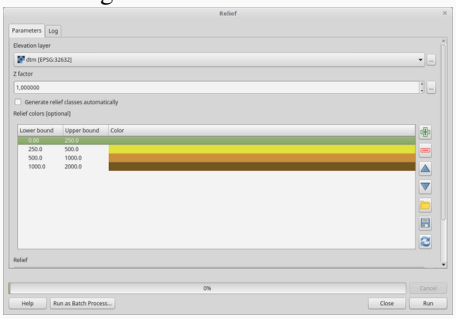
Abb. 24.65: Relief layer

Parameter

Bezeichnung	Name	Typ	Beschreibung
Elevation layer	`INPUT`	[Raster]	Digital Terrain Model raster layer
Z factor	`Z_FACTOR`	[numerisch: Double] Standard: 1,0	Vertical exaggeration. This parameter is useful when the Z units differ from the X and Y units, for example feet and meters. You can use this parameter to adjust for this. Increasing the value of this parameter will exaggerate the final result (making it look more „hilly“). The default is 1 (no exaggeration).
Generate relief classes automatically	`AUTO_COLORS`	[Boolean] Standard: falsch	If you check this option the algorithm will create all the relief color classes automatically

Fortsetzung auf der nächsten Seite

Tab. 24.428 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Relief colors Optional	`COLORS`	[table widget]	Use the table widget if you want to choose the relief colors manually. You can add as many color classes as you want: for each class you can choose the lower and upper bound and finally by clicking on the color row you can choose the color thanks to the color widget.  Abb. 24.66: Manually setting of relief color classes The buttons in the right side panel give you the chance to: add or remove color classes, change the order of the color classes already defined, open an existing file with color classes and save the current classes as file.
Relief	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output relief raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…
Frequency distribution Optional	`FREQUENCY_DIST`	[vector: table] Standard: `[Ausgabe überspringen]`	Specify the CSV table for the output frequency distribution. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Relief	`OUTPUT`	[Raster]	The output relief raster layer
Frequency distribution	`OUTPUT`	[vector: table]	The output frequency distribution

Pythoncode

Algorithm ID: `qgis:relief`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Ruggedness index

Calculates the quantitative measurement of terrain heterogeneity described by Riley et al. (1999). It is calculated for every location, by summarizing the change in elevation within the 3x3 pixel grid.

Each pixel contains the difference in elevation from a center cell and the 8 cells surrounding it.

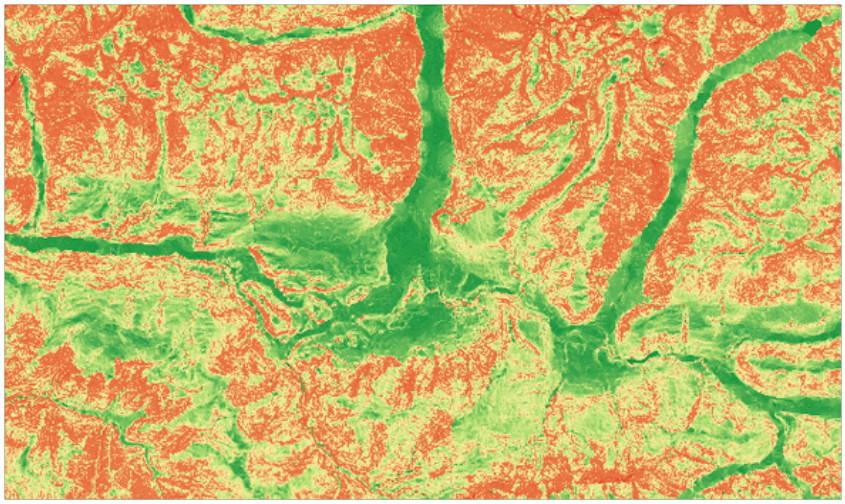


Abb. 24.67: Ruggedness layer from low (red) to high values (green)

Parameter

Bezeichnung	Name	Typ	Beschreibung
Elevation layer	`INPUT`	[Raster]	Digital Terrain Model raster layer
Z factor	`Z_FACTOR`	[numerisch: Double] Standard: 1,0	Vertical exaggeration. This parameter is useful when the Z units differ from the X and Y units, for example feet and meters. You can use this parameter to adjust for this. Increasing the value of this parameter will exaggerate the final result (making it look more rugged). The default is 1 (no exaggeration).

Fortsetzung auf der nächsten Seite

Tab. 24.430 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Ruggedness	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output ruggedness raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ruggedness	`OUTPUT`	[Raster]	The output ruggedness raster layer

Pythoncode

Algorithm ID: `native:ruggednessindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Slope

Calculates the slope from an input raster layer. The slope is the angle of inclination of the terrain and is expressed in **degrees**.

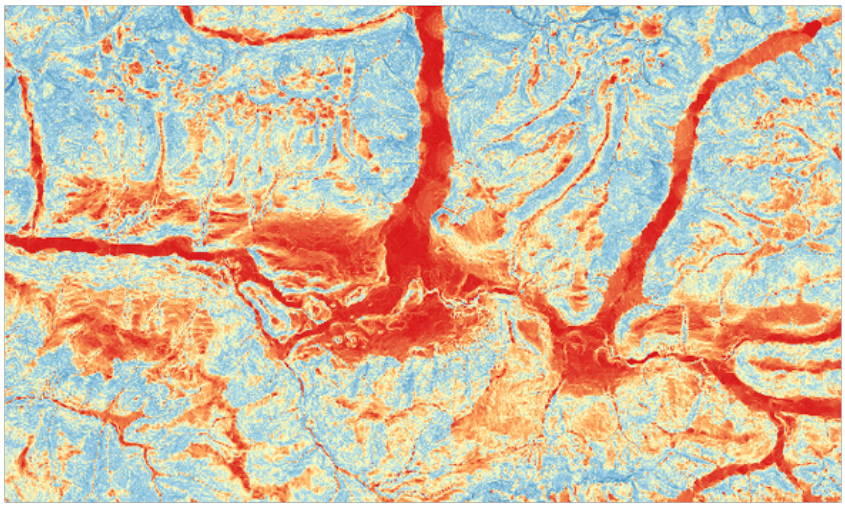


Abb. 24.68: Flat areas in red, steep areas in blue

Parameter

Bezeichnung	Name	Typ	Beschreibung
Elevation layer	`INPUT`	[Raster]	Digital Terrain Model raster layer
Z factor	`Z_FACTOR`	[numerisch: Double] Standard: 1,0	Vertical exaggeration. This parameter is useful when the Z units differ from the X and Y units, for example feet and meters. You can use this parameter to adjust for this. Increasing the value of this parameter will exaggerate the final result (making it steeper). The default is 1 (no exaggeration).
Slope	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output slope raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Slope	`OUTPUT`	[Raster]	The output slope raster layer

Pythoncode

Algorithm ID: `native:slope`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.21 Raster tools

Align raster

Aligns raster by resampling it to the same cell size and reprojecting to the same CRS as a reference raster.

Warnung

This algorithm is ONLY available in the *Model Designer* context. For other contexts, use instead *Align rasters*.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer to align
Resampling method	`RESAMPLING_MET`	[Aufzählung] Standard: 0	Method to use for input layer resampling. `Available options` are: • 0 — Nächster Nachbar • 1 — Bilinear (2x2 Kern) • 2 — Kubisch (4x4 Kern) • 3 — Kubisch B-Spline (4x4 Kern) • 4 — Lanczos (6x6 Kern) • 5 — Durchschnitt • 6 — Modus • 7 — Maximum • 8 — Minimum • 9 — Median • 10 — erstes Quartil (Q1) • 11 — drittes Quartil (Q3)
Rescale values according to the cell size	`RESCALE`	[Boolean] Default: No	
Referenz-Layer	`REFERENCE_LAYE`	[Raster]	A raster layer that will be used to fetch extent, cell size and CRS that will be applied to input layers.
Override reference CRS Optional	`CRS`	[KBS]	CRS to be used instead of the reference layer's
Override reference cell size X Optional	`CELL_SIZE_X`	[numerisch: Double]	Cell size in X direction to be used instead of the reference layer's
Override reference cell size Y Optional	`CELL_SIZE_Y`	[numerisch: Double]	Cell size in Y direction to be used instead of the reference layer's
Override reference grid offset X Optional	`GRID_OFFSET_X`	[numerisch: Double]	Offset in X direction to apply to cells grid
Override reference grid offset Y Optional	`GRID_OFFSET_Y`	[numerisch: Double]	Offset in Y direction to apply to cells grid

Fortsetzung auf der nächsten Seite

Tab. 24.434 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Clip to extent Optional	`EXTENT`	[Ausdehnung]	Specify the extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Aligned raster	`OUTPUT`	[Raster]	output raster with pixels resampled

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Aligned raster	`OUTPUT`	[Raster]	output raster with pixels resampled

Pythoncode

Algorithm ID: `native:alignsingleraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Align rasters

Aligns rasters by resampling them to the same cell size and reprojecting to the same CRS as a reference raster.

Warnung

This algorithm is NOT available in the *Model Designer* context. Use instead *Align raster*.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[raster] [list]	List of input raster layers with resampling options associated (filled as a `QgsProcessingParameterAlignRasterLayers` item — done in GUI by pressing *Configure Raster…* button for each selected layer): **Input layer [string] (`inputFile`)** Full path of the input layer to align **Output file [string] (`outputFile`)** Full path of the corresponding aligned output layer **Resampling method [enumeration] (`resampleMethod`)** Method to use for input layer resampling. `Available options` are: • 0 — Nächster Nachbar • 1 — Bilinear (2x2 Kern) • 2 — Kubisch (4x4 Kern) • 3 — Kubisch B-Spline (4x4 Kern) • 4 — Lanczos (6x6 Kern) • 5 — Durchschnitt • 6 — Modus • 7 — Maximum • 8 — Minimum • 9 — Median • 10 — erstes Quartil (Q1) • 11 — drittes Quartil (Q3) **Rescale values according to the cell size** [boolean] (`rescale`)
Referenz-Layer	`REFERENCE_LAYE`	[Raster]	A raster layer that will be used to fetch extent, cell size and CRS that will be applied to input layers.
Override reference CRS Optional	`CRS`	[KBS]	CRS to be used instead of the reference layer's
Override reference cell size X Optional	`CELL_SIZE_X`	[numerisch: Double]	Cell size in X direction to be used instead of the reference layer's
Override reference cell size Y Optional	`CELL_SIZE_Y`	[numerisch: Double]	Cell size in Y direction to be used instead of the reference layer's

Fortsetzung auf der nächsten Seite

Tab. 24.436 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Override reference grid offset X Optional	`GRID_OFFSET_X`	[numerisch: Double]	Offset in X direction to apply to cells grid
Override reference grid offset Y Optional	`GRID_OFFSET_Y`	[numerisch: Double]	Offset in Y direction to apply to cells grid
Clip to extent Optional	`EXTENT`	[Ausdehnung]	Specify the extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Aligned rasters	`OUTPUT_LAYERS`	[raster] [list]	output rasters with pixels resampled

Pythoncode

Algorithm ID: `native:alignrasters`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Convert map to raster

Creates a raster image of map canvas content.

A *map theme* can be selected to render a predetermined set of layers with a defined style for each layer.

Alternatively, a single layer can be selected if no map theme is set.

If neither map theme nor layer is set, the current map content will be rendered. The minimum extent entered will internally be extended to be a multiple of the tile size.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Minimum extent to render (xmin, xmax, ymin, ymax)	`EXTENT`	[Ausdehnung]	Specify the extent of the output raster layer. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Tile size	`TILE_SIZE`	[numerisch: Ganzzahl] Default: 1024	Size of the tile of the output raster layer. Minimum value: 64.
Map units per pixel	`MAP_UNITS_PER_`	[numerisch: Double] Default: 100.0	Pixel size (in map units). Minimum value: 0.0
Make background transparent	`MAKE_BACKGROUN`	[Boolean] Standard: falsch	Allows exporting the map with a transparent background. Outputs an RGBA (instead of RGB) image if set to `True`.
Map theme to render Optional	`MAP_THEME`	[Aufzählung]	Use an existing *map theme* for the rendering.
Single layer to render Optional	`LAYER`	[Aufzählung]	Choose a single layer for the rendering

Fortsetzung auf der nächsten Seite

Tab. 24.438 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[Raster]	Output raster layer

Pythoncode

Algorithm ID: `native:rasterize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Fill NoData cells

Resets the NoData values in the input raster to a chosen value, resulting in raster dataset with no NoData pixels.

The algorithm respects the input raster data type, e.g. a floating point fill value will be truncated when applied to an integer raster.

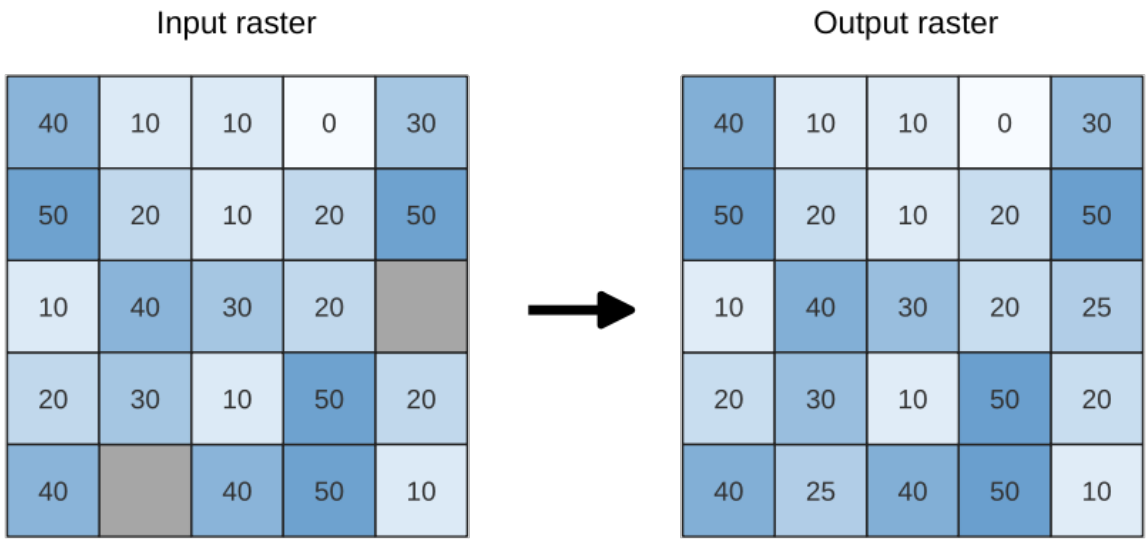


Abb. 24.69: Filling NoData values (in grey) of a raster

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input raster	`INPUT`	[Raster]	The raster to process.
Band number	`BAND`	[raster band] Standard: 1	The band of the raster
Fill value	`FILL_VALUE`	[numerisch: Double] Standard: 1,0	Set the value to use for the NoData pixels
Output raster	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `CREATE_OPTIONS`	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output raster	`OUTPUT`	[Raster]	The output raster layer with filled data cells.

Pythoncode

Algorithm ID: `native:fillnodata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Generate XYZ tiles (Directory)

Generates raster “XYZ” tiles using the current QGIS project as individual images to a directory structure.

Optionally, a Leaflet HTML output file using the generated tiles as a map layer could be created.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Extent (xmin, xmax, ymin, ymax)	`EXTENT`	[Ausdehnung]	Specify the extent of the tiles. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Minimum zoom	`ZOOM_MIN`	[numerisch: Ganzzahl] Default: 12	Minimum 0, maximum 25.
Maximum zoom	`ZOOM_MAX`	[numerisch: Ganzzahl] Default: 12	Minimum 0, maximum 25.
DPI	`DPI`	[numerisch: Ganzzahl] Default: 96	Minimum 48, maximum 600.
Background color Optional	`BACKGROUND_COL`	[color] Default: QColor(0, 0, 0, 0)	Choose the background color for the tiles
Enable antialiasing	`ANTIALIAS`	[Boolean] Standard: True	Determines if antialiasing should be enabled
Tile format	`TILE_FORMAT`	[Aufzählung] Standard: 0	One of: • 0 — PNG • 1 — JPG
Quality (JPG only) Optional	`QUALITY`	[numerisch: Ganzzahl] Default: 75	Minimum 1, maximum 100.

Fortsetzung auf der nächsten Seite

Tab. 24.443 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Metatile size Optional	`METATILESIZE`	[numerisch: Ganzzahl] Default: 4	Specify a custom metatile size when generating XYZ tiles. Larger values may speed up the rendering of tiles and provide better labelling (fewer gaps without labels) at the expense of using more memory. Minimum 1, maximum 20.
Tile width Optional	`TILE_WIDTH`	[numerisch: Ganzzahl] Default: 256	Minimum 1, maximum 4096.
Tile height Optional	`TILE_HEIGHT`	[numerisch: Ganzzahl] Default: 256	Minimum 1, maximum 4096.
Use inverted tile Y axis (TMS conventions)	`TMS_CONVENTION`	[Boolean] Standard: falsch	
Ausgabeverzeichnis	`OUTPUT_DIRECTO`	[Verzeichnis] Standard: `[In temporärem Verzeichnis speichern]`	Specification of the output directory (for the tiles). *One of*: • Ausgabe übersrpingen • In einem temporären Verzeichnis speichern • In Verzeichnis speichern
Output html (Leaflet) Optional	`OUTPUT_HTML`	[html] Standard: `[In temporäre Datei speichern]`	Specification of the output HTML file. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Leaflet HTML output title Optional	`HTML_TITLE`	[Text] Standard: Nicht gesetzt	HTML <title>-tag used for the Leaflet HTML output file.
Leaflet HTML output attribution Optional	`HTML_ATTRIBUTI`	[Text] Standard: Nicht gesetzt	Custom map attribution used for the Leaflet HTML output file. HTML links are possible.
Include OpenStreetMap basemap in Leaflet HTML output	`HTML_OSM`	[Boolean] Standard: falsch	An OpenStreetMap basemap layer (source: https://tile.openstreetmap.org) is included in the Leaflet HTML output file. Proper map attribution is added automatically.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabeverzeichnis	`OUTPUT_DIRECTO`	[Verzeichnis]	Output directory (for the tiles)
Output html (Leaflet)	`OUTPUT_HTML`	[html]	The output HTML (Leaflet) file

Pythoncode

Algorithm ID: `native:tilesxyzdirectory`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Generate XYZ tiles (MBTiles)

Generates raster “XYZ” tiles using the current QGIS project as a single file in the “MBTiles” format.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Extent (xmin, xmax, ymin, ymax)	`EXTENT`	[Ausdehnung]	Specify the extent of the tiles. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Fortsetzung auf der nächsten Seite

Tab. 24.446 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Minimum zoom	`ZOOM_MIN`	[numerisch: Ganzzahl] Default: 12	Minimum 0, maximum 25.
Maximum zoom	`ZOOM_MAX`	[numerisch: Ganzzahl] Default: 12	Minimum 0, maximum 25.
DPI	`DPI`	[numerisch: Ganzzahl] Default: 96	Minimum 48, maximum 600.
Background color Optional	`BACKGROUND_COL`	[color] Default: QColor(0, 0, 0, 0)	Choose the background color for the tiles
Enable antialiasing	`ANTIALIAS`	[Boolean] Standard: True	Determines if antialiasing should be enabled
Tile format	`TILE_FORMAT`	[Aufzählung] Standard: 0	One of: • 0 — PNG • 1 — JPG
Quality (JPG only) Optional	`QUALITY`	[numerisch: Ganzzahl] Default: 75	Minimum 1, maximum 100.
Metatile size Optional	`METATILESIZE`	[numerisch: Ganzzahl] Default: 4	Specify a custom metatile size when generating XYZ tiles. Larger values may speed up the rendering of tiles and provide better labelling (fewer gaps without labels) at the expense of using more memory. Minimum 1, maximum 20.
Output file (for MBTiles)	`OUTPUT_FILE`	[Datei] Standard: `[In temporäre Datei speichern]`	Specification of the output file. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Output file (for MBTiles)	`OUTPUT_FILE`	[Datei]	The output file.

Pythoncode

Algorithm ID: `native:tilesxyzmbtiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.22 Vektoranalyse

Grundstatistik für Felder

Generates basic statistics for a field of the attribute table of a vector layer. Numeric, date, time and string fields are supported. The statistics returned will depend on the field type.

Statistics can be generated as a table or an HTML file and are available in the *Processing* ► *Results viewer*.

Default menu: *Vector* ► *Analysis Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input vector	`INPUT_LAYER`	[Vektor: alle]	Vector layer to calculate the statistics on
Field to calculate statistics on	`FIELD_NAME`	[tablefield: any]	Any supported table field to calculate the statistics
Statistics Optional	`OUTPUT`	[vector: table] Standard: `[Create temporary layer]`	Specify the output table for the generated statistics. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Statistics report Optional	`OUTPUT_HTML_FI`	[html] Standard: `[In temporäre Datei speichern]`	Specification of the file for the calculated statistics. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Statistics	`OUTPUT`	[vector: table]	Table containing the calculated statistics
Statistics report	`OUTPUT_HTML_FI`	[html]	HTML file with the calculated statistics
Count	`COUNT`	[numerisch: Ganzzahl]	
Number of unique values	`UNIQUE`	[numerisch: Ganzzahl]	
Number of empty (null) values	`EMPTY`	[numerisch: Ganzzahl]	
Number of non-empty values	`FILLED`	[numerisch: Ganzzahl]	
Minimum value	`MIN`	[wie Eingabe]	
Maximum value	`MAX`	[wie Eingabe]	
Minimum length	`MIN_LENGTH`	[numerisch: Ganzzahl]	

Fortsetzung auf der nächsten Seite

Tab. 24.449 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Maximum length	`MAX_LENGTH`	[numerisch: Ganzzahl]	
Mean length	`MEAN_LENGTH`	[numerisch: Double]	
Coefficient of Variation	`CV`	[numerisch: Double]	
Sum	`SUM`	[numerisch: Double]	
Mean value	`MEAN`	[numerisch: Double]	
Standard deviation	`STD_DEV`	[numerisch: Double]	
Range	`RANGE`	[numerisch: Double]	
Median	`MEDIAN`	[numerisch: Double]	
Minority (rarest occurring value)	`MINORITY`	[wie Eingabe]	
Majority (most frequently occurring value)	`MAJORITY`	[wie Eingabe]	
First quartile	`FIRSTQUARTILE`	[numerisch: Double]	
Third quartile	`THIRDQUARTILE`	[numerisch: Double]	
Interquartile Range (IQR)	`IQR`	[numerisch: Double]	

Pythoncode

Algorithm ID: `native:basicstatisticsforfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Climb along line

Calculates the total climb and descent along line geometries. The input layer must have Z values present. If Z values are not available, the *Drape (set Z value from raster)* algorithm may be used to add Z values from a DEM layer.

The output layer is a copy of the input layer with additional fields that contain the total climb (`climb`), total descent (`descent`), the minimum elevation (`minelev`) and the maximum elevation (`maxelev`) for each line geometry. If the input layer contains fields with the same names as these added fields, they will be renamed (field names will be altered to „name_2“, „name_3“, etc, finding the first non-duplicate name).

Parameter

Bezeichnung	Name	Typ	Beschreibung
Line layer	`INPUT`	[vector: line]	Line layer to calculate the climb for. Must have Z values
Climb layer	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specification of the output (line) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Climb layer	`OUTPUT`	[vector: line]	Line layer containing new attributes with the results from climb calculations.
Total climb	`TOTALCLIMB`	[numerisch: Double]	The sum of the climb for all the line geometries in the input layer
Total descent	`TOTALDESCENT`	[numerisch: Double]	The sum of the descent for all the line geometries in the input layer
Minimum elevation	`MINELEVATION`	[numerisch: Double]	The minimum elevation for the geometries in the layer
Maximum elevation	`MAXELEVATION`	[numerisch: Double]	The maximum elevation for the geometries in the layer

Pythoncode

Algorithm ID: `native:climbalongline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Count points in polygon

Takes a point and a polygon layer and counts the number of points from the point layer in each of the polygons of the polygon layer.

A new polygon layer is generated, with the exact same content as the input polygon layer, but containing an additional field with the points count corresponding to each polygon.

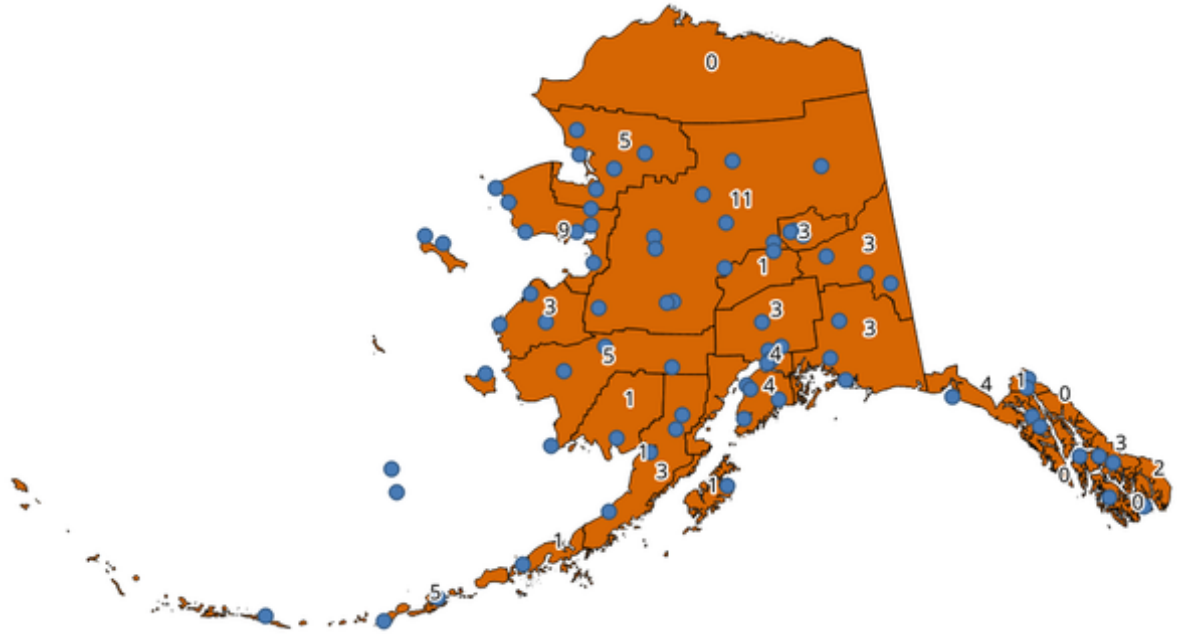


Abb. 24.70: The labels in the polygons show the point count

An optional weight field can be used to assign weights to each point. Alternatively, a unique class field can be specified. If both options are used, the weight field will take precedence and the unique class field will be ignored.

Allows *features in-place modification* of polygon features

`Default menu`: *Vector* ► *Analysis Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Polygone	`POLYGONS`	[Vektor: Polygon]	Polygon layer whose features are associated with the count of points they contain
Points	`POINTS`	[Vektor: Punkt]	Point layer with features to count
Weight field Optional	`WEIGHT`	[tablefield: any]	A field from the point layer. The count generated will be the sum of the weight field of the points contained by the polygon. If the weight field is not numeric, the count will be `0`.

Fortsetzung auf der nächsten Seite

Tab. 24.452 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Class field Optional	`CLASSFIELD`	[tablefield: any]	Points are classified based on the selected attribute and if several points with the same attribute value are within the polygon, only one of them is counted. The final count of the points in a polygon is, therefore, the count of different classes that are found in it.
Count field name	`FIELD`	[Text] Default: ‚NUMPOINTS'	The name of the field to store the count of points
Count	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specification of the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Count	`OUTPUT`	[Vektor: Polygon]	Resulting layer with the attribute table containing the new column with the points count

Pythoncode

Algorithm ID: `native:countpointsinpolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

DBSCAN clustering

Clusters point features based on a 2D implementation of Density-based spatial clustering of applications with noise (DBSCAN) algorithm.

The algorithm requires two parameters, a minimum cluster size, and the maximum distance allowed between clustered points.

Siehe auch

ST-DBSCAN clustering, *K-means clustering*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	Layer to analyze
Minimum cluster size	`MIN_SIZE`	[numerisch: Ganzzahl] Default: 5	Minimum number of features to generate a cluster
Maximum distance between clustered points	`EPS`	[numerisch: Double] Standard: 1,0	Distance beyond which two features can not belong to the same cluster (eps)
Clusters	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the vector layer for the result of the clustering. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Treat border points as noise (DBSCAN*)	`DBSCAN*`	[Boolean] Standard: falsch	If checked, points on the border of a cluster are themselves treated as unclustered points, and only points in the interior of a cluster are tagged as clustered.
Cluster field name	`FIELD_NAME`	[Text] Default: ‚CLUSTER_ID‘	Name of the field where the associated cluster number shall be stored
Cluster size field name	`SIZE_FIELD_NAM`	[Text] Default: ‚CLUSTER_SIZE‘	Name of the field with the count of features in the same cluster

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Clusters	`OUTPUT`	[Vektor: Punkt]	Vector layer containing the original features with a field setting the cluster they belong to
Number of clusters	`NUM_CLUSTERS`	[numerisch: Ganzzahl]	The number of clusters discovered

Pythoncode

Algorithm ID: `native:dbscanclustering`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Distance matrix

Calculates for point features distances to their nearest features in the same layer or in another layer.

Default menu: *Vector* ► *Analysis Tools*

Siehe auch

Join attributes by nearest

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input point layer	`INPUT`	[Vektor: Punkt]	Point layer for which the distance matrix is calculated (**from** points)
Input unique ID field	`INPUT_FIELD`	[tablefield: any]	Field to use to uniquely identify features of the input layer. Used in the output attribute table.
Target point layer	`TARGET`	[Vektor: Punkt]	Point layer containing the nearest point(s) to search (**to** points)
Target unique ID field	`TARGET_FIELD`	[tablefield: any]	Field to use to uniquely identify features of the target layer. Used in the output attribute table.
Output matrix type	`MATRIX_TYPE`	[Aufzählung] Standard: 0	Different types of calculation are available: • 0 — Linear (N * *k* x 3) distance matrix: for each input point, reports the distance to each of the *k* nearest target points. The output matrix consists of up to *k* rows per input point, and each row has three columns: *InputID*, *TargetID* and *Distance*. • 1 — Standard (N x T) distance matrix • 2 — Summary distance matrix (mean, std. dev., min, max): for each input point, reports statistics on the distances to its target points.
Use only the nearest (k) target points	`NEAREST_POINTS`	[numerisch: Ganzzahl] Standard: 0	You can choose to calculate the distance to all the points in the target layer (*0*) or limit to a number (*k*) of closest features.

Fortsetzung auf der nächsten Seite

Tab. 24.457 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Distance matrix	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specification of the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Distance matrix	`OUTPUT`	[Vektor: Punkt]	Point (or MultiPoint for the „Linear (N * *k* x 3)“ case) vector layer containing the distance calculation for each input feature. Its features and attribute table depend on the selected output matrix type.

Pythoncode

Algorithm ID: `qgis:distancematrix`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Distance to nearest hub (line to hub)

Creates lines that join each feature of an input vector to the nearest feature in a destination layer. Distances are calculated based on the *center* of each feature.

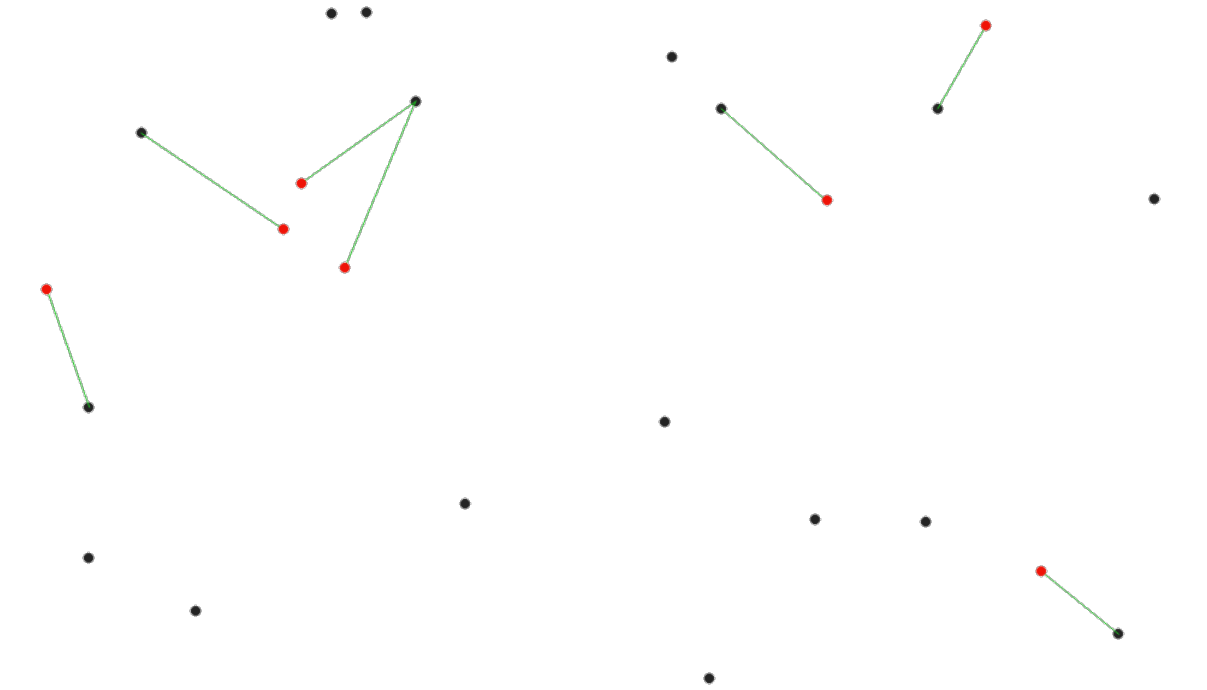


Abb. 24.71: Display the nearest hub for the red input features

Siehe auch

Distance to nearest hub (points), *Join attributes by nearest*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Source points layer	`INPUT`	[Vektor: Geometrie]	Vector layer for which the nearest feature is searched
Destination hubs layer	`HUBS`	[Vektor: Geometrie]	Vector layer containing the features to search for
Hub layer name attribute	`FIELD`	[tablefield: any]	Field to use to uniquely identify features of the destination layer. Used in the output attribute table
Measurement unit	`UNIT`	[Aufzählung] Standard: 0	Units in which to report the distance to the closest feature: • 0 — Meters • 1 — Feet • 2 — Miles • 3 — Kilometers • 4 — Layer units

Fortsetzung auf der nächsten Seite

Tab. 24.459 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Hub distance	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line vector layer connecting the matching points. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Hub distance	`OUTPUT`	[vector: line]	Line vector layer with the attributes of the input features, the identifier of their closest feature and the calculated distance.

Pythoncode

Algorithm ID: `qgis:distancetonearesthublinetohub`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Distance to nearest hub (points)

Creates a point layer representing the *center* of the input features with the addition of two fields containing the identifier of the nearest feature (based on its center point) and the distance between the points.

Siehe auch

Distance to nearest hub (line to hub), *Join attributes by nearest*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Source points layer	`INPUT`	[Vektor: alle]	Vector layer for which the nearest feature is searched
Destination hubs layer	`HUBS`	[Vektor: alle]	Vector layer containing the features to search for

Fortsetzung auf der nächsten Seite

Tab. 24.461 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Hub layer name attribute	`FIELD`	[tablefield: any]	Field to use to uniquely identify features of the destination layer. Used in the output attribute table
Measurement unit	`UNIT`	[Aufzählung] Standard: 0	Units in which to report the distance to the closest feature: • 0 — Meters • 1 — Feet • 2 — Miles • 3 — Kilometers • 4 — Layer units
Hub distance	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output point vector layer with the nearest hub. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Hub distance	`OUTPUT`	[Vektor: Punkt]	Point vector layer representing the center of the source features with their attributes, the identifier of their closest feature and the calculated distance.

Pythoncode

Algorithm ID: `qgis:distancetonearesthubpoints`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Join by lines (hub lines)

Creates hub and spoke diagrams by connecting lines from points on the Spoke layer to matching points in the Hub layer.

Determination of which hub goes with each point is based on a match between the Hub ID field on the hub points and the Spoke ID field on the spoke points.

If input layers are not point layers, a point on the surface of the geometries will be taken as the connecting location.

Optionally, geodesic lines can be created, which represent the shortest path on the surface of an ellipsoid. When geodesic mode is used, it is possible to split the created lines at the antimeridian (±180 degrees longitude), which can improve rendering of the lines. Additionally, the distance between vertices can be specified. A smaller distance results in a denser, more accurate line.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

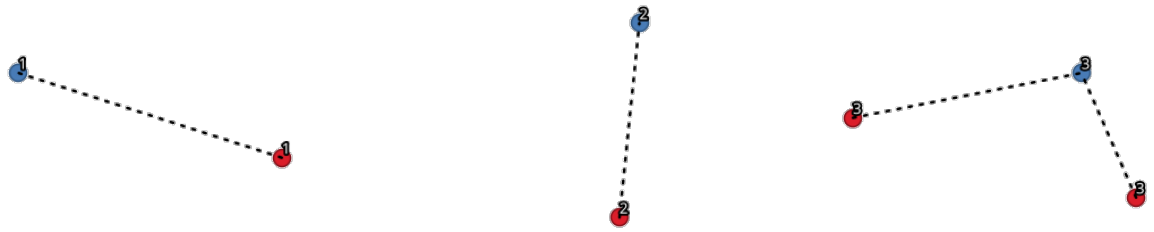


Abb. 24.72: Join points based on a common field / attribute

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Hub layer	`HUBS`	[Vektor: Geometrie]	Input layer
Hub ID field	`HUB_FIELD`	[tablefield: any]	Field of the hub layer with ID to join
Hub layer fields to copy (leave empty to copy all fields) Optional	`HUB_FIELDS`	[tablefield: any] [list]	The field(s) of the hub layer to be copied. If no field(s) are chosen all fields are taken.
Spoke layer	`SPOKES`	[Vektor: Geometrie]	Additional spoke point layer
Spoke ID field	`SPOKE_FIELD`	[tablefield: any]	Field of the spoke layer with ID to join
Spoke layer fields to copy (leave empty to copy all fields) Optional	`SPOKE_FIELDS`	[tablefield: any] [list]	Field(s) of the spoke layer to be copied. If no fields are chosen all fields are taken.

Fortsetzung auf der nächsten Seite

Tab. 24.463 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Create geodesic lines	`GEODESIC`	[Boolean] Standard: falsch	Create geodesic lines (the shortest path on the surface of an ellipsoid)
Hub lines	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output hub line vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Distance between vertices (geodesic lines only)	`GEODESIC_DISTA`	[numerisch: Double] Default: 1000.0 (kilometers)	Distance between consecutive vertices (in kilometers). A smaller distance results in a denser, more accurate line
Split lines at antimeridian (±180 degrees longitude)	`ANTIMERIDIAN_S`	[Boolean] Standard: falsch	Split lines at ±180 degrees longitude (to improve rendering of the lines)

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Hub lines	`OUTPUT`	[vector: line]	The resulting line layer connecting matching points in input layers

Pythoncode

Algorithm ID: `native:hublines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

K-means clustering

Calculates the 2D distance based k-means cluster number for each input feature.

K-means clustering aims to partition the features into k clusters in which each feature belongs to the cluster with the nearest mean. The mean point is represented by the barycenter of the clustered features.

If input geometries are lines or polygons, the clustering is based on the centroid of the feature.

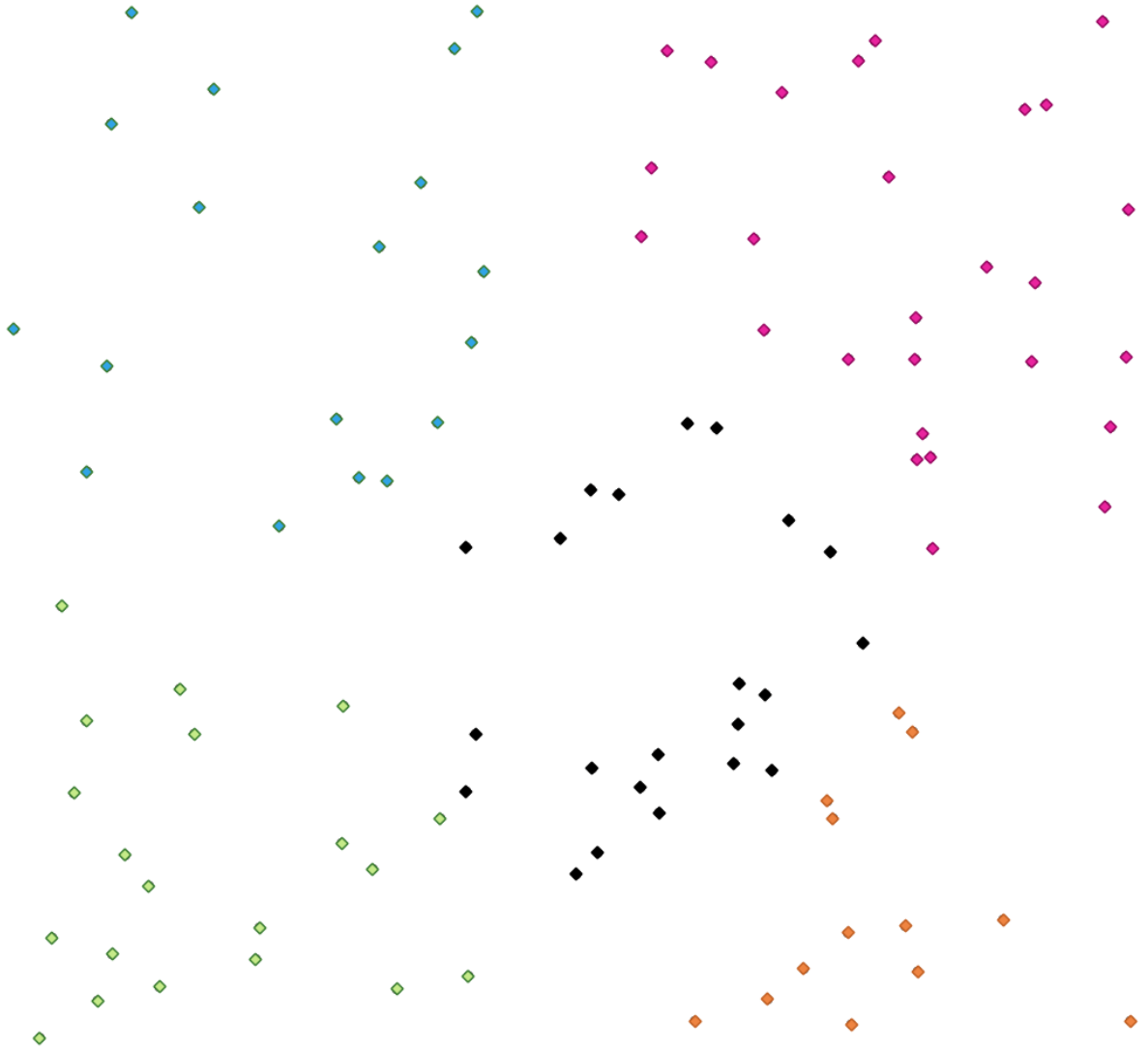


Abb. 24.73: A five class point clusters

Siehe auch

DBSCAN clustering, *ST-DBSCAN clustering*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Layer to analyze
Number of clusters	`CLUSTERS`	[numerisch: Ganzzahl] Default: 5	Number of clusters to create with the features
Methode	`METHOD`	[Aufzählung] Standard: 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`	[vector: same as input] Default: `[Create temporary layer]`	Specify the output vector layer for the generated clusters. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Cluster field name	`FIELD_NAME`	[Text] Default: ‚CLUSTER_ID'	Name of the field where the associated cluster number shall be stored
Cluster size field name	`SIZE_FIELD_NAM`	[Text] Default: ‚CLUSTER_SIZE'	Name of the field with the count of features in the same cluster

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Clusters	`OUTPUT`	[vector: same as input]	Vector layer containing the original features with fields specifying the cluster they belong to and their number in it

Pythoncode

Algorithm ID: `native:kmeansclustering`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

List unique values

Lists unique values of an attribute table field and counts their number.

Default menu: *Vector* ► *Analysis Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Layer to analyze
Target field(s)	`FIELDS`	[tablefield: any] [list]	Field(s) to analyze
Unique values Optional	`OUTPUT`	[vector: table] Default: `[Create temporary layer]`	Specify the summary table layer with unique values. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
HTML report Optional	`OUTPUT_HTML_FI`	[html] Default: `[Save to temporary file]`	HTML report of unique values in the *Processing* ► *Results viewer*. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Unique values	`OUTPUT`	[vector: table]	Summary table layer with unique values
HTML report	`OUTPUT_HTML_FI`	[html]	HTML report of unique values. Can be opened from the *Processing* ► *Results viewer*
Total unique values	`TOTAL_VALUES`	[numerisch: Ganzzahl]	The number of unique values in the input field
Unique values concatenated	`UNIQUE_VALUES`	[Text]	A string with the comma separated list of unique values found in the input field

Pythoncode

Algorithm ID: `native:listuniquevalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Mean coordinate(s)

Computes a point layer with the center of mass of geometries in an input layer.

An attribute can be specified as containing weights to be applied to each feature when computing the center of mass.

If an attribute is selected in the parameter, features will be grouped according to values in this field. Instead of a single point with the center of mass of the whole layer, the output layer will contain a center of mass for the features in each category.

Default menu: *Vector* ► *Analysis Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Weight field Optional	`WEIGHT`	[tablefield: numeric]	Field to use if you want to perform a weighted mean
Unique ID field	`UID`	[tablefield: any]	Unique field on which the calculation of the mean will be made

Fortsetzung auf der nächsten Seite

Tab. 24.471 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Mean coordinates	`OUTPUT`	[Vektor: Punkt] Default: `[Create temporary layer]`	Specify the (point vector) layer for the result. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Mean coordinates	`OUTPUT`	[Vektor: Punkt]	Resulting point(s) layer

Pythoncode

Algorithm ID: `native:meancoordinates`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Nearest neighbour analysis

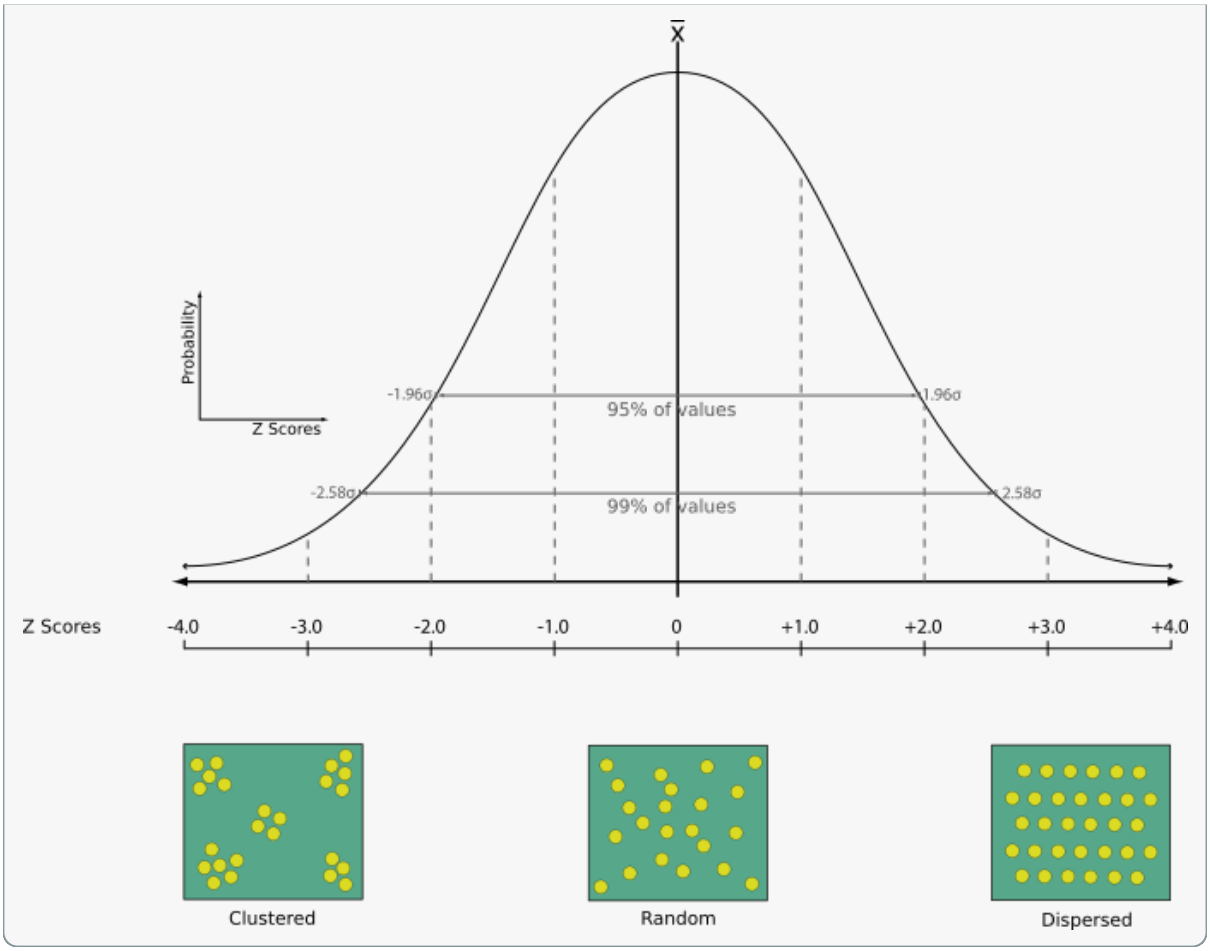
Performs nearest neighbor analysis for a point layer. The output tells you how your data are distributed (clustered, randomly or distributed).

Output is generated as an HTML file with the computed statistical values:

- Observed mean distance
- Expected mean distance
- Nearest neighbour index
- Number of points
- Z-Score: Comparing the Z-Score with the normal distribution tells you how your data are distributed. A low Z-Score means that the data are unlikely to be the result of a spatially random process, while a high Z-Score means that your data are likely to be a result of a spatially random process.

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.



Default menu: *Vector* ► *Analysis Tools*

Siehe auch

Join attributes by nearest

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	Point vector layer to calculate the statistics on
Nearest neighbour Optional	`OUTPUT_HTML_FI`	[html] Default: `[Save to temporary file]`	Specification of the HTML file for the computed statistics. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Nearest neighbour	`OUTPUT_HTML_FI`	[html]	HTML file with the computed statistics
Observed mean distance	`OBSERVED_MD`	[numerisch: Double]	Observed mean distance
Expected mean distance	`EXPECTED_MD`	[numerisch: Double]	Expected mean distance
Nearest neighbour index	`NN_INDEX`	[numerisch: Ganzzahl]	Nearest neighbour index
Number of points	`POINT_COUNT`	[numerisch: Ganzzahl]	Number of points
Z-Score	`Z_SCORE`	[numerisch: Double]	Z-Score

Pythoncode

Algorithm ID: `native:nearestneighbouranalysis`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Overlap analysis

Calculates the area and percentage cover by which features from an input layer are overlapped by features from a selection of overlay layers.

New attributes are added to the output layer reporting the total area of overlap and percentage of the input feature overlapped by each of the selected overlay layers.

> **Bemerkung**
>
> This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	The input layer.
Overlay layers	`LAYERS`	[vector: polygon] [list]	The overlay layers.

Fortsetzung auf der nächsten Seite

Tab. 24.475 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Overlap	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Grid size Optional	`GRID_SIZE`	[numerisch: Double] Standard: Nicht gesetzt	If provided, the input geometries are snapped to a grid of the given size, and the result vertices are computed on that same grid.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Overlap	`OUTPUT`	[wie Eingabe]	The output layer with additional fields reporting the overlap (in map units and percentage) of the input feature overlapped by each of the selected layers.

Pythoncode

Algorithm ID: `native:calculatevectoroverlaps`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Shortest line between features

Creates a line layer as the shortest line between the source and the destination layer. By default only the first nearest feature of the destination layer is taken into account. The n-nearest neighboring features number can be specified. If a maximum distance is specified, then only features which are closer than this distance will be considered.

The output features will contain all the source layer attributes, all the attributes from the n-nearest feature and the additional field of the distance.

Wichtig

This algorithm uses purely Cartesian calculations for distance, and does not consider geodetic or ellipsoid properties when determining feature proximity. The measurement and output coordinate system is based on the coordinate system of the source layer.

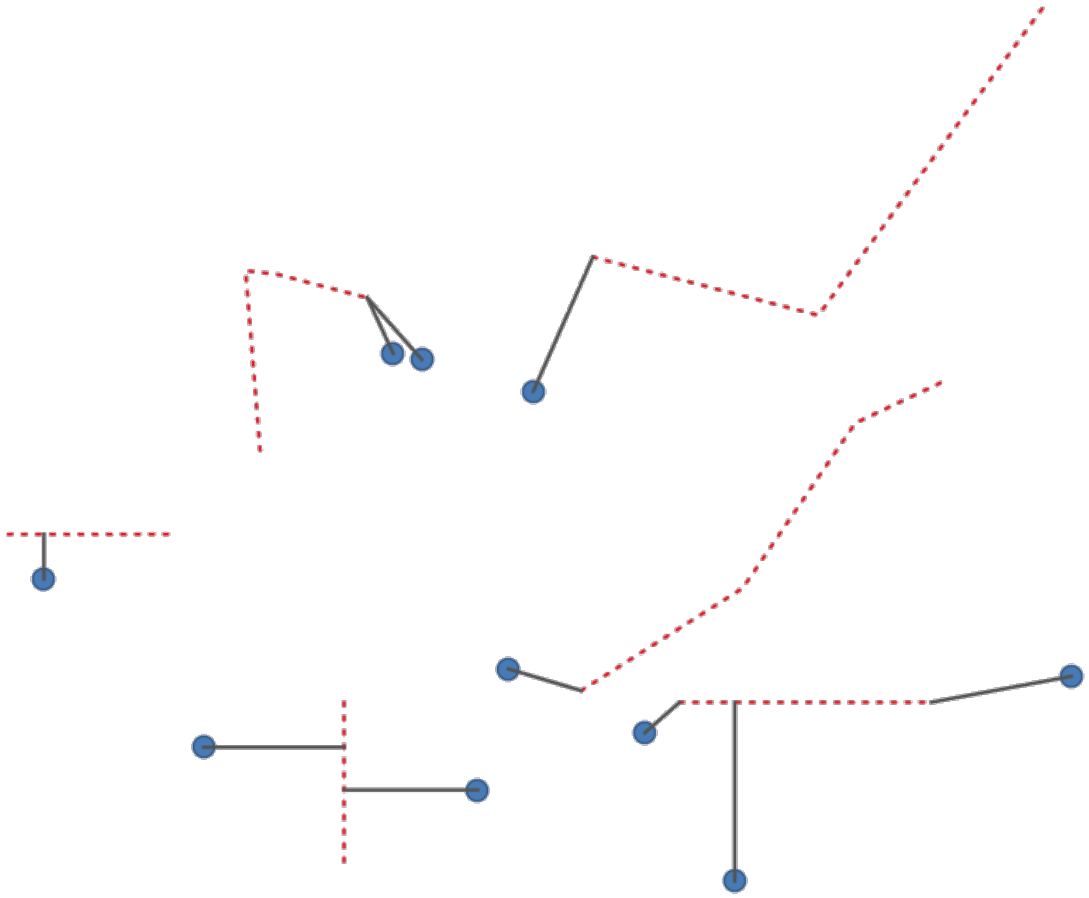


Abb. 24.74: Shortest line from point features to lines

Parameter

Bezeichnung	Name	Typ	Beschreibung
Source layer	`SOURCE`	[Vektor: Geometrie]	Origin layer for which to search for nearest neighbors
Destination layer	`DESTINATION`	[Vektor: Geometrie]	Target Layer in which to search for nearest neighbors
Methode	`METHOD`	[Aufzählung] Standard: 0	Shortest distance calculation method Possible values are: • 0 — Distance to nearest point on feature • 1 — Distance to feature centroid
Maximum number of neighbors	`NEIGHBORS`	[numerisch: Ganzzahl] Standard: 1	Maximum number of neighbors to look for
Maximum distance Optional	`DISTANCE`	[numerisch: Double]	Only destination features which are closer than this distance will be considered.
Shortest lines	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[vector: line]	Line vector layer joining source features to their nearest neighbor(s) in the destination layer. Contains all attributes for both source and destination features, and the computed distance.

Pythoncode

Algorithm ID: `native:shortestline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

ST-DBSCAN clustering

Clusters point features based on a 2D implementation of spatiotemporal Density-based clustering of applications with noise (ST-DBSCAN) algorithm.

Siehe auch

DBSCAN clustering, *K-means clustering*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	Layer to analyze
Date/time field	`DATETIME_FIELD`	[tablefield: date]	Field containing the temporal information
Minimum cluster size	`MIN_SIZE`	[numerisch: Ganzzahl] Default: 5	Minimum number of features to generate a cluster
Maximum distance between clustered points	`EPS`	[numerisch: Double] Standard: 1,0	Distance beyond which two features can not belong to the same cluster (eps)
Maximum time duration between clustered points	`EPS2`	[numerisch: Double] Default: 0.0 (days)	Time duration beyond which two features can not belong to the same cluster (eps2). Available time units are milliseconds, seconds, minutes, hours, days and weeks.
Clusters	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the vector layer for the result of the clustering. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Treat border points as noise (DBSCAN*)	`DBSCAN*`	[Boolean] Standard: falsch	If checked, points on the border of a cluster are themselves treated as unclustered points, and only points in the interior of a cluster are tagged as clustered.
Cluster field name	`FIELD_NAME`	[Text] Default: ‚CLUSTER_ID‘	Name of the field where the associated cluster number shall be stored
Cluster size field name	`SIZE_FIELD_NAM`	[Text] Default: ‚CLUSTER_SIZE‘	Name of the field with the count of features in the same cluster

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Clusters	`OUTPUT`	[Vektor: Punkt]	Vector layer containing the original features with a field setting the cluster they belong to
Number of clusters	`NUM_CLUSTERS`	[numerisch: Ganzzahl]	The number of clusters discovered

Pythoncode

Algorithm ID: `native:stdbscanclustering`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Statistics by categories

Calculates statistics of a field depending on a parent class. The parent class is a combination of values from other fields.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input vector layer	`INPUT`	[Vektor: alle]	Input vector layer with unique classes and values
Field to calculate statistics on (if empty, only count is calculated) Optional	`VALUES_FIELD_N`	[tablefield: any]	If empty only the count will be calculated
Field(s) with categories	`CATEGORIES_FIE`	[tablefield: any] [list]	The fields that (combined) define the categories
Statistics by category	`OUTPUT`	[vector: table] Standard: `[Create temporary layer]`	Specify the output table for the generated statistics. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Statistics by category	`OUTPUT`	[vector: table]	Table containing the statistics

Depending on the type of the field being analyzed, the following statistics are returned for each grouped value:

Statistik	Text	Numeric	Datum
Count (`COUNT`)	☑	☑	☑
Unique values (`UNIQUE`)	☑		☑
Empty (null) values (`EMPTY`)	☑		☑
Non-empty values (`FILLED`)	☑		☑
Minimal value (`MIN`)	☑	☑	☑
Maximal value (`MAX`)	☑	☑	☑
Range (`RANGE`)		☑	
Sum (`SUM`)		☑	
Mean value (`MEAN`)		☑	
Median value (`MEDIAN`)		☑	
Standard Deviation (`STD_DEV`)		☑	
Coefficient of variation (`CV`)		☑	
Minority (rarest occurring value - `MINORITY`)		☑	
Majority (most frequently occurring value - `MAJORITY`)		☑	
First Quartile (`FIRSTQUARTILE`)		☑	
Third Quartile (`THIRDQUARTILE`)		☑	
Inter Quartile Range (`IQR`)		☑	
Minimum Length (`MIN_LENGTH`)	☑		
Mean Length (`MEAN_LENGTH`)	☑		
Maximum Length (`MAX_LENGTH`)	☑		

Pythoncode

Algorithm ID: `qgis:statisticsbycategories`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Sum line lengths

Takes a polygon layer and a line layer and measures the total length of lines and the total number of them that cross each polygon.

The resulting layer has the same features as the input polygon layer, but with two additional attributes containing the length and count of the lines across each polygon.

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Allows *features in-place modification* of polygon features

Default menu: *Vector* ► *Analysis Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Linien	`LINES`	[vector: line]	Input vector line layer
Polygone	`POLYGONS`	[Vektor: Polygon]	Polygon vector layer
Lines length field name	`LEN_FIELD`	[Text] Default: ‚LENGTH‘	Name of the field for the lines length
Lines count field name	`COUNT_FIELD`	[Text] Default: ‚COUNT‘	Name of the field for the lines count
Line length	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output polygon layer with generated statistics. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Line length	`OUTPUT`	[Vektor: Polygon]	Polygon output layer with fields of lines length and line count

Pythoncode

Algorithm ID: `native:sumlinelengths`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.23 Vector coverage

Dissolve coverage

Operates on a coverage (represented as a set of polygon features with exactly matching edge geometry) to dissolve (union) the geometries. It provides a heavily optimized approach for unioning these features compared with the standard Dissolve tools.

Siehe auch

Dissolve

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Input polygon vector layer
Dissolved	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Dissolved	`OUTPUT`	[Vektor: Polygon]	The output polygon vector layer with dissolved geometries.

Python-Code

Algorithm ID: `native:coverageunion`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Simplify coverage

Operates on a coverage (represented as a set of polygon features with exactly matching edge geometry) to apply a Visvalingam–Whyatt simplification to the edges, reducing complexity in proportion with the provided tolerance, while retaining a valid coverage (i.e. no edges will cross or touch after the simplification). Geometries will never be removed, but they may be simplified down to just a triangle. Also, some geometries (such as polygons which have too few non-repeated points) will be returned unchanged.

If the input dataset is not a valid coverage due to overlaps, it will still be simplified, but invalid topology such as crossing edges will still be invalid.

Achtung

Running this algorithm requires QGIS installed with GEOS >= 3.12 (see *Help* ► *About* menu).

Siehe auch

Simplify, *Smooth*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Input polygon vector layer
Tolerance	`TOLERANCE`	[numerisch: Double] Default: 0.0	The maximum distance (in unit of choice) between two consecutive vertices to get merged.
Preserve boundary	`PRESERVE_BOUND`	[Boolean] Standard: falsch	When enabled, the outside edges of the coverage will be preserved without simplification
Simplified	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Simplified	`OUTPUT`	[Vektor: Polygon]	The output polygon vector layer, with a lower number of vertices.

Python-Code

Algorithm ID: `native:coveragesimplify`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Validate coverage

Analyzes a coverage (represented as a set of polygon features with exactly matching edge geometry) to find places where the assumption of exactly matching edges is not met. Invalidity includes polygons that overlap or that have gaps smaller than the specified gap width.

Achtung

Running this algorithm requires QGIS installed with GEOS >= 3.12 (see *Help* ► *About* menu).

Siehe auch

Check validity, *Small gaps*, *Overlaps*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Input polygon vector layer
Gap width	`GAP_WIDTH`	[numerisch: Double] Default: 0.0	The maximum width of gap to detect
Invalid edges	`INVALID_EDGES`	[vector: line] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Invalid edges	`OUTPUT`	[vector: line]	The output line vector layer showing error edges for each invalid polygon.
Validity result	`IS_VALID`	[Boolean]	Returns whether the coverage is valid or not.

Python-Code

Algorithm ID: `native:coveragevalidate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.24 Vector creation

Array of offset (parallel) lines

Creates copies of line features in a layer, by creating multiple offset versions of each feature. Each new version is incrementally offset by a specified distance.

Positive distance will offset lines to the left, and negative distances will offset them to the right.

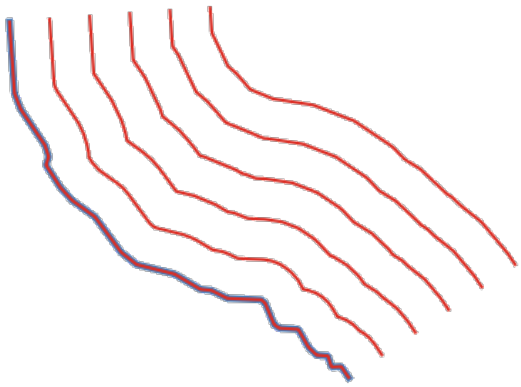


Abb. 24.75: In Blau der Quell-Layer, in Rot der Offset-Layer

Allows *features in-place modification* of line features

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Offset lines, *Array of translated features*

Parameter

Grundlegende Parameter

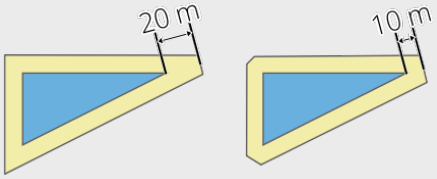
Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer to use for the offsets.
Number of features to create	`COUNT`	[numeric: integer] Default: 10	Number of offset copies to generate for each feature
Offset step distance	`OFFSET`	[numeric: double] Standard: 1,0	Distance between two consecutive offset copies
Offset lines	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line layer with offset features. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Segmente	`SEGMENTS`	[numerisch: Double] Default: 8	Number of line segments to use to approximate a quarter circle when creating rounded offsets
Verbindungsstil	`JOIN_STYLE`	[Aufzählung] Standard: 0	Specify whether round, miter or beveled joins should be used when offsetting corners in a line. One of: • 0 — Rund • 1 — Eckig • 2 — Abgefast Abb. 24.76: Runder, eckiger und abgefaster Verbindungsstil

Fortsetzung auf der nächsten Seite

Tab. 24.495 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Eckgrenze	`MITER_LIMIT`	[numerisch: Double] Default: 2.0	Sets the maximum distance from the offset geometry to use when creating a mitered join as a factor of the offset distance (only applicable for miter join styles). Minimum: 1.0  Abb. 24.77: A 10m buffer with a limit of 2 and a 10m buffer with a limit of 1

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Offset lines	`OUTPUT`	[vector: line]	Output line layer with the original and offset features. Attributes will be copied to the corresponding outputs. The following attributes related to the input parameters will be added: `instance` indexing the `COUNT` offset lines and `offset` designating the offset distance from the original line.

Pythoncode

Algorithm ID: `native:arrayoffsetlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Array of translated features

Creates copies of features in a layer by creating multiple translated versions of each. Each copy is incrementally displaced by a preset amount in the X, Y and/or Z axis.

M values present in the geometry can also be translated.

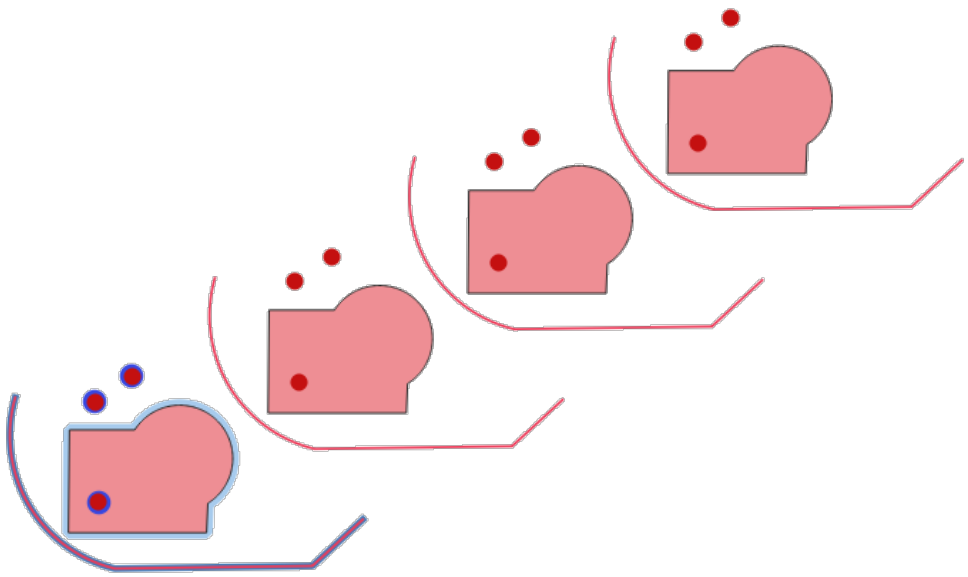


Abb. 24.78: Input layers in blue tones, output layers with translated features in red tones

Allows *features in-place modification* of point, line, and polygon features

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Translate, *Array of offset (parallel) lines*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Input vector layer to translate
Number of features to create	`COUNT`	[numeric: integer] Default: 10	Number of copies to generate for each feature
Step distance (x-axis)	`DELTA_X`	[numeric: double] Default: 0.0	Displacement to apply on the X axis
Step distance (y-axis)	`DELTA_Y`	[numeric: double] Default: 0.0	Displacement to apply on the Y axis
Step distance (z-axis)	`DELTA_Z`	[numeric: double] Default: 0.0	Displacement to apply on the Z axis

Fortsetzung auf der nächsten Seite

Tab. 24.497 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Step distance (m values)	`DELTA_M`	[numeric: double] Default: 0.0	Displacement to apply on M
Translated	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Output vector layer with translated (moved) copies of the features. The original features are also copied. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Translated	`OUTPUT`	[wie Eingabe]	Output vector layer with translated (moved) copies of the features. The original features are also copied.

Pythoncode

Algorithm ID: `native:arraytranslatedfeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create grid

Creates a vector layer with a grid covering a given extent. Grid cells can have different shapes:

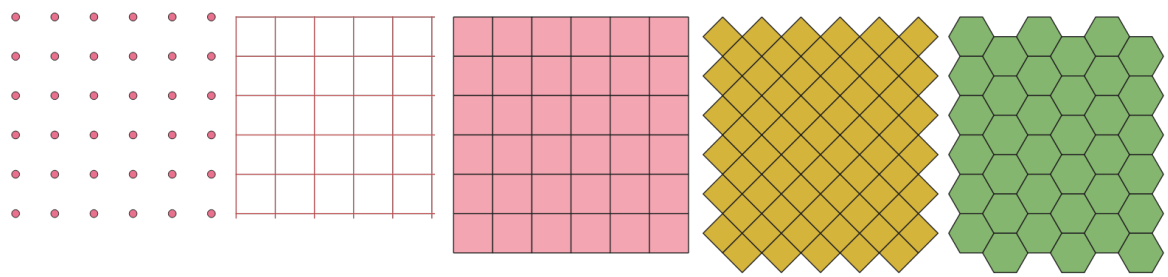


Abb. 24.79: Different grid cell shapes applied to the same extent, without overlaps

The size and/or placement of each element in the grid is defined using a horizontal and vertical spacing. The CRS of the output layer must be defined. The grid extent and the spacing values must be expressed in the coordinates and units of this CRS.

Default menu: *Vector ► Research Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Grid type	`TYPE`	[Aufzählung] Standard: 0	Shape of the grid. One of: • 0 — Point • 1 — Line • 2 — Rectangle (polygon) • 3 — Diamond (polygon) • 4 — Hexagon (polygon)
Grid extent	`EXTENT`	[Ausdehnung]	Extent of the grid Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Horizontal spacing	`HSPACING`	[numerisch: Double] Standard: 1,0	Size of a grid cell on the X-axis
Vertical spacing	`VSPACING`	[numerisch: Double] Standard: 1,0	Size of a grid cell on the Y-axis
Horizontal overlay	`HOVERLAY`	[numerisch: Double] Default: 0.0	Overlay distance between two consecutive grid cells on the X-axis
Vertical overlay	`VOVERLAY`	[numerisch: Double] Default: 0.0	Overlay distance between two consecutive grid cells on the Y-axis
Grid CRS	`CRS`	[KBS] Default: *Project CRS*	Coordinate reference system to apply to the grid

Fortsetzung auf der nächsten Seite

Tab. 24.499 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Grid	`OUTPUT`	[Vektor: Geometrie] Standard: `[Create temporary layer]`	Resulting vector grid layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Grid	`OUTPUT`	[Vektor: Geometrie]	Resulting vector grid layer. The output geometry type (point, line or polygon) depends on the *Grid type*. Features are created from top to bottom, left to right. The attribute table is filled with: • an `id` • coordinates on the `left`, `right`, `top` and `bottom` sides • and their placement in the grid: `row_index` and `column_index` (available for point, rectangle and hexagon grid types)

Pythoncode

Algorithm ID: `native:creategrid`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create layer from extent

Creates a new vector layer that contains a single feature with geometry matching the extent of the input layer.

It can be used in models to convert a literal extent (`xmin`, `xmax`, `ymin`, `ymax` format) into a layer which can be used for other algorithms which require a layer based input.

Siehe auch

Create layer from point

Parameter

Bezeichnung	Name	Typ	Beschreibung
Extent (xmin, xmax, ymin, ymax)	`INPUT`	[Ausdehnung]	Input extent Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Extent	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extent	`OUTPUT`	[Vektor: Polygon]	The output (extent) vector layer

Pythoncode

Algorithm ID: `native:extenttolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create layer from point

Creates a new vector layer that contains a single feature with geometry matching a point parameter. It can be used in models to convert a point into a point layer for algorithms which require a layer based input.

Siehe auch

Create layer from extent

Parameter

Bezeichnung	Name	Typ	Beschreibung
Point	`INPUT`	[coordinates]	Input point, including CRS info (example: `397254,6214446 [EPSG:32632]`). If the CRS is not provided, the Project CRS will be used. The point can be specified by clicking on the map canvas.
Point	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Point	`OUTPUT`	[Vektor: Punkt]	The output point vector layer containing the input point.

Pythoncode

Algorithm ID: `native:pointtolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create points layer from table

Creates points layer from a table with columns that contain coordinates fields.

Besides X and Y coordinates you can also specify Z and M fields.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Input vector layer or a table.
X field	`XFIELD`	[tablefield: any]	Field containing the X coordinate
Y field	`YFIELD`	[tablefield: any]	Field containing the Y coordinate
Z field Optional	`ZFIELD`	[tablefield: any]	Field containing the Z coordinate
M field Optional	`MFIELD`	[tablefield: any]	Field containing the M value
Ziel KBS	`TARGET_CRS`	[KBS] Default: `EPSG:4326`	Coordinate reference system to use for layer. The provided coordinates are assumed to be compliant.
Points from table	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the resulting point layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Points from table	`OUTPUT`	[Vektor: Punkt]	The resulting point layer

Pythoncode

Algorithm ID: `native:createpointslayerfromtable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Generate points (pixel centroids) along line

Generates a point vector layer from an input raster and line layer.

The points correspond to the pixel centroids that intersect the line layer.

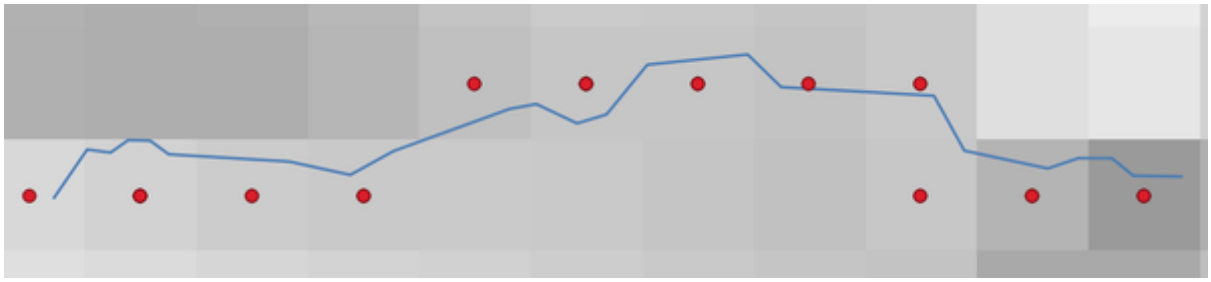


Abb. 24.80: Points of the pixel centroids

Parameter

Bezeichnung	Name	Typ	Beschreibung
Raster layer	`INPUT_RASTER`	[Raster]	Input raster layer
Vector layer	`INPUT_VECTOR`	[vector: line]	Input line vector layer
Points along line	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Resulting point layer with pixel centroids. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Points along line	`OUTPUT`	[Vektor: Punkt]	Resulting point layer with pixel centroids

Pythoncode

Algorithm ID: `qgis:generatepointspixelcentroidsalongline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Generate points (pixel centroids) inside polygon

Generates a point vector layer from an input raster and polygon layer.

The points correspond to the pixel centroids that intersect the polygon layer.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

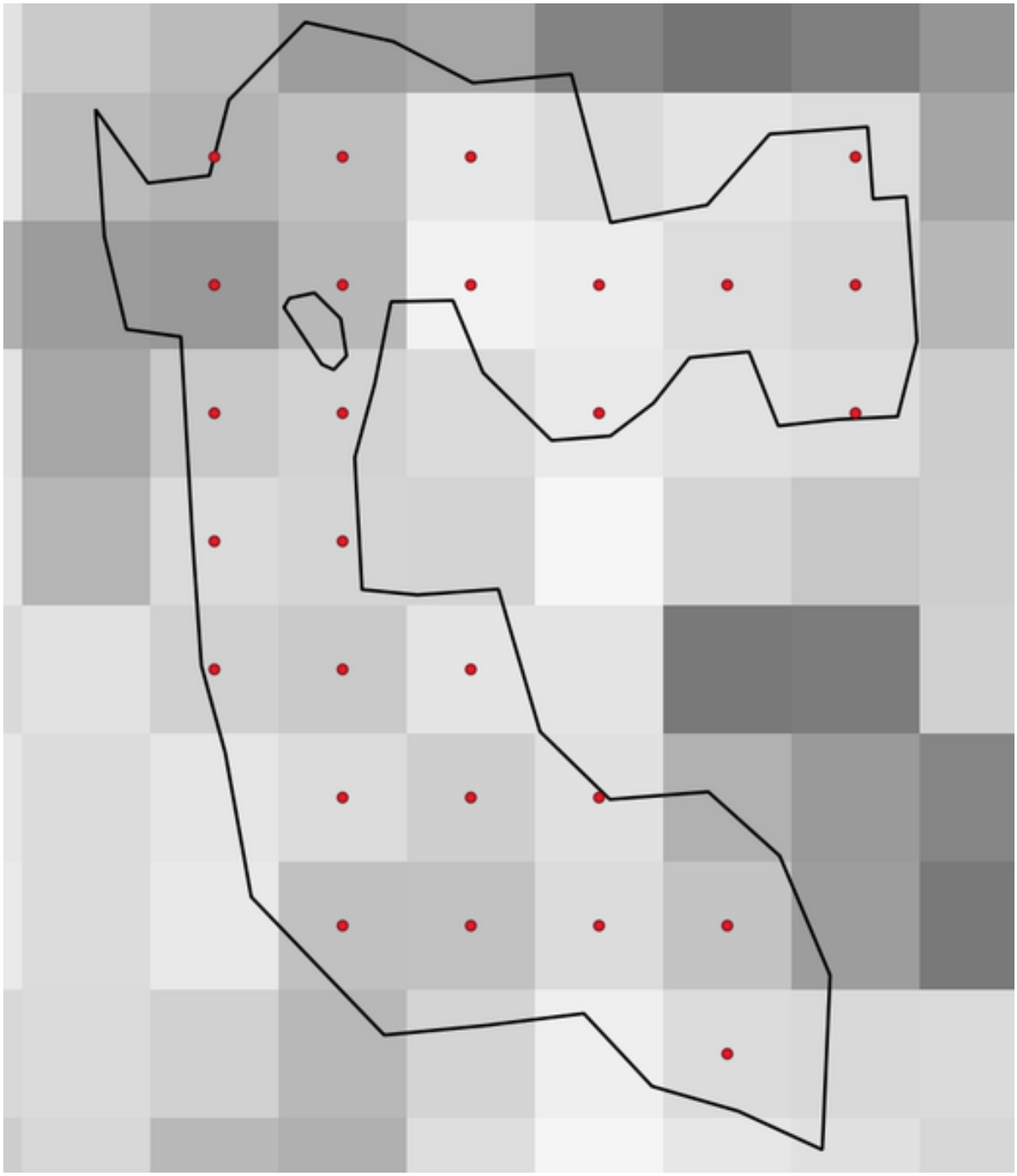


Abb. 24.81: Points of the pixel centroids

Parameter

Bezeichnung	Name	Typ	Beschreibung
Raster layer	`INPUT_RASTER`	[Raster]	Input raster layer
Vector layer	`INPUT_VECTOR`	[Vektor: Polygon]	Input polygon vector layer
Points inside polygons	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Resulting point layer of pixel centroids. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Points inside polygons	`OUTPUT`	[Vektor: Punkt]	Resulting point layer of pixel centroids

Pythoncode

Algorithm ID: `native:generatepointspixelcentroidsinsidepolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Import geotagged photos

Creates a point layer corresponding to the geotagged locations from JPEG or HEIC/HEIF images from a source folder.

The point layer will contain a single PointZ feature per input file from which the geotags could be read. Any altitude information from the geotags will be used to set the point's Z value.

Besides longitude and latitude also altitude, direction and timestamp information, if present in the photo, will be added to the point as attributes.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input folder	`FOLDER`	[Verzeichnis]	Path to the source folder containing the geotagged photos
Scan recursively	`RECURSIVE`	[Boolean] Standard: falsch	If checked, the folder and its subfolders will be scanned
Photos Optional	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the point vector layer for the geotagged photos. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Invalid photos table Optional	`INVALID`	[vector: table] Standard: `[Ausgabe überspringen]`	Specify the table of unreadable or non-geotagged photos. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Photos	`OUTPUT`	[Vektor: Punkt]	Point vector layer with geotagged photos. The form of the layer is automatically filled with paths and photo previews settings.
Invalid photos table Optional	`INVALID`	[vector: table]	Table of unreadable or non-geotagged photos can also be created.

Pythoncode

Algorithm ID: `native:importphotos`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Points to path

Converts a point layer to a line layer, by joining points in an order defined by an expression or a field in the input point layer.

Points can be grouped by a field or an expression to distinguish line features.

In addition to the line vector layer, a text file is output that describes the resulting line as a start point and a sequence of bearings / directions (relative to azimuth) and distances.

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input point layer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Create closed paths	`CLOSE_PATH`	[Boolean] Standard: falsch	If checked, the first and last points of the line will be connected and close the generated path
Order expression Optional	`ORDER_EXPRESSI`	[Ausdruck]	Field or expression providing the order to connect the points in the path. If not set, the feature ID (`$id`) is used.
Sort text containing numbers naturally	`NATURAL_SORT`	[Boolean] Standard: falsch	If checked, naturally sorts the features based on the provided expression (i.e., ‚a9' < ‚a10').
Path group expression Optional	`GROUP_EXPRESSI`	[Ausdruck]	Point features of the same value in the field or expression will be grouped in the same line. If not set, a single path is drawn with all the input points.
Paths	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the line vector layer of the path. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Directory for text output Optional	`OUTPUT_TEXT_DI`	[Verzeichnis] Standard: `[Ausgabe überspringen]`	Specify the directory that will contain the description files of points and paths. *One of*: • Ausgabe übersrpingen • In einem temporären Verzeichnis speichern • In Verzeichnis speichern

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Paths	`OUTPUT`	[vector: line]	Line vector layer of the path
Directory for text output	`OUTPUT_TEXT_DI`	[Verzeichnis]	Directory containing description files of points and paths

Pythoncode

Algorithm ID: `native:pointstopath`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Random points along line

Creates a new point layer, with points placed on the lines of another layer.

For each line in the input layer, a given number of points is added to the resulting layer. The procedure for adding a point is to:

1. randomly select a line feature from the input layer
2. if the feature is multi-part, randomly select a part of it
3. randomly select a segment of that line
4. randomly select a position on that segment.

The procedure means that curved parts of the lines (with relatively short segments) will get more points than straight parts (with relatively long segments), as demonstrated in the illustration below, where the output of the *Random points along lines* algorithm can be compared with the output of the *Random points on lines* algorithm (that produces points with an, on average, even distribution along the lines).

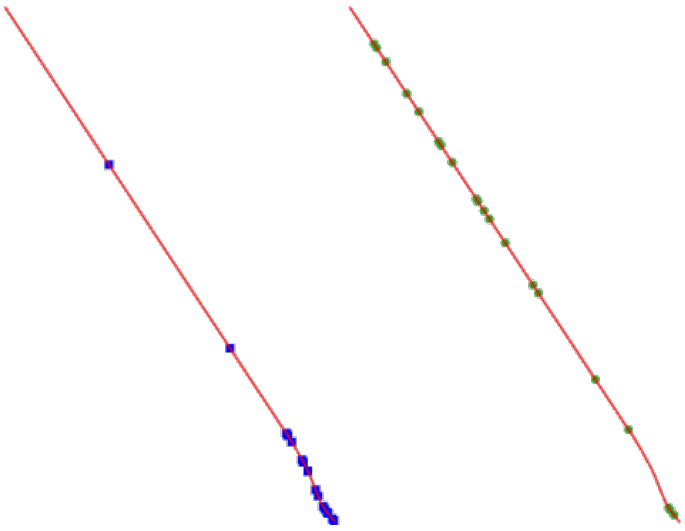


Abb. 24.82: Example algorithm output. Left: *Random points along line*, right: *Random points on lines*

A minimum distance can be specified, to avoid points being too close to each other.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Random points on lines

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input point layer	`INPUT`	[vector: line]	Input line vector layer
Number of points	`POINTS_NUMBER`	[numerisch: Ganzzahl] Standard: 1	Number of points to create
Minimum distance between points	`MIN_DISTANCE`	[numerisch: Double] Default: 0.0	The minimum distance between points
Random points	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	The output random points. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Random points	`OUTPUT`	[Vektor: Punkt]	The output random points layer.

Pythoncode

Algorithm ID: `qgis:qgisrandompointsalongline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Random points in extent

Creates a new point layer with a given number of random points, all of them within a given extent.

A distance factor can be specified, to avoid points being too close to each other. If the minimum distance between points makes it impossible to create new points, either distance can be decreased or the maximum number of attempts may be increased.

Default menu: *Vector* ► *Research Tools*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input extent	`EXTENT`	[Ausdehnung]	Map extent for the random points Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Number of points	`POINTS_NUMBER`	[numerisch: Ganzzahl] Standard: 1	Number of point to create
Minimum distance between points	`MIN_DISTANCE`	[numerisch: Double] Default: 0.0	The minimum distance between points
Ziel KBS	`TARGET_CRS`	[KBS] Default: *Project CRS*	CRS of the random points layer
Random points	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	The output random points. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Maximum number of search attempts given the minimum distance	`MAX_ATTEMPTS`	[numerisch: Ganzzahl] Default: 200	Maximum number of attempts to place the points

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Random points	`OUTPUT`	[Vektor: Punkt]	The output random points layer.

Pythoncode

Algorithm ID: `native:randompointsinextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Random points in layer bounds

Creates a new point layer with a given number of random points, all of them within the extent of a given layer.

A minimum distance can be specified, to avoid points being too close to each other.

Default menu: *Vector ► Research Tools*

> **Warnung**
>
> This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Input polygon layer defining the area
Number of points	`POINTS_NUMBER`	[numerisch: Ganzzahl] Standard: 1	Number of points to create
Minimum distance between points	`MIN_DISTANCE`	[numerisch: Double] Default: 0.0	The minimum distance between points

Fortsetzung auf der nächsten Seite

Tab. 24.520 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Random points	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	The output random points. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Random points	`OUTPUT`	[Vektor: Punkt]	The output random points layer.

Pythoncode

Algorithm ID: `qgis:randompointsinlayerbounds`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Random points in polygons

Creates a point layer with points placed inside the polygons of another layer.

For each feature (polygon / multi-polygon) geometry in the input layer, the given number of points is added to the result layer.

Per feature and global minimum distances can be specified in order to avoid points being too close in the output point layer. If a minimum distance is specified, it may not be possible to generate the specified number of points for each feature. The total number of generated points and missed points are available as output from the algorithm.

The illustration below shows the effect of per feature and global minimum distances and zero/non-zero minimum distances (generated with the same seed, so at least the first point generated will be the same).

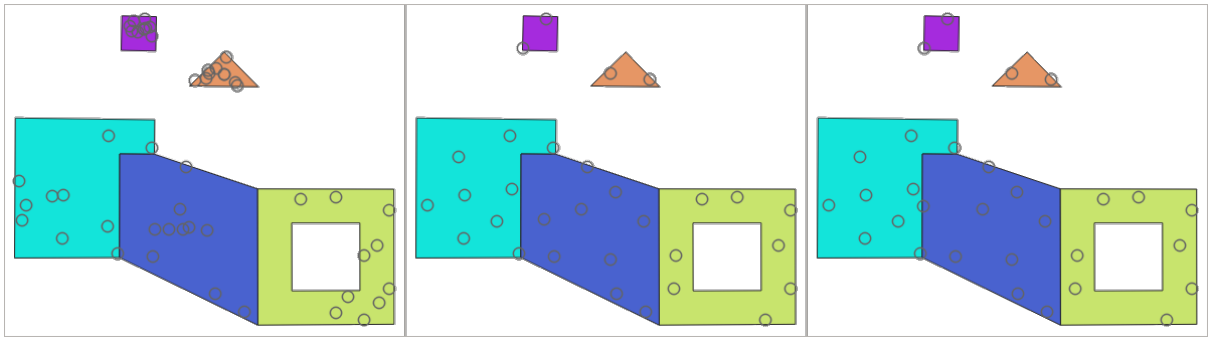


Abb. 24.83: Ten points per polygon feature, *left*: min. distances = 0, *middle*: min.distances = 1, *right*: min. distance = 1, global min. distance = 0

The maximum number of tries per point can be specified. This is only relevant for non-zero minimum distance.

A seed for the random number generator can be provided, making it possible to get identical random number sequences for different runs of the algorithm.

The attributes of the polygon feature on which a point was generated can be included (*Include polygon attributes*).

If you want approximately the same point density for all the features, you can data-define the number of points using the area of the polygon feature geometry.

Siehe auch

Random points inside polygons

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input polygon layer	`INPUT`	[vector: line]	Input polygon vector layer
Number of points for each feature	`POINTS_NUMBER`	[numeric: integer] Standard: 1	Number of points to create
Minimum distance between points Optional	`MIN_DISTANCE`	[numeric: double] Default: 0.0	The minimum distance between points within one polygon feature
Random points in polygons	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	The output random points. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Global minimum distance between points Optional	MIN_DISTANCE_G	[numeric: double] Default: 0.0	The global minimum distance between points. Should be smaller than the *Minimum distance between points (per feature)* for that parameter to have an effect.
Maximum number of search attempts (for Min. dist. > 0) Optional	MAX_TRIES_PER_	[numeric: integer] Default: 10	The maximum number of tries per point. Only relevant if the minimum distance between points is set (and greater than 0).
Random seed Optional	SEED	[numerisch: Ganzzahl] Standard: Nicht gesetzt	The seed to use for the random number generator.
Include polygon attributes	INCLUDE_POLYGO	[Boolean] Standard: True	If set, a point will get the attributes from the line on which it is placed.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Random points in polygons	OUTPUT	[Vektor: Punkt]	The output random points layer.
Number of features with empty or no geometry	FEATURES_WITH_	[numerisch: Ganzzahl]	
Total number of points generated	OUTPUT_POINTS	[numerisch: Ganzzahl]	
Number of missed points	POINTS_MISSED	[numerisch: Ganzzahl]	The number of points that could not be generated due to the minimum distance constraint.
Number of features with missed points	POLYGONS_WITH_	[numerisch: Ganzzahl]	Not including features with empty or no geometry

Pythoncode

Algorithm ID: `native:randompointsinpolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Random points inside polygons

Creates a new point layer with a given number of random points inside each polygon of the input polygon layer.

Two sampling strategies are available:

- Points count: number of points for each feature
- Points density: density of points for each feature

A minimum distance can be specified, to avoid points being too close to each other.

Default menu: *Vector* ► *Research Tools*

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Random points in polygons

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Input polygon vector layer
Sampling strategy	`STRATEGY`	[Aufzählung] Standard: 0	Sampling strategy to use. One of: • 0 — Points count: number of points for each feature • 1 — Points density: density of points for each feature
Point count or density	`VALUE`	[numeric: double] Standard: 1,0	The number or density of points, depending on the chosen *Sampling strategy*.
Minimum distance between points	`MIN_DISTANCE`	[numerisch: Double] Default: 0.0	The minimum distance between points
Random points	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	The output random points. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Random points	`OUTPUT`	[Vektor: Punkt]	The output random points layer.

Pythoncode

Algorithm ID: `qgis:randompointsinsidepolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Random points on lines

Creates a point layer with points placed on the lines of another layer.

For each feature (line / multi-line) geometry in the input layer, the given number of points is added to the result layer.

Per feature and global minimum distances can be specified in order to avoid points being too close in the output point layer. If a minimum distance is specified, it may not be possible to generate the specified number of points for each feature. The total number of generated points and missed points are available as output from the algorithm.

The illustration below shows the effect of per feature and global minimum distances and zero/non-zero minimum distances (generated with the same seed, so at least the first point generated will be the same).

Abb. 24.84: Five points per line feature, *left*: min. distances = 0, *middle*: min.distances != 0, *right*: min. distance != 0, global min. distance = 0

The maximum number of tries per point can be specified. This is only relevant for non-zero minimum distance.

A seed for the random number generator can be provided, making it possible to get identical random number sequences for different runs of the algorithm.

The attributes of the line feature on which a point was generated can be included (*Include line attributes*).

If you want approximately the same point density for all the line features, you can data-define the number of points using the length of the line feature geometry.

Siehe auch

Random points along line

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input line layer	`INPUT`	[vector: line]	Input line vector layer
Number of points for each feature	`POINTS_NUMBER`	[numeric: integer] Standard: 1	Number of points to create

Fortsetzung auf der nächsten Seite

Tab. 24.527 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Minimum distance between points (per feature) Optional	MIN_DISTANCE	[numeric: double] Default: 0.0	The minimum distance between points within one line feature
Random points on lines	OUTPUT	[Vektor: Punkt] Standard: [Create temporary layer]	The output random points. *One of*: • Temporäre Ebene erstellen (TEMPORARY_OUTPUT) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Global minimum distance between points Optional	MIN_DISTANCE_G	[numeric: double] Default: 0.0	The global minimum distance between points. Should be smaller than the *Minimum distance between points (per feature)* for that parameter to have an effect.
Maximum number of search attempts (for Min. dist. > 0) Optional	MAX_TRIES_PER_	[numeric: integer] Default: 10	The maximum number of tries per point. Only relevant if the minimum distance between points is set (and greater than 0).
Random seed Optional	SEED	[numerisch: Ganzzahl] Standard: Nicht gesetzt	The seed to use for the random number generator.
Include line attributes	INCLUDE_LINE_A	[Boolean] Standard: True	If set, a point will get the attributes from the line on which it is placed.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Random points on lines	OUTPUT	[Vektor: Punkt]	The output random points layer.
Number of features with empty or no geometry	FEATURES_WITH_	[numerisch: Ganzzahl]	
Number of features with missed points	LINES_WITH_MIS	[numerisch: Ganzzahl]	Not including features with empty or no geometry
Total number of points generated	POINTS_GENERAT	[numerisch: Ganzzahl]	

Fortsetzung auf der nächsten Seite

Tab. 24.529 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Number of missed points	`POINTS_MISSED`	[numerisch: Ganzzahl]	The number of points that could not be generated due to the minimum distance constraint.

Pythoncode

Algorithm ID: `native:randompointsonlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster pixels to points

Creates a vector layer of points corresponding to each pixel in a raster layer.

Converts a raster layer to a vector layer, by creating point features for each individual pixel's center in the raster layer. Any NoData pixels are skipped in the output.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Raster layer	`INPUT_RASTER`	[Raster]	Input raster layer
Band number	`RASTER_BAND`	[raster band]	Raster band to extract data from
Field name	`FIELD_NAME`	[Text] Default: ‚VALUE‘	Name of the field to store the raster band value
Vector points	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the resulting point layer of pixels centroids. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Vector points	`OUTPUT`	[Vektor: Punkt]	Resulting point layer with pixels centroids

Pythoncode

Algorithm ID: `native:pixelstopoints`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster pixels to polygons

Creates a vector layer of polygons corresponding to each pixel in a raster layer.

Converts a raster layer to a vector layer, by creating polygon features for each individual pixel's extent in the raster layer. Any NoData pixels are skipped in the output.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Raster layer	`INPUT_RASTER`	[Raster]	Input raster layer
Band number	`RASTER_BAND`	[raster band]	Raster band to extract data from
Field name	`FIELD_NAME`	[Text] Default: ‚VALUE‘	Name of the field to store the raster band value
Vector polygons	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the resulting polygon layer of pixel extents. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Vector polygons	`OUTPUT`	[Vektor: Polygon]	Resulting polygon layer of pixel extents

Pythoncode

Algorithm ID: `native:pixelstopolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Regular points

Creates a new point layer with its points placed in a regular grid within a given extent.

The grid is specified either by the spacing between the points (same spacing for all dimensions) or by the number of points to generate. In the latter case, the spacing will be determined from the extent. In order to generate a full rectangular grid, at least the number of points specified by the user is generated for the latter case.

Random offsets to the point spacing can be applied, resulting in a non-regular point pattern.

Default menu: *Vector* ► *Research Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input extent (xmin, xmax, ymin, ymax)	`EXTENT`	[Ausdehnung]	Map extent for the random points Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Point spacing/count	`SPACING`	[numerisch: Ganzzahl] Default: 100	Spacing between the points, or the number of points, depending on whether `Use point spacing` is checked or not.
Initial inset from corner (LH side)	`INSET`	[numerisch: Double] Default: 0.0	Offsets the points relative to the upper left corner. The value is used for both the X and Y axis.
Apply random offset to point spacing	`RANDOMIZE`	[Boolean] Standard: falsch	If checked the points will have a random spacing
Use point spacing	`IS_SPACING`	[Boolean] Standard: True	If unchecked the point spacing is not taken into account
Output layer CRS	`CRS`	[KBS] Default: *Project CRS*	CRS of the random points layer
Regular points	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output regular point layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Regular points	`OUTPUT`	[Vektor: Punkt]	The output regular point layer.

Pythoncode

Algorithm ID: `qgis:regularpoints`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.25 Vector general

Projektion zuweisen

Weist einem Vektorlayer eine neue Projektion zu.

Erzeugt einen neuen Layer mit exakt den gleichen Objekten und Geometrien wie die Eingabe, aber mit einem neu zugewiesenen KBS. Die Geometrien werden **nicht** umprojiziert, sondern es wird nur ein neues KBS zugewiesen.

Der Algorithmus kann benutzt werden, um Layer zu reparieren, denen eine falsche Projektion zugewiesen wurde.

Attribute werden von diesem Algorithmus nicht verändert.

Siehe auch

Define projection, *Find projection*, *Reproject layer*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Vector layer with wrong or missing CRS
Assigned CRS	`CRS`	[KBS] Default: `EPSG:4326 - WGS84`	Select the new CRS to assign to the vector layer
Assigned CRS Optional	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Assigned CRS	`OUTPUT`	[wie Eingabe]	Vector layer with assigned projection

Python-Code

Algorithm ID: `native:assignprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Batch Nominatim geocoder

Performs batch geocoding using the Nominatim service against an input layer string field. The output layer will have a point geometry reflecting the geocoded location as well as a number of attributes associated to the geocoded location.

Erlaubt auch die *Insitu-Bearbeitung* von Punkt-Objekten

Bemerkung

This algorithm is compliant with the usage policy of the Nominatim geocoding service provided by the OpenStreetMap Foundation.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Vector layer to geocode the features
Address field	`FIELD`	[tablefield: string]	Field containing the addresses to geocode
Geocoded	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output layer containing only the geocoded addresses. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Geocoded	`OUTPUT`	[Vektor: Punkt]	Vector layer with point features corresponding to the geocoded addresses

Python-Code

Algorithm ID: `native:batchnominatimgeocoder`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Convert layer to spatial bookmarks

Creates spatial bookmarks corresponding to the extent of features contained in a layer.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[vector: line, polygon]	The input vector layer
Bookmark destination	`DESTINATION`	[Aufzählung] Standard: 0	Select the destination for the bookmarks. One of: • 0 — Project bookmarks • 1 — User bookmarks
Name field	`NAME_EXPRESSIO`	[Ausdruck]	Field or expression that will give names to the generated bookmarks
Group field	`GROUP_EXPRESSI`	[Ausdruck]	Field or expression that will provide groups for the generated bookmarks

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Count of bookmarks added	`COUNT`	[numerisch: Ganzzahl]	

Python-Code

Algorithm ID: `native:layertobookmarks`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Convert spatial bookmarks to layer

Creates a new layer containing polygon features for stored spatial bookmarks. The export can be filtered to only bookmarks belonging to the current project, to all user bookmarks, or a combination of both.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Bookmark source	`SOURCE`	[Aufzählung] [Liste] Default: [0,1]	Select the source(s) of the bookmarks. One or more of: • 0 — Project bookmarks • 1 — User bookmarks
Output CRS	`CRS`	[KBS] Default: `EPSG:4326 - WGS 84`	The CRS of the output layer
Ergebnis	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ergebnis	`OUTPUT`	[Vektor: Polygon]	The output (bookmarks) vector layer

Python-Code

Algorithm ID: `native:bookmarkstolayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create attribute index

Creates an index against a field of the attribute table to speed up queries. The support for index creation depends on both the layer's data provider and the field type.

No outputs are created: the index is stored on the layer itself.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: alle]	Select the vector layer you want to create an attribute index for
Attribute to index	`FIELD`	[tablefield: any]	Field of the vector layer

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Indexed layer	`OUTPUT`	[wie Eingabe]	A copy of the input vector layer with an index for the specified field

Python-Code

Algorithm ID: `native:createattributeindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create spatial index

Creates an index to speed up access to the features in a layer based on their spatial location. Support for spatial index creation is dependent on the layer's data provider.

No new output layers are created.

Default menu: *Vector* ► *Data Management Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Indexed layer	`OUTPUT`	[wie Eingabe]	A copy of the input vector layer with a spatial index

Python-Code

Algorithm ID: `native:createspatialindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Define projection

Sets an existing layer's projection to the provided CRS without reprojecting features.

Contrary to the *Projektion zuweisen* algorithm, it modifies the current layer and will not output a new layer.

Bemerkung

For Shapefile datasets, the `.prj` and `.qpj` files will be overwritten - or created if missing - to match the provided CRS.

Default menu: *Vector* ► *Data Management Tools*

Siehe auch

Projektion zuweisen, *Find projection*, *Reproject layer*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Vector layer with missing projection information
CRS	`CRS`	[KBS]	Select the CRS to assign to the vector layer

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
	`INPUT`	[wie Eingabe]	The input vector layer with the defined projection

Python-Code

Algorithm ID: `native:definecurrentprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Delete duplicate geometries

Finds and removes duplicated geometries.

Attributes are not checked, so in case two features have identical geometries but different attributes, only one of them will be added to the result layer.

Bemerkung

This algorithm does not require valid geometries as input.

Siehe auch

Drop geometries, *Remove null geometries*, *Delete duplicates by attribute*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	The layer with duplicate geometries you want to clean
Cleaned	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output layer containing the unique features. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Count of discarded duplicate records	`DUPLICATE_COUN`	[numerisch: Ganzzahl]	Count of discarded duplicate records
Cleaned	`OUTPUT`	[wie Eingabe]	The output layer without any duplicated geometries
Count of retained records	`RETAINED_COUNT`	[numerisch: Ganzzahl]	Count of unique records

Python-Code

Algorithm ID: `native:deleteduplicategeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Delete duplicates by attribute

Deletes duplicate rows by only considering the specified field(s). The first matching row will be retained, and duplicates will be discarded.

Optionally, these duplicate records can be saved to a separate output for analysis.

Siehe auch

Delete duplicate geometries

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	The input layer
Fields to match duplicates by	`FIELDS`	[tablefield: any] [list]	Fields defining duplicates. Features with identical values for all these fields are considered duplicates.
Filtered (no duplicates)	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output layer containing the unique features. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Filtered (duplicates) Optional	`DUPLICATES`	[wie Eingabe] Standard: `[Ausgabe überspringen]`	Specify the output layer containing only the duplicates. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Filtered (duplicates)	`DUPLICATES`	[wie Eingabe]	Vector layer containing the duplicate features.
Count of discarded duplicate records	`DUPLICATE_COUN`	[numerisch: Ganzzahl]	Count of discarded duplicate records
Filtered (no duplicates)	`OUTPUT`	[wie Eingabe]	Vector layer containing the unique features.
Count of retained records	`RETAINED_COUNT`	[numerisch: Ganzzahl]	Count of unique records

Python-Code

Algorithm ID: `native:removeduplicatesbyattribute`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Detect dataset changes

Compares two vector layers, and determines which features are unchanged, added or deleted between the two. It is designed for comparing two different versions of the same dataset.

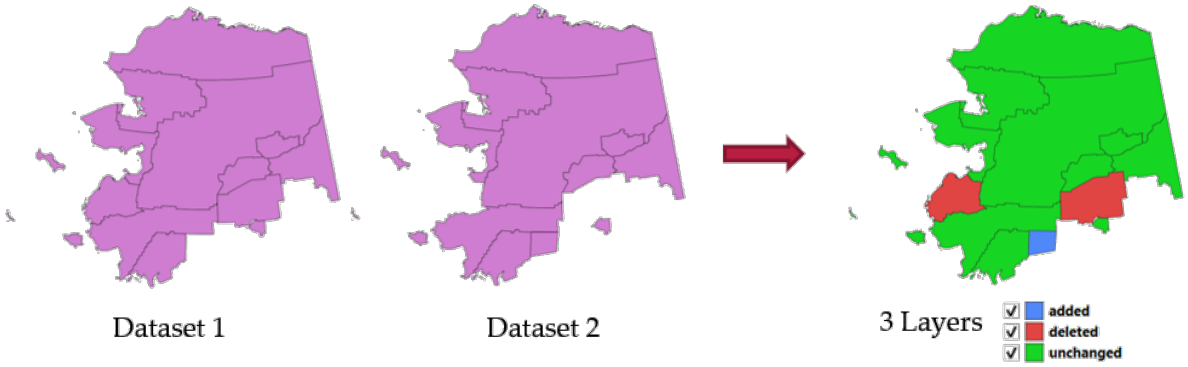


Abb. 24.85: Detect dataset change example

Parameter

Bezeichnung	Name	Typ	Beschreibung
Original layer	`ORIGINAL`	[Vektor: Geometrie]	The vector layer considered as the original version
Revised layer	`REVISED`	[Vektor: Geometrie]	The revised or modified vector layer
Attributes to consider for match Optional	`COMPARE_ATTRIB`	[tablefield: any] [list]	Attributes to consider for match. By default, all attributes are compared.
Geometry comparison behavior Optional	`MATCH_TYPE`	[Aufzählung] Standard: 1	Defines the criteria for comparison. Options: • 0 — Exact Match: includes the order and vertices count of geometries • 1 — Tolerant Match (Topological Equality): geometries are considered equal
Unchanged features Optional	`UNCHANGED`	[vector: same as Original layer]	Specify the output vector layer containing the unchanged features. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortsetzung auf der nächsten Seite

Tab. 24.554 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Added features Optional	ADDED	[vector: same as Original layer]	Specify the output vector layer containing the added features. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (TEMPORARY_OUTPUT) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Deleted features Optional	DELETED	[vector: same as Original layer]	Specify the output vector layer containing the deleted features. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (TEMPORARY_OUTPUT) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Unchanged features	UNCHANGED	[vector: same as Original layer]	Vector layer containing the unchanged features.
Added features	ADDED	[vector: same as Original layer]	Vector layer containing the added features.
Deleted features	DELETED	[vector: same as Original layer]	Vector layer containing the deleted features.
Count of unchanged features	UNCHANGED_COUN	[numerisch: Ganzzahl]	Count of unchanged features.
Count of features added in revised layer	ADDED_COUNT	[numerisch: Ganzzahl]	Count of features added in revised layer.
Count of features deleted from original layer	DELETED_COUNT	[numerisch: Ganzzahl]	Count of features deleted from original layer.

Python-Code

Algorithm ID: `native:detectvectorchanges`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Drop geometries

Creates a simple *geometryless* copy of the input layer attribute table. It keeps the attribute table of the source layer.

If the file is saved in a local folder, you can choose between many file formats.

Allows *features in-place modification* of point, line, and polygon features

Siehe auch

Delete duplicate geometries, *Remove null geometries*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	The input vector layer
Dropped geometries	`OUTPUT`	[vector: table]	Specify the output geometryless layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Dropped geometries	`OUTPUT`	[vector: table]	The output geometryless layer. A copy of the original attribute table.

Python-Code

Algorithm ID: `native:dropgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Execute SQL

Runs a simple or complex query based only on SELECT with `SQL` syntax on the source layer.

Input datasources are identified with `input1`, `input2`... `inputN` and a simple query will look like `SELECT * FROM input1`.

Beside a simple query, you can add expressions or variables within the `SQL query` parameter itself. This is particulary useful if this algorithm is executed within a Processing model and you want to use a model input as a parameter of the query. An example of a query will then be `SELECT * FROM [% @table %]` where `@table` is the variable that identifies the model input.

The result of the query will be added as a new layer.

Siehe auch

SpatiaLite execute SQL, *PostgreSQL execute SQL*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional input datasources (called input1, .., inputN in the query)	`INPUT_DATASOUR`	[vector: any] [list]	List of layers to query. In the SQL editor you can refer these layers with their **real** name or also with **input1**, **input2**, **inputN** depending on how many layers have been chosen.
SQL query	`INPUT_QUERY`	[Text]	Type the string of your SQL query, e.g. `SELECT * FROM input1`.
Unique identifier field Optional	`INPUT_UID_FIEL`	[Text]	Specify the column with unique ID
Geometry field Optional	`INPUT_GEOMETRY`	[Text]	Specify the geometry field
Geometry type Optional	`INPUT_GEOMETRY`	[Aufzählung] Standard: 0	Choose the geometry of the result. By default the algorithm will autodetect it. One of: • 0 — Autodetect • 1 — No geometry • 2 — Point • 3 — LineString • 4 — Polygon • 5 — MultiPoint • 6 — MultiLineString • 7 — MultiPolygon
CRS Optional	`INPUT_GEOMETRY`	[KBS]	The CRS to assign to the output layer

Fortsetzung auf der nächsten Seite

Tab. 24.558 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
SQL Output	`OUTPUT`	[Vektor: Geometrie] Standard: `[Create temporary layer]`	Specify the output layer created by the query. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
SQL Output	`OUTPUT`	[Vektor: Geometrie]	Vector layer created by the query

Python-Code

Algorithm ID: `qgis:executesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export layers to DXF

Exports layers to DXF file. For each layer, you can choose a field whose values are used to split features in generated destination layers in DXF output.

Siehe auch

Erstellen neuer DXF-Dateien

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	LAYERS	[vector: geometry] [list]	List of input vector layers with options associated (filled as a QgsProcessingParameterDxfLayers item — done in GUI by pressing *Configure Layer…* button for each selected layer): **Layer [string] (layer)** Full path of the input layer to export **Output layer attribute [tablefield: any] (attributeIndex)** Attribute index to split the input layer using unique values **Output layer name [string] (overriddenLayerName)** Overridden layer name to be used in the exported DXF file (the original project layer remains untouched) **Allow data defined symbol blocks** [boolean] (buildDataDefinedBlocks) **Maximum number of symbol blocks [numeric: integer] (dataDefinedBlocksMaximumNumberOfClasses)** −1 means no limitation.
Symbology mode	SYMBOLOGY_MODE	[Aufzählung] Standard: 0	Type of symbology to apply to output layers. You can choose between: • 0 — No Symbology • 1 — Feature Symbology • 2 — Symbol Layer Symbology
Symbology scale	SYMBOLOGY_SCAL	[scale] Default: 1:1 000 000	Default scale of data export.
Map theme Optional	MAP_THEME	[map theme]	Match layer styling to the provided map theme.
Encoding	ENCODING	[Aufzählung]	Encoding to apply to layers.
CRS	CRS	[KBS]	Choose the CRS for the output layer.
Extent Optional	EXTENT	[Ausdehnung]	Limit exported features to those with geometries intersecting the provided extent.
Use layer title as name	USE_LAYER_TITL	[Boolean] Standard: falsch	Name the output layer with the layer title (as set in layer metadata or QGIS Server properties) instead of the layer name.
Force 2D	FORCE_2D	[Boolean] Standard: falsch	
Export labels as MTEXT elements	MTEXT	[Boolean] Standard: True	Exports labels as MTEXT or TEXT elements
Use only selected features	SELECTED_FEATU	[Boolean] Standard: falsch	Exports only the selected features.
DXF	OUTPUT	[Datei] Standard: [In temporäre Datei speichern]	Specification of the output DXF file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
DXF	`OUTPUT`	[Datei]	`.DXF` file containing the input layers

Python-Code

Algorithm ID: `native:dxfexport`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract selected features

Saves the selected features as a new layer.

Bemerkung

If the selected layer has no selected features, the newly created layer will be empty.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: alle]	Layer to save the selection from
Selected features	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the vector layer for the selected features. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Selected features	`OUTPUT`	[wie Eingabe]	Vector layer with only the selected features, or no feature if none was selected.

Python-Code

Algorithm ID: `native:saveselectedfeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract Shapefile encoding

Extracts the attribute encoding information embedded in a Shapefile. Both the encoding specified by an optional `.cpg` file and any encoding details present in the `.dbf` LDID header block are considered.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: Geometrie]	ESRI Shapefile (`.SHP`) Layer to extract the encoding information.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Shapefile encoding	`ENCODING`	[Text]	Encoding information specified in the input file
CPG encoding	`CPG_ENCODING`	[Text]	Encoding information specified in any optional `.CPG` file
LDID encoding	`LDID_ENCODING`	[Text]	Encoding information specified in `.dbf` LDID header block

Python-Code

Algorithm ID: `native:shpencodinginfo`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Find projection

Creates a shortlist of candidate coordinate reference systems, for instance for a layer with an unknown projection.

The area that the layer is expected to cover must be specified via the target area parameter. The coordinate reference system for this target area must be known to QGIS.

The algorithm operates by testing the layer's extent in every known reference system and then listing any for which the bounds would be near the target area if the layer was in this projection.

Siehe auch

Projektion zuweisen, *Define projection*, *Reproject layer*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: Geometrie]	Layer with unknown projection
Target area for layer (xmin, xmax, ymin, ymax)	`TARGET_AREA`	[Ausdehnung]	The area that the layer covers. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Fortsetzung auf der nächsten Seite

Tab. 24.566 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
CRS candidates	`OUTPUT`	[vector: table] Standard: `[Create temporary layer]`	Specify the table (geometryless layer) for the CRS suggestions (EPSG codes). *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
CRS candidates	`OUTPUT`	[vector: table]	A table with all the CRS (EPSG codes) of the matching criteria.

Python-Code

Algorithm ID: `native:findprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Flatten relationship

Flattens a *relationship* for a vector layer, exporting a single layer containing one parent feature per related child feature. This master feature contains all the attributes for the related features. This allows to have the relation as a plain table that can be e.g. exported to CSV.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

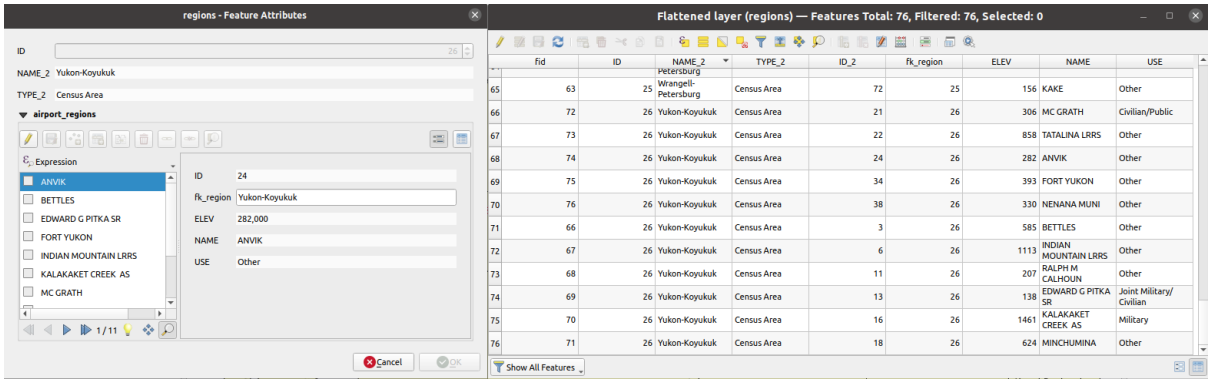


Abb. 24.86: Form of a region with related children (left) - A duplicate region feature for each related child, with joined attributes (right)

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: alle]	Layer with the relationship that should be de-normalized
Flattened Layer Optional	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output (flattened) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Flattened layer	`OUTPUT`	[wie Eingabe]	A layer containing master features with all the attributes for the related features

Python-Code

Algorithm ID: `native:flattenrelationships`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Join attributes by field value

Takes an input vector layer and creates a new vector layer that is an extended version of the input one, with additional attributes in its attribute table.

The additional attributes and their values are taken from a second vector layer. An attribute is selected in each of them to define the join criteria.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Join attributes by nearest, *Join attributes by location*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: alle]	Input vector layer. The output layer will consist of the features of this layer with attributes from matching features in the second layer.
Table field	`FIELD`	[tablefield: any]	Field of the source layer to use for the join
Input layer 2	`INPUT_2`	[Vektor: alle]	Layer with the attribute table to join
Table field 2	`FIELD_2`	[tablefield: any]	Field of the second (join) layer to use for the join The type of the field must be equal to (or compatible with) the input table field type.
Layer 2 fields to copy Optional	`FIELDS_TO_COPY`	[tablefield: any] [list]	Select the specific fields you want to add. By default all the fields are added.
Join type	`METHOD`	[Aufzählung] Standard: 1	The type of the final joined layer. One of: • 0 — Create separate feature for each matching feature (one-to-many) • 1 — Take attributes of the first matching feature only (one-to-one)
Discard records which could not be joined	`DISCARD_NONMAT`	[Boolean] Standard: True	Check if you don't want to keep the features that could not be joined
Joined field prefix Optional	`PREFIX`	[Text]	Add a prefix to joined fields in order to easily identify them and avoid field name collision

Fortsetzung auf der nächsten Seite

Tab. 24.570 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Joined layer Optional	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer for the join. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Unjoinable features from first layer Optional	`NON_MATCHING`	[wie Eingabe] Standard: `[Ausgabe überspringen]`	Specify the output vector layer for unjoinable features from first layer. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Number of joined features from input table	`JOINED_COUNT`	[numerisch: Ganzzahl]	
Unjoinable features from first layer Optional	`NON_MATCHING`	[wie Eingabe]	Vector layer with the non-matched features
Joined layer Optional	`OUTPUT`	[wie Eingabe]	Output vector layer with added attributes from the join
Number of unjoinable features from input table Optional	`UNJOINABLE_COU`	[numerisch: Ganzzahl]	

Python-Code

Algorithm ID: `native:joinattributestable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Join attributes by location

Takes an input vector layer and creates a new vector layer that is an extended version of the input one, with additional attributes in its attribute table.

The additional attributes and their values are taken from a second vector layer. A spatial criteria is applied to select the values from the second layer that are added to each feature from the first layer.

Default menu: *Vector ► Data Management Tools*

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Join attributes by nearest, *Join attributes by field value*, *Join attributes by location (summary)*

Exploring spatial relations

Geometrische Prädikate sind boolesche Funktionen, die verwendet werden, um die räumliche Beziehung eines Features zu einem anderen zu bestimmen, indem verglichen wird, ob und wie ihre Geometrien einen Teil des Raums teilen.

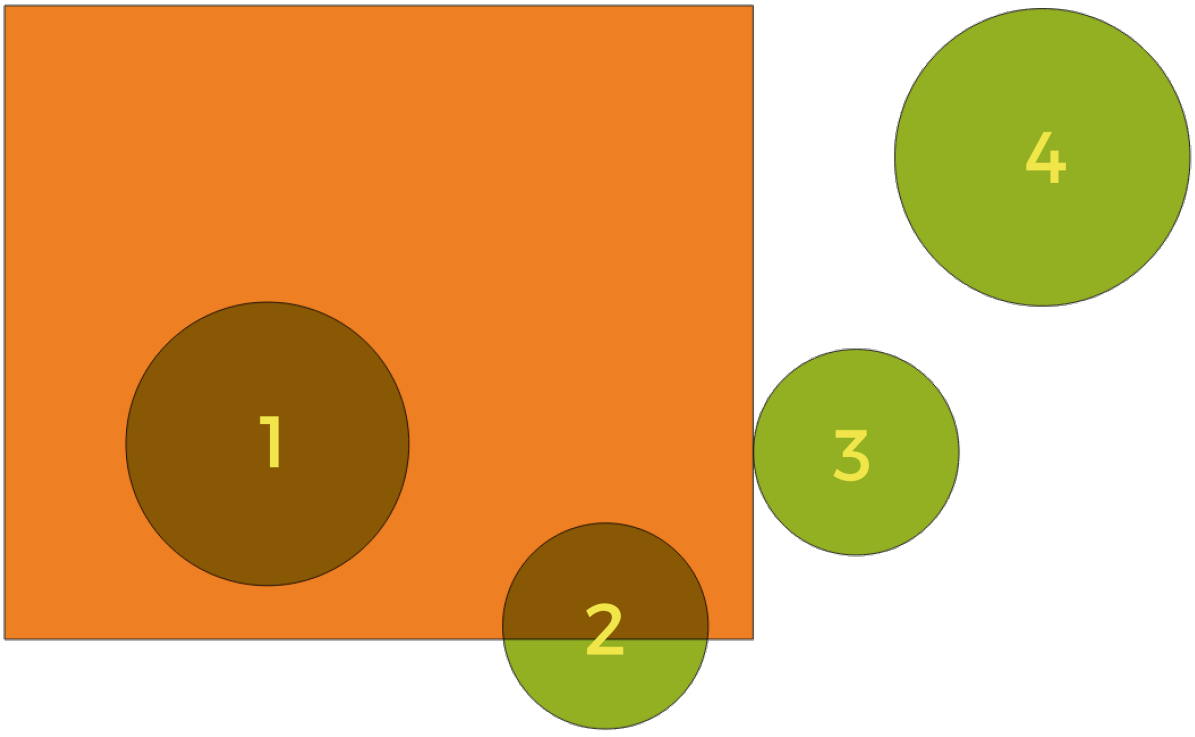


Abb. 24.87: Suche nach räumlichen Beziehungen zwischen Layern

Unter Verwendung der obigen Abbildung suchen wir nach den grünen Kreisen, indem wir sie räumlich mit dem orangefarbenen Rechteck-Feature vergleichen. Verfügbare geometrische Prädikate sind:

Schneidet
: Testet, ob eine Geometrie eine andere schneidet. Gibt 1 (wahr) zurück, wenn sich die Geometrien räumlich

überschneiden (einen Teil des Raums teilen – überlappen oder berühren) und 0, wenn dies nicht der Fall ist. Im obigen Bild werden die Kreise 1, 2 und 3 zurückgegeben.

Enthält
Gibt 1 (wahr) zurück, wenn und nur wenn keine Punkte von b außerhalb von a liegen und mindestens ein Punkt des Inneren von b innerhalb von a liegt. Im Bild wird kein Kreis zurückgegeben, sondern das Rechteck wäre es, wenn man es umgekehrt suchen würde, da es den Kreis 1 komplett enthält. Das ist das Gegenteil von *sind innerhalb*.

Getrennt
Gibt 1 (wahr) zurück, wenn die Geometrien keinen Teil des Raums teilen (keine Überlappung, keine Berührung). Nur Kreis 4 wird zurückgegeben.

Gleich
Gibt 1 (wahr) zurück, wenn und nur wenn die Geometrien genau gleich sind. Es werden keine Kreise zurückgegeben.

Berührt
Testet, ob eine Geometrie eine andere berührt. Gibt 1 (wahr) zurück, wenn die Geometrien mindestens einen Punkt gemeinsam haben, aber ihre Innenräume sich nicht schneiden. Nur Kreis 3 wird zurückgegeben.

Überlappt
Testet, ob eine Geometrie eine andere überlappt. Gibt 1 (wahr) zurück, wenn die Geometrien sich den Raum teilen, dieselbe Dimension haben, aber nicht vollständig ineinander liegen. Nur Kreis 2 wird zurückgegeben.

Sind innerhalb
Testet, ob sich eine Geometrie in einer anderen befindet. Gibt 1 (wahr) zurück, wenn sich Geometrie a vollständig innerhalb von Geometrie b befindet. Nur Kreis 1 wird zurückgegeben.

Kreuzen
Gibt 1 (wahr) zurück, wenn die bereitgestellten Geometrien einige, aber nicht alle inneren Punkte gemeinsam haben und die tatsächliche Kreuzung eine geringere Dimension als die höchste bereitgestellte Geometrie hat. Beispielsweise wird eine Linie, die ein Polygon kreuzt, als Linie gekreuzt (wahr). Zwei Linien, die sich kreuzen, kreuzen sich als Punkt (wahr). Zwei Polygone kreuzen sich als Polygon (falsch). Im Bild werden keine Kreise zurückgegeben.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Join to features in	`INPUT`	[Vektor: Geometrie]	Input vector layer. The output layer will consist of the features of this layer with attributes from matching features in the second layer.
Where the features	`PREDICATE`	[Aufzählung] [Liste] Default: [0]	Type of spatial relation the source feature should have with the target feature so that they could be joined. One or more of: • 0 — intersect • 1 — contain • 2 — equal • 3 — touch • 4 — overlap • 5 — are within • 6 — cross If more than one condition is chosen, at least one of them (OR operation) has to be met for a feature to be extracted.

Fortsetzung auf der nächsten Seite

Tab. 24.572 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
By comparing to	`JOIN`	[Vektor: Geometrie]	The join layer. Features of this vector layer will **add** their attributes to the source layer attribute table if they satisfy the spatial relationship.
Fields to add (leave empty to use all fields) Optional	`JOIN_FIELDS`	[tablefield: any] [list]	Select the specific fields you want to add from the join layer. By default all the fields are added.
Join type	`METHOD`	[Aufzählung]	The type of the final joined layer. One of: • 0 — Create separate feature for each matching feature (one-to-many) • 1 — Take attributes of the first matching feature only (one-to-one) • 2 — Take attributes of the feature with largest overlap only (one-to-one)
Discard records which could not be joined	`DISCARD_NONMAT`	[Boolean] Standard: falsch	Remove from the output the input layer's features which could not be joined
Joined field prefix Optional	`PREFIX`	[Text]	Add a prefix to joined fields in order to easily identify them and avoid field name collision
Joined layer Optional	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer for the join. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Unjoinable features from first layer Optional	`NON_MATCHING`	[wie Eingabe] Standard: `[Ausgabe überspringen]`	Specify the output vector layer for unjoinable features from first layer. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortsetzung auf der nächsten Seite

Tab. 24.573 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Number of joined features from input table	`JOINED_COUNT`	[numerisch: Ganzzahl]	
Unjoinable features from first layer Optional	`NON_MATCHING`	[wie Eingabe]	Vector layer of the non-matched features
Joined layer	`OUTPUT`	[wie Eingabe]	Output vector layer with added attributes from the join

Python-Code

Algorithm ID: `native:joinattributesbylocation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Join attributes by location (summary)

Takes an input vector layer and creates a new vector layer that is an extended version of the input one, with additional attributes in its attribute table.

The additional attributes and their values are taken from a second vector layer. A spatial criteria is applied to select the values from the second layer that are added to each feature from the first layer.

The algorithm calculates a statistical summary for the values from matching features in the second layer (e.g. maximum value, mean value, etc).

Siehe auch

Join attributes by location

Exploring spatial relations

Geometrische Prädikate sind boolesche Funktionen, die verwendet werden, um die räumliche Beziehung eines Features zu einem anderen zu bestimmen, indem verglichen wird, ob und wie ihre Geometrien einen Teil des Raums teilen.

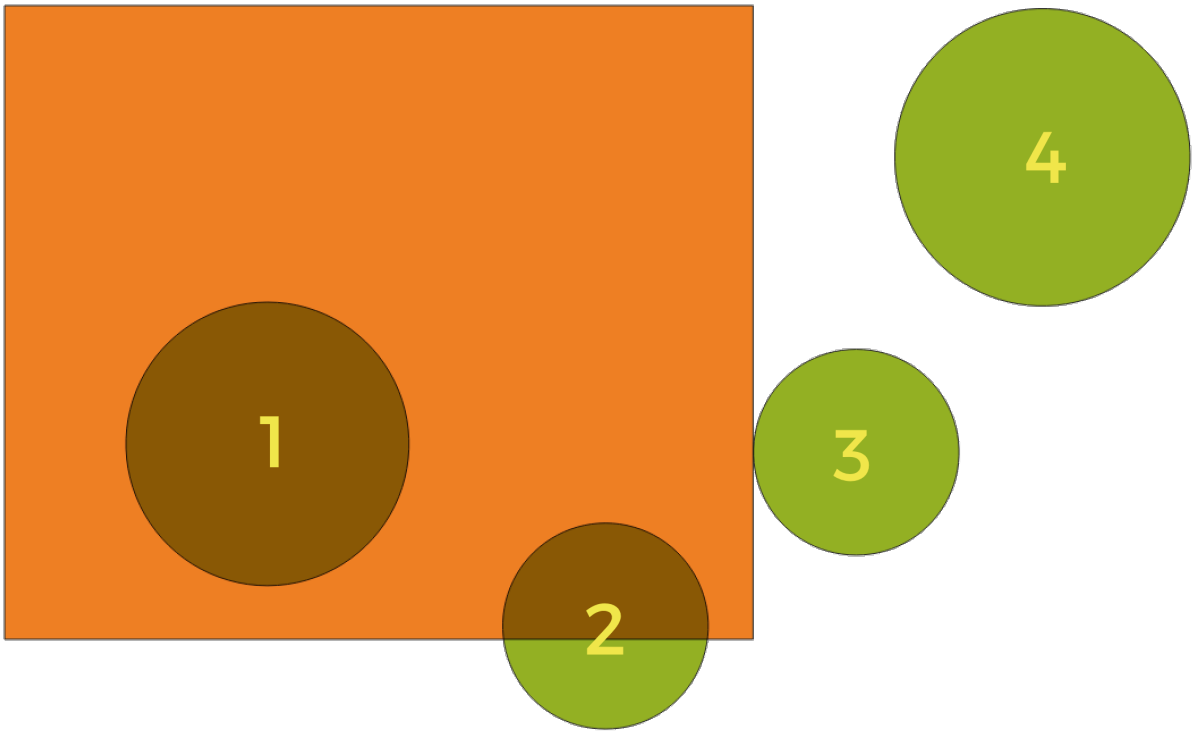


Abb. 24.88: Suche nach räumlichen Beziehungen zwischen Layern

Unter Verwendung der obigen Abbildung suchen wir nach den grünen Kreisen, indem wir sie räumlich mit dem orangefarbenen Rechteck-Feature vergleichen. Verfügbare geometrische Prädikate sind:

Schneidet
: Testet, ob eine Geometrie eine andere schneidet. Gibt 1 (wahr) zurück, wenn sich die Geometrien räumlich überschneiden (einen Teil des Raums teilen – überlappen oder berühren) und 0, wenn dies nicht der Fall ist. Im obigen Bild werden die Kreise 1, 2 und 3 zurückgegeben.

Enthält
: Gibt 1 (wahr) zurück, wenn und nur wenn keine Punkte von b außerhalb von a liegen und mindestens ein Punkt des Inneren von b innerhalb von a liegt. Im Bild wird kein Kreis zurückgegeben, sondern das Rechteck wäre es, wenn man es umgekehrt suchen würde, da es den Kreis 1 komplett enthält. Das ist das Gegenteil von *sind innerhalb*.

Getrennt
: Gibt 1 (wahr) zurück, wenn die Geometrien keinen Teil des Raums teilen (keine Überlappung, keine Berührung). Nur Kreis 4 wird zurückgegeben.

Gleich
: Gibt 1 (wahr) zurück, wenn und nur wenn die Geometrien genau gleich sind. Es werden keine Kreise zurückgegeben.

Berührt
: Testet, ob eine Geometrie eine andere berührt. Gibt 1 (wahr) zurück, wenn die Geometrien mindestens einen Punkt gemeinsam haben, aber ihre Innenräume sich nicht schneiden. Nur Kreis 3 wird zurückgegeben.

Überlappt
: Testet, ob eine Geometrie eine andere überlappt. Gibt 1 (wahr) zurück, wenn die Geometrien sich den Raum

teilen, dieselbe Dimension haben, aber nicht vollständig ineinander liegen. Nur Kreis 2 wird zurückgegeben.

Sind innerhalb
Testet, ob sich eine Geometrie in einer anderen befindet. Gibt 1 (wahr) zurück, wenn sich Geometrie a vollständig innerhalb von Geometrie b befindet. Nur Kreis 1 wird zurückgegeben.

Kreuzen
Gibt 1 (wahr) zurück, wenn die bereitgestellten Geometrien einige, aber nicht alle inneren Punkte gemeinsam haben und die tatsächliche Kreuzung eine geringere Dimension als die höchste bereitgestellte Geometrie hat. Beispielsweise wird eine Linie, die ein Polygon kreuzt, als Linie gekreuzt (wahr). Zwei Linien, die sich kreuzen, kreuzen sich als Punkt (wahr). Zwei Polygone kreuzen sich als Polygon (falsch). Im Bild werden keine Kreise zurückgegeben.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Join to features in	`INPUT`	[Vektor: Geometrie]	Input vector layer. The output layer will consist of the features of this layer with attributes from matching features in the second layer.
Where the features	`PREDICATE`	[Aufzählung] [Liste] Default: [0]	Type of spatial relation the source feature should have with the target feature so that they could be joined. One or more of: • 0 — intersect • 1 — contain • 2 — equal • 3 — touch • 4 — overlap • 5 — are within • 6 — cross If more than one condition is chosen, at least one of them (OR operation) has to be met for a feature to be extracted.
By comparing to	`JOIN`	[Vektor: Geometrie]	The join layer. Features of this vector layer will **add** summaries of their attributes to the source layer attribute table if they satisfy the spatial relationship.
Fields to summarize (leave empty to use all fields) Optional	`JOIN_FIELDS`	[tablefield: any] [list]	Select the specific fields you want to add from the join layer. By default all the fields are added.

Fortsetzung auf der nächsten Seite

Tab. 24.574 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Summaries to calculate (leave empty to use all fields) Optional	SUMMARIES	[Aufzählung] [Liste] Default: []	For each input feature, statistics are calculated on joined fields of their matching features. One or more of: • 0 — count • 1 — unique • 2 — min • 3 — max • 4 — range • 5 — sum • 6 — mean • 7 — median • 8 — stddev • 9 — minority • 10 — majority • 11 — q1 • 12 — q3 • 13 — iqr • 14 — empty • 15 — filled • 16 — min_length • 17 — max_length • 18 — mean_length
Discard records which could not be joined	DISCARD_NONMAT	[Boolean] Standard: falsch	Remove from the output the input layer's features which could not be joined
Joined layer	OUTPUT	[wie Eingabe] Standard: [Create temporary layer]	Specify the output vector layer for the join. *One of*: • Temporäre Ebene erstellen (TEMPORARY_OUTPUT) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Joined layer	OUTPUT	[wie Eingabe]	Output vector layer with summarized attributes from the join

Python-Code

Algorithm ID: `native:joinbylocationsummary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Join attributes by nearest

Takes an input vector layer and creates a new vector layer with additional fields in its attribute table. The additional attributes and their values are taken from a second vector layer. Features are joined by finding the closest features from each layer.

By default only the nearest feature is joined, but the join can also join to the k-nearest neighboring features.

If a maximum distance is specified, only features which are closer than this distance will be matched.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Nearest neighbour analysis, *Join attributes by field value*, *Join attributes by location*, *Distance matrix*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	The input layer.
Input layer 2	`INPUT_2`	[Vektor: Geometrie]	The join layer.
Layer 2 fields to copy (leave empty to copy all fields)	`FIELDS_TO_COPY`	[fields]	Join layer fields to copy (if empty, all fields will be copied).
Discard records which could not be joined	`DISCARD_NONMAT`	[Boolean] Standard: falsch	Remove from the output the input layer records which could not be joined
Joined field prefix	`PREFIX`	[Text]	Joined field prefix
Maximum nearest neighbors	`NEIGHBORS`	[numerisch: Ganzzahl] Standard: 1	Maximum number of nearest neighbors
Maximum distance	`MAX_DISTANCE`	[numerisch: Double]	Maximum search distance

Fortsetzung auf der nächsten Seite

Tab. 24.576 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Joined layer Optional	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the vector layer containing the joined features. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Unjoinable features from first layer	`NON_MATCHING`	[wie Eingabe] Standard: `[Ausgabe überspringen]`	Specify the vector layer containing the features that could not be joined. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Joined layer	`OUTPUT`	[wie Eingabe]	The output joined layer.
Unjoinable features from first layer	`NON_MATCHING`	[wie Eingabe]	Layer containing the features from first layer that could not be joined to any features in the join layer.
Number of joined features from input table	`JOINED_COUNT`	[numerisch: Ganzzahl]	Number of features from the input table that have been joined.
Number of unjoinable features from input table	`UNJOINABLE_COU`	[numerisch: Ganzzahl]	Number of features from the input table that could not be joined.

Python-Code

Algorithm ID: `native:joinbynearest`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

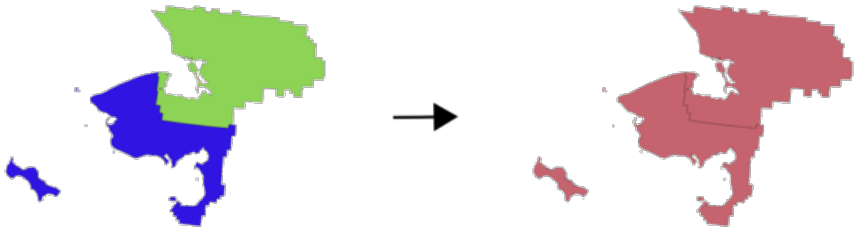
Merge vector layers

Combines multiple vector layers of the **same geometry** type into a single one.

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.

If any input layers contain Z or M values, then the output layer will also contain these values. Similarly, if any of the input layers are multi-part, the output layer will also be a multi-part layer.

Optionally, the destination coordinate reference system (CRS) for the merged layer can be set. If it is not set, the CRS will be taken from the first input layer. All layers will be reprojected to match this CRS.



Default menu: *Vector ► Data Management Tools*

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Split vector layer

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layers	`LAYERS`	[vector: any] [list]	The layers that are to be merged into a single layer. Layers should be of the same geometry type.
Destination CRS Optional	`CRS`	[KBS]	Choose the CRS for the output layer. If not specified, the CRS of the first input layer is used.
Add source layer information (layer name and path)	`ADD_SOURCE_FIE`	[Boolean] Standard: True	Add fields storing the original layer name and path
Zusammengeführt	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Zusammengeführt	`OUTPUT`	[wie Eingabe]	Output vector layer containing all the features and attributes from the input layers.

Python-Code

Algorithm ID: `native:mergevectorlayers`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Order by expression

Sorts a vector layer according to an expression: changes the feature index according to an expression.

Be careful, it might not work as expected with some providers, the order might not be kept every time.

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: alle]	Input vector layer to sort
Expression	`EXPRESSION`	[Ausdruck]	Expression to use for the sorting
Sort ascending	`ASCENDING`	[Boolean] Standard: True	If checked the vector layer will be sorted from small to large values.
Sort nulls first	`NULLS_FIRST`	[Boolean] Standard: falsch	If checked, Null values are placed first
Ordered	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ordered	`OUTPUT`	[wie Eingabe]	Output (sorted) vector layer

Python-Code

Algorithm ID: `native:orderbyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Repair Shapefile

Repairs a broken ESRI Shapefile dataset by (re)creating the SHX file.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Shapefile	`INPUT`	[Datei]	Full path to the ESRI Shapefile dataset with a missing or broken SHX file

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Repaired layer	`OUTPUT`	[Vektor: Geometrie]	The input vector layer with the SHX file repaired

Python-Code

Algorithm ID: `native:repairshapefile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Reproject layer

Reprojects a vector layer in a different CRS. The reprojected layer will have the same features and attributes of the input layer.

Allows *features in-place modification* of point, line, and polygon features

Siehe auch

Projektion zuweisen, *Define projection*, *Find projection*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: Geometrie]	Input vector layer to reproject
Ziel KBS	`TARGET_CRS`	[KBS] Default: `EPSG:4326 - WGS 84`	Destination coordinate reference system
Convert curved geometries to straight segments	`CONVERT_CURVED`	[Boolean] Standard: falsch	If checked, curved geometries will be converted to straight segments in the process, avoiding potential distortion issues.
Reprojiziert	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Coordinate Operation Optional	`OPERATION`	[Text]	Specific operation to use for a particular reprojection task, instead of always forcing use of the current project's transformation settings. Useful when reprojecting a particular layer and control over the exact transformation pipeline is required. Requires proj version >= 6. Read more at *Datumstransformationen*.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Reprojiziert	`OUTPUT`	[wie Eingabe]	Output (reprojected) vector layer

Python-Code

Algorithm ID: `native:reprojectlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Save vector features to file

Saves vector features to a specified file dataset.

For dataset formats supporting layers, an optional layer name parameter can be used to specify a custom string. Optional GDAL-defined dataset and layer options can be specified. For more information on this, read the online GDAL documentation on the format.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Vector features	`INPUT`	[Vektor: alle]	Input vector layer.
Saved features	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specify the file to save the features to. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Layer name Optional	`LAYER_NAME`	[Text]	Name to use for the output layer
GDAL dataset options Optional	`DATASOURCE_OPT`	[Text]	GDAL dataset creation options of the output format. Separate individual options with semicolons.
GDAL layer options Optional	`LAYER_OPTIONS`	[Text]	GDAL layer creation options of the output format. Separate individual options with semicolons.

Fortsetzung auf der nächsten Seite

Tab. 24.588 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Action to take on pre-existing file	ACTION_ON_EXIS	[Aufzählung] Standard: 0	How to manage existing features. Valid methods are: • 0 — Create or overwrite file • 1 — Create or overwrite layer • 2 — Append features to existing layer, but do not create new fields • 3 — Append features to existing layer, and create new fields if needed

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Saved features	OUTPUT	[wie Eingabe]	Vector layer with the saved features.
File name and path	FILE_PATH	[Text]	Output file name and path.
Layer name	LAYER_NAME	[Text]	Name of the layer, if any.

Python-Code

Algorithm ID: `native:savefeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Set layer encoding

Sets the encoding used for reading a layer's attributes. No permanent changes are made to the layer, rather it affects only how the layer is read during the current session.

Bemerkung

Changing the encoding is only supported for some vector layer data sources.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Saved features	`INPUT`	[Vektor: Geometrie]	Vector layer to set the encoding.
Encoding	`ENCODING`	[Text]	Text encoding to assign to the layer in the current QGIS session.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabelayer	`OUTPUT`	[wie Eingabe]	Input vector layer with the set encoding.

Python-Code

Algorithm ID: `native:setlayerencoding`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Split features by character

Features are split into multiple output features by splitting a field's value at a specified character. For instance, if a layer contains features with multiple comma separated values contained in a single field, this algorithm can be used to split these values up across multiple output features. Geometries and other attributes remain unchanged in the output. Optionally, the separator string can be a regular expression for added flexibility.

Allows *features in-place modification* of point, line, and polygon features

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer
Split using values in the field	`FIELD`	[tablefield: any]	Field to use for splitting
Split value using character	`CHAR`	[Text]	Character to use for splitting

Fortsetzung auf der nächsten Seite

Tab. 24.592 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Use regular expression separator	`REGEX`	[Boolean] Standard: falsch	
Split	`OUTPUT`	[wie Eingabe] Default: `Create temporary layer`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Split	`OUTPUT`	[wie Eingabe]	The output vector layer.

Python-Code

Algorithm ID: `native:splitfeaturesbycharacter`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Split vector layer

Creates a set of vectors in an output folder based on an input layer and an attribute. The output folder will contain as many layers as the unique values found in the desired field.

The number of files generated is equal to the number of different values found for the specified attribute.

It is the opposite operation of *merging*.

Default menu: *Vector ► Data Management Tools*

Siehe auch

Merge vector layers

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer
Unique ID field	`FIELD`	[tablefield: any]	Field to use for splitting
Ausgabeverzeichnis	`OUTPUT`	[Verzeichnis] Standard: `[In temporärem Verzeichnis speichern]`	Specify the directory for the output layers. *One of*: • In einem temporären Verzeichnis speichern • In Verzeichnis speichern

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Output file type Optional	`FILE_TYPE`	[Aufzählung] Default: `gpkg` in the dialog window	Select the extension of the output files. If not specified or invalid, the output files format will be the one set in the „Default output vector layer extension“ Processing setting.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabeverzeichnis	`OUTPUT`	[Verzeichnis]	The directory for the output layers
Output layers	`OUTPUT_LAYERS`	[same as input] [list]	The output vector layers resulting from the split.

Python-Code

Algorithm ID: `native:splitvectorlayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Truncate table

Truncates a layer, by deleting all features from within the layer.

> **Warnung**
>
> This algorithm modifies the layer in place, and deleted features cannot be restored!

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Layer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Truncated layer	`OUTPUT`	[Verzeichnis]	The input layer, all features deleted

Python-Code

Algorithm ID: `native:truncatetable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.26 Vektor-Geometrie

Geometrieattribute hinzufügen

Berechnet die geometrischen Eigenschaften der Objekte in einem Vektorlayer und nimmt sie in das Outputlayer auf.

Erzeugt eine neue Vektorebene mit demselben Inhalt wie das Eingabelayer, aber mit zusätzlichen Attributen, die geometrische Messungen auf der Grundlage eines ausgewählten KBS enthalten.

Die der Tabelle hinzugefügten Attribute hängen von der Geometrieart und der Dimension des Eingabelayers ab:

- f für **Punkt**-Layer: X (`xcoord`), Y (`ycoord`), Z (`zcoord`) Koordinaten und/oder M-Wert (`mvalue`)
- for **line** layers: `length` and, for the LineString and CompoundCurve geometry types, the feature `sinuosity` and straight distance (`straightdis`)
- für **Polygon**- Layer: `perimeter` und `area`

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Default menu: *Vector ► Geometry Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
verwendete Berechnung	`CALC_METHOD`	[Aufzählung] Standard: 0	Calculation parameters to use for the geometric properties. One of: • 0 — Layer-KBS • 1 — Projekt-KBS • 2 — Ellipsoidisch
Added geom info	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output (input copy with geometry) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Added geom info	`OUTPUT`	[wie Eingabe]	Copy of the input vector layer with the addition of the geometry fields

Pythoncode

Algorithm ID: `native:exportaddgeometrycolumns`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Affine transform

Applies an affine transformation to the layer geometries. Affine transformations can include translation, scaling and rotation. The operations are performed in the following order: scale, rotation, and translation.

Z and M values (if present) can be translated and scaled.

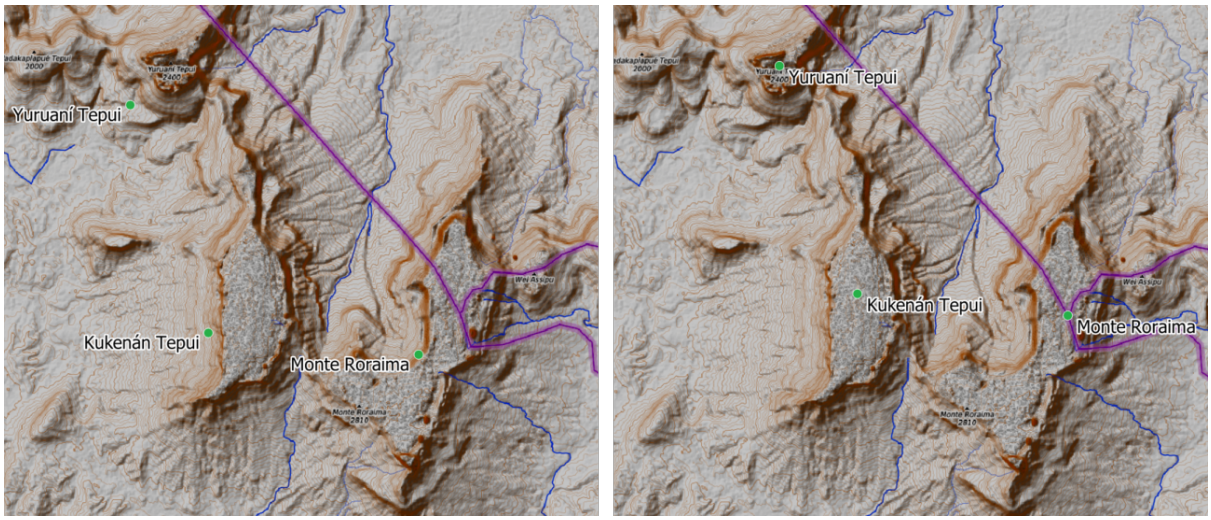


Abb. 24.89: Vektor-Punkt-Layer (grüne Punkte) vor (links) und nach (rechts) einer affinen Transformation (Translation).

Allows *features in-place modification* of point, line, and polygon features

Siehe auch

Translate

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Translation (x-axis)	`DELTA_X`	[numeric: double] Default: 0.0	Displacement to apply on the X axis.
Translation (y-axis)	`DELTA_Y`	[numeric: double] Default: 0.0	Displacement to apply on the Y axis.
Translation (z-axis)	`DELTA_Z`	[numeric: double] Default: 0.0	Displacement to apply on the Z axis.
Translation (m-values)	`DELTA_M`	[numeric: double] Default: 0.0	Offset to apply on m values.

Fortsetzung auf der nächsten Seite

Tab. 24.601 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Scale factor (x-axis)	`SCALE_X`	[numeric: double] Standard: 1,0	Scaling value (expansion or contraction) to apply on the X axis.
Scale factor (y-axis)	`SCALE_Y`	[numeric: double] Standard: 1,0	Scaling value (expansion or contraction) to apply on the Y axis.
Scale factor (z-axis)	`SCALE_Z`	[numeric: double] Standard: 1,0	Scaling value (expansion or contraction) to apply on the Z axis.
Scale factor (m-values)	`SCALE_M`	[numeric: double] Standard: 1,0	Scaling value (expansion or contraction) to apply on m values.
Rotation around z-axis (degrees counter-clockwise)	`ROTATION_Z`	[numeric: double] Default: 0.0	Angle of the rotation in degrees.
Transformed	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Transformed	`OUTPUT`	[wie Eingabe]	Output (transformed) vector layer.

Pythoncode

Algorithm ID: `native:affinetransform`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Aggregate

Takes a vector or table layer and creates a new layer by aggregating features based on a `group by` expression.

Features for which `group by` expression returns the same value are grouped together.

It is possible to group all source features together using constant value in `group by` parameter, example: NULL.

It is also possible to group features by multiple fields using Array function, example: Array(„Field1“, „Field2“).

Geometries (if present) are combined into one multipart geometry for each group. Output attributes are computed depending on each given aggregate definition.

This algorithm allows to use the default *aggregates functions* of the QGIS Expression engine.

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Siehe auch

Collect geometries, *Dissolve*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer
Group by expression	`GROUP_BY`	[tablefield: any] Default: ‚NULL‘	Choose the grouping field. If *NULL* all features will be grouped.

Fortsetzung auf der nächsten Seite

Tab. 24.603 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Aggregates	AGGREGATES	[list]	List of output layer field definitions. Example of a field definition: *{‚aggregate‘: ‚sum‘, ‚delimiter‘: „‘, ‚input‘: , $area‘, ‚length‘: 10, ‚name‘: ‚totarea‘, ‚precision‘: 0, ‚type‘: 6}* By default, the list contains all the fields of the input layer. In the GUI, you can edit these fields and their definitions, and you can also: • Click the button to add a new field. • Click to delete the selected field. • Use and to change order of the fields. • Click to reset to the default (the fields of the input layer). For each of the fields you’d like to retrieve information from, you need to define the following: ***Input expression* (`input`) [expression]** Field or expression from the input layer. ***Aggregate function* (`aggregate`) [enumeration]** Function to use on the input expression to return the aggregated value. These are mainly the *Aggregatsfunktionen*, plus dedicated „first_value“ and „last_value“ functions for ordered list items. Default: *concatenate* (for string data type), *sum* (for numeric data type) ***Delimiter* (`delimiter`) [string]** Text string to separate aggregated values, for example in case of concatenation. Default: , ***Output field name* (`name`) [string]** Name of the aggregated field in the output layer. By default input field name is kept. ***Type* (`type`) [enumeration]** Data type of the output field. Available types may not be compatible with the output layer provider. One of: **Achtung** For certain field types, e.g. lists, an extra `sub_type` parameter helps refine the specific type of the data. It is automatically set in the GUI but may be needed if you’re running the algorithm in Python or from the command line. • 1 — Boolescher Wert • 2 — Ganzzahl (32bit) • 4 — Ganzzahl (64bit)

Tab. 24.603 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Load fields from layer	GUI only	[Vektor: alle]	You can load fields from another layer and use them for the aggregation
Aggregated	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output (aggregate) layer *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Aggregated	`OUTPUT`	[wie Eingabe]	Multigeometry vector layer with the aggregated values

Pythoncode

Algorithm ID: `native:aggregate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Boundary

Returns the closure of the combinatorial boundary of the input geometries (i.e. the topological boundary of the geometry).

Only for polygon and line layers.

For **polygon geometries** , the boundary consists of all the lines making up the rings of the polygon.

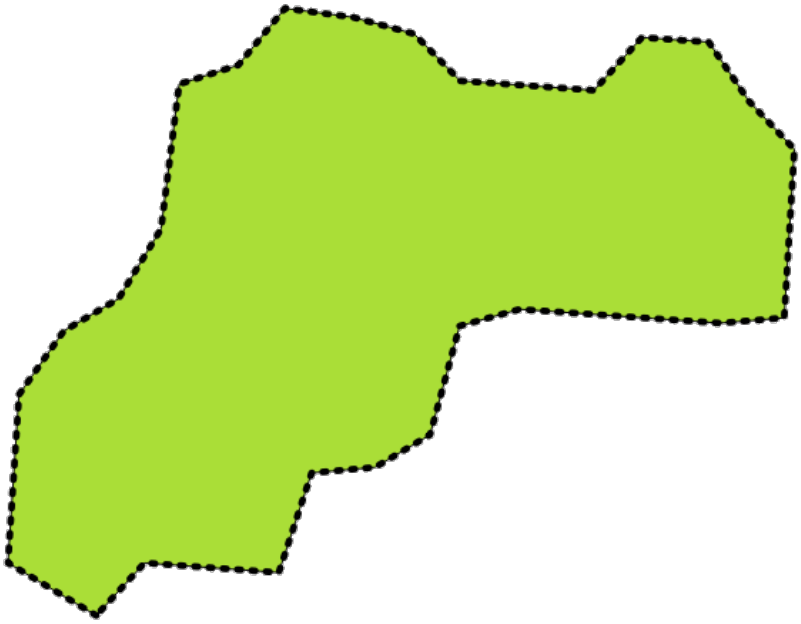


Abb. 24.90: Boundaries (black dashed line) of the source polygon layer

For **lines geometries**, the boundaries are their end points.

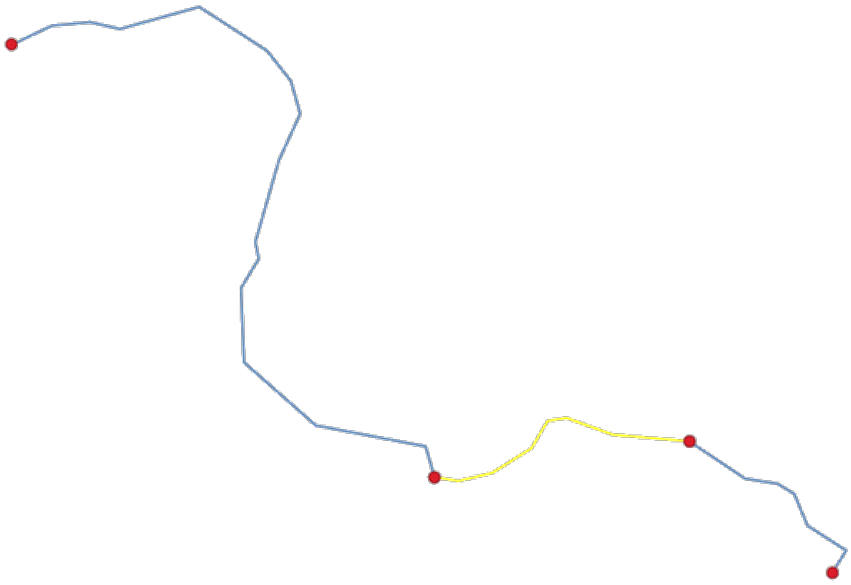


Abb. 24.91: Boundary layer (red points) for lines. In yellow a selected feature.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Boundary	`OUTPUT`	[vector: point, line] Standard: `[Create temporary layer]`	Specify the output (boundary) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Boundary	`OUTPUT`	[vector: point, line]	Boundaries from the input layer (point for line, and line for polygon)

Pythoncode

Algorithm ID: `native:boundary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Bounding boxes

Calculates the bounding box (envelope) of each feature in an input layer. Polygon and line geometries are supported.

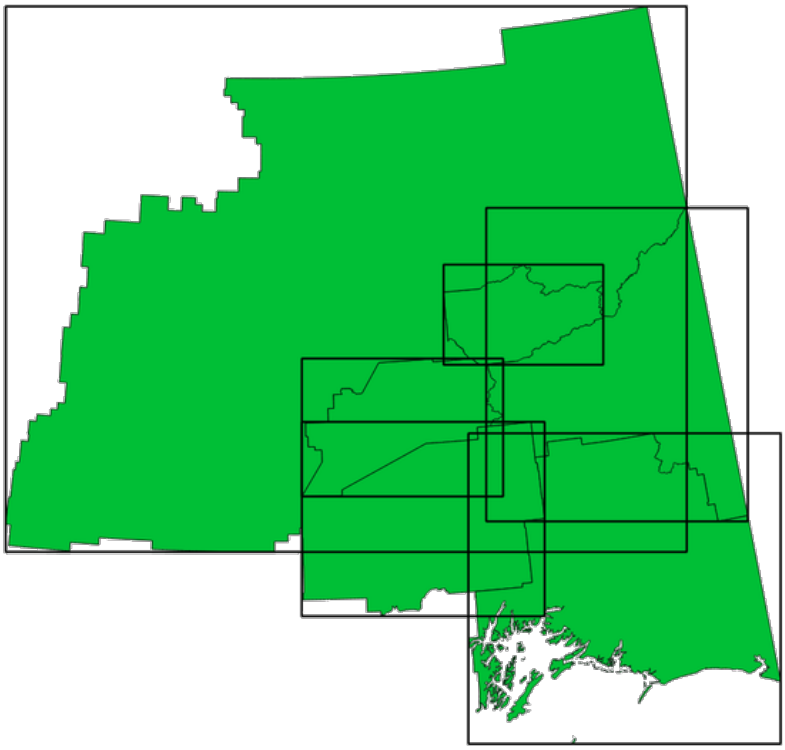


Abb. 24.92: Schwarze Linien stellen die Begrenzungslinien der einzelnen Polygonobjekte dar

Allows *features in-place modification* of polygon features

Siehe auch

Minimum bounding geometry

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Bounds	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output (bounding box) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Bounds	`OUTPUT`	[Vektor: Polygon]	Bounding boxes of input layer. Other than the input attributes, the output layer also contains following fields: `width`, `height`, `area` and `perimeter` of the generated polygon.

Pythoncode

Algorithm ID: `native:boundingboxes`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Puffer

Berechnet einen Pufferbereich für alle Objekte in einem Eingabelayer, wobei eine feste oder datendefinierte Distanz verwendet wird.

Es kann eine negative Distanz für Polygoneingabelayer verwendet werden. In diesem Fall führt der Puffer zu einem kleineren Polygon (Rücksprung).

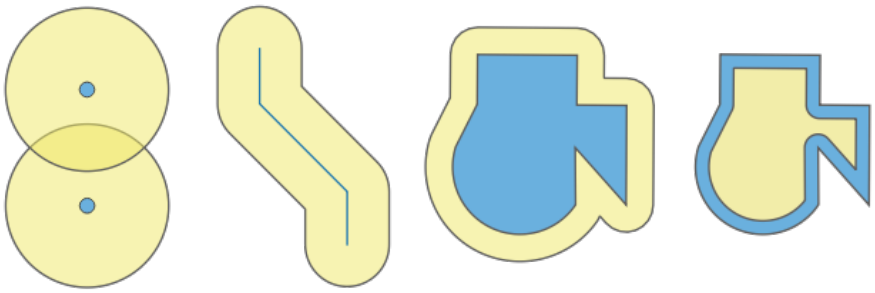


Abb. 24.93: Puffer (in gelb) von Punkten, Linie, Polygon mit positivem Puffer und Polygon mit negativem Puffer

Allows *features in-place modification* of polygon features

Default menu: *Vector* ► *Geoprocessing Tools*

Warnung

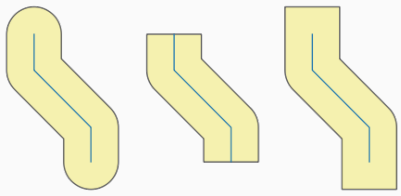
This algorithm may drop existing primary keys or FID values and regenerate them in output layers, depending on the input parameters.

Siehe auch

Variable distance buffer, *Multi-ring buffer (constant distance)*, *Variable width buffer (by M value)*

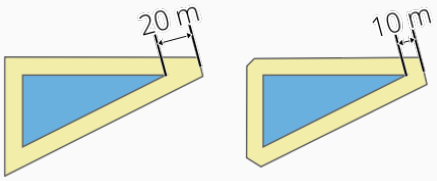
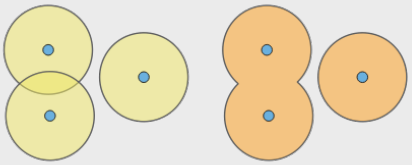
Parameter

Grundlegende Parameter

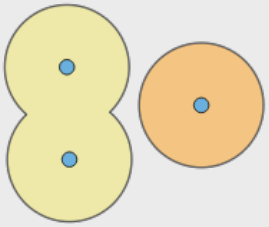
Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Abstand	`DISTANCE`	[numeric: double] Default: 10.0	Pufferabstand (von der Grenze jedes Objekts). Mit der Schaltfläche „Datendefinierte Übersteuerung“ auf der rechten Seite kann ein Feld ausgewählt werden, aus dem der Radius berechnet wird. Auf diese Weise kann für jedes Objekt ein anderer Radius festgelegt werden.
Segmente	`SEGMENTS`	[numerisch: Ganzzahl] Default: 5	Steuert die Anzahl der Liniensegmente, die zur Annäherung an einen Viertelkreis beim Erstellen von abgerundeten Puffern verwendet werden sollen.
Endstil	`END_CAP_STYLE`	[Aufzählung] Standard: 0	Steuert, wie Linienenden im Puffer behandelt werden. Mögliche Optionen: • 0 — Rund • 1 — Flach • 2 — Quadratisch  Abb. 24.94: Runder, flacher und quadratischer Endstil
Verbindungsstil	`JOIN_STYLE`	[Aufzählung] Standard: 0	Legt fest, ob beim Puffern von Ecken in einer Linie runde, Gehrungs- oder abgeschrägte Verbindungen verwendet werden sollen. Mögliche Optionen: • 0 — Rund • 1 — Eckig • 2 — Abgefast Abb. 24.95: Runder, eckiger und abgefaster Verbindungsstil

Fortsetzung auf der nächsten Seite

Tab. 24.609 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Eckgrenze	`MITER_LIMIT`	[numerisch: Double] Default: 2.0	Sets the maximum distance from the offset geometry to use when creating a mitered join as a factor of the offset distance (only applicable for miter join styles). Minimum: 1.0  Abb. 24.96: A 10m buffer with a limit of 2 and a 10m buffer with a limit of 1
Ergebnis auflösen	`DISSOLVE`	[Boolean] Standard: falsch	Dissolve the final buffer. If `True` (checked), overlapping buffers will be dissolved (combined) into a single multipart feature.  Abb. 24.97: Standard (three single part features - left), dissolved (1 multipart feature with 2 parts - right)
Gepuffert	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output (buffer) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Unzusammenhänge Ergebnisse getrennt halten	`SEPARATE_DISJO`	[Boolean] Standard: falsch	If `True` (checked) and dissolved is checked, features that do not overlap or touch will be exported as separate features (instead of parts of a multipart feature).  Abb. 24.98: Results in 2 single part features

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Gepuffert	`OUTPUT`	[Vektor: Polygon]	Output (buffer) polygon layer

Pythoncode

Algorithm ID: `native:buffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Centroids

Creates a new point layer, with points representing the centroids of the geometries of the input layer.

The centroid is a single point representing the barycenter (of all parts) of the feature, so it can be outside the feature borders. But can also be a point on each part of the feature.

The attributes of the points in the output layer are the same as for the original features.

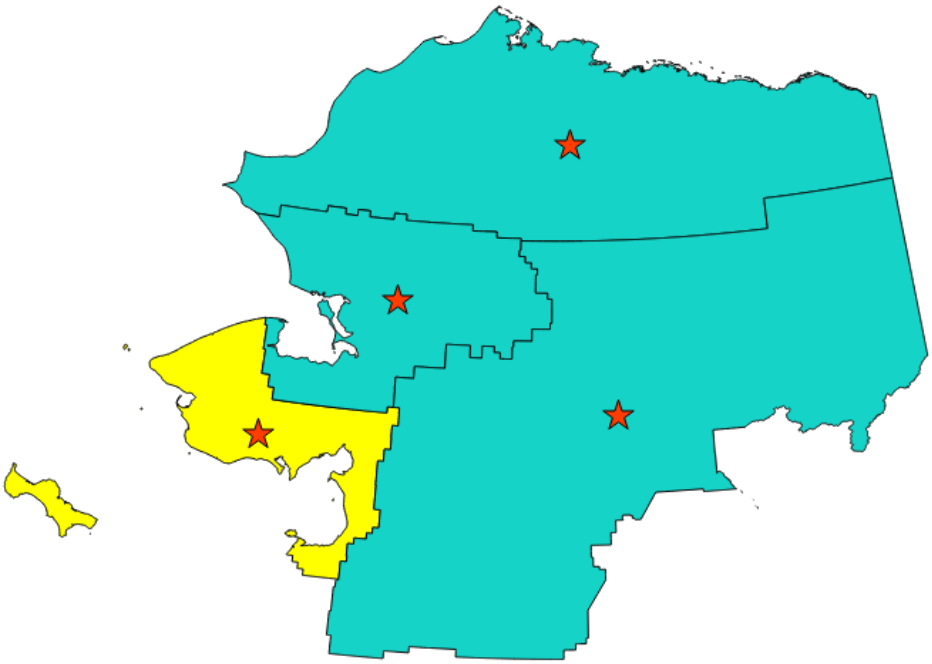


Abb. 24.99: Die roten Sterne stellen die Zentren der Objekte des Input Layers dar.

Erlaubt auch die *Insitu-Bearbeitung* von Punkt-Objekten

Default menu: *Vector ► Geometry Tools*

Warnung

This algorithm may drop existing primary keys or FID values and regenerate them in output layers, depending on the input parameters.

Siehe auch

Point on Surface

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Create centroid for each part	`ALL_PARTS`	[boolean] Standard: falsch	If True (checked), a centroid will be created for each part of the geometry

Fortsetzung auf der nächsten Seite

Tab. 24.612 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Centroids	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output (centroid) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Centroids	`OUTPUT`	[Vektor: Punkt]	Output point vector layer (centroids)

Pythoncode

Algorithm ID: `native:centroids`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Check validity

Performs a validity check on the geometries of a vector layer.

The geometries are classified in three groups (valid, invalid and error) and for each group, a vector layer with its features is generated:

- The **Valid output** layer contains only the valid features (without topological errors).
- The **Invalid output** layer contains all the invalid features found by the algorithm.
- The **Error output** layer is a point layer that points to where the invalid features were found.

The attribute tables of the generated layers will contain some additional information („message“ for the **error** layer, „FID“ and „_errors“ for the **invalid** layer and only „FID“ for the **valid** layer):

The attribute table of each generated vector layer will contain some additional information (number of errors found and types of error):

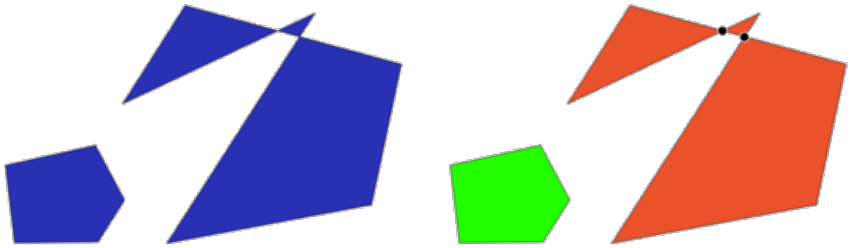


Abb. 24.100: Left: the input layer. Right: the valid layer (green), the invalid layer (orange)

Default menu: *Vector* ► *Geometry Tools*

Siehe auch

Check Geometry, *Fix geometries*, *Validate coverage*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT_LAYER`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Methode	`METHOD`	[Aufzählung] Default: 2	Method to use to check validity. Options: • 0: The one selected in digitizing settings • 1: QGIS • 2: GEOS
Ignore ring self intersection	`IGNORE_RING_SE`	[Boolean] Standard: falsch	Ignore self intersecting rings when checking for validity.
Valid output	`VALID_OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the vector layer to contain a copy of the valid features of the source layer. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Invalid output	`INVALID_OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Vector layer containing copy of the invalid features of the source layer with the field `_errors` listing the summary of the error(s) found. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortsetzung auf der nächsten Seite

Tab. 24.614 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Error output	`ERROR_OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Point layer of the exact position of the validity problems detected with the `message` field describing the error(s) found. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Count of errors	`ERROR_COUNT`	[numerisch: Ganzzahl]	The number of geometries that caused errors.
Error output	`ERROR_OUTPUT`	[Vektor: Punkt]	Point layer of the exact position of the validity problems detected with the `message` field describing the error(s) found.
Count of invalid features	`INVALID_COUNT`	[numerisch: Ganzzahl]	The number of invalid geometries.
Invalid output	`INVALID_OUTPUT`	[wie Eingabe]	Vector layer containing copy of the invalid features of the source layer with the field `_errors` listing the summary of the error(s) found.
Count of valid features	`VALID_COUNT`	[numerisch: Ganzzahl]	The number of valid geometries.
Valid output	`VALID_OUTPUT`	[wie Eingabe]	Vector layer containing a copy of the valid features of the source layer.

Pythoncode

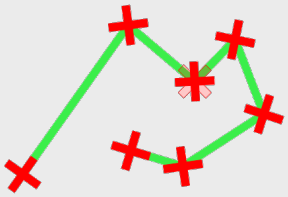
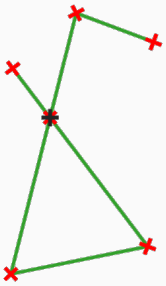
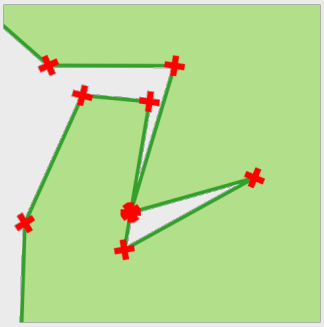
Algorithm ID: `native:checkvalidity`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

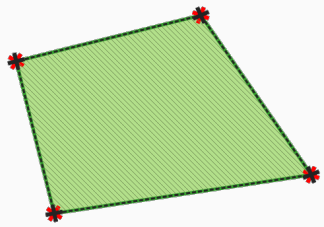
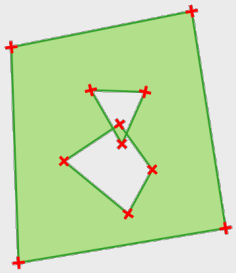
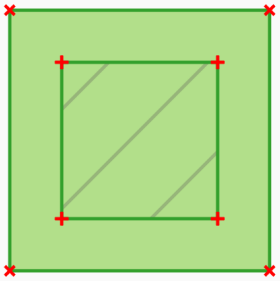
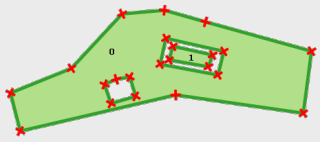
Types of error messages and their meanings

Tab. 24.616: Some examples of geometry check failures in QGIS

Geometry context	GEOS validation and error message	QGIS validation and error message
Consecutive points on a line have the same coordinates 	*Repeated point*	*Line a contains x duplicate node(s) at b*
Segments of a line intersect each other 		*Segments a and b of line c intersect at d*
Polygon geometry touches itself and generates a ring 	*Ring self-intersection*	*Ring self-intersection*

Fortsetzung auf der nächsten Seite

Tab. 24.616 – Fortsetzung der vorherigen Seite

Geometry context	GEOS validation and error message	QGIS validation and error message
Two rings (exterior or interior) of a polygon geometry are identical 	*Duplicate rings*	
Geometry touches itself 	*Self-intersection*	
A polygon geometry is on top of another polygon geometry 	*Nested shell*	
Part of a MultiPolygon geometry is within a hole of a MultiPolygon geometry 		*Polygon a lies inside polygon b*

Fortsetzung auf der nächsten Seite

Tab. 24.616 – Fortsetzung der vorherigen Seite

Geometry context	GEOS validation and error message	QGIS validation and error message
Point geometry does not have a proper coordinate pair. The coordinate pair does not contain a latitude value and a longitude value in that order.	*Invalid coordinate*	

Collect geometries

Takes a vector layer and collects its geometries into new multipart geometries.

One or more attributes can be specified to collect only geometries belonging to the same class (having the same value for the specified attributes), alternatively all geometries can be collected.

All output geometries will be converted to multi geometries, even those with just a single part. This algorithm does not dissolve overlapping geometries - they will be collected together without modifying the shape of each geometry part.

See the ‚Promote to multipart' or ‚Aggregate' algorithms for alternative options.

Default menu: *Vector ► Geometry Tools*

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Aggregate, *Promote to multipart*, *Dissolve*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Unique ID fields	`FIELD`	[tablefield: any] [list]	Choose one or more attributes to collect the geometries
Collected	`OUTPUT`	[wie Eingabe]	Vector layer with collected geometries

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Collected	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer for the collected geometries. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Pythoncode

Algorithm ID: `native:collect`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Concave hull

Computes the concave hull of the features from an input point layer.

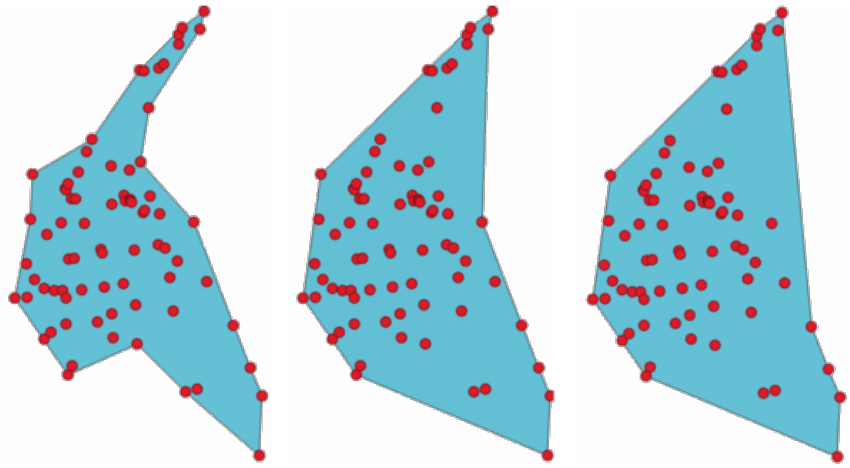


Abb. 24.101: Concave hulls with different thresholds (0.3, 0.6, 0.9)

Siehe auch

Convex hull

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input point layer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Threshold	`ALPHA`	[numerisch: Double] Default: 0.3	Number from 0 (maximum concave hull) to 1 (convex hull).
Allow holes	`HOLES`	[Boolean] Standard: True	Choose whether to allow holes in the final concave hull
Split multipart geometry into singlepart geometries	`NO_MULTIGEOMET`	[Boolean] Standard: falsch	Check if you want to have singlepart geometries instead of multipart ones.
Concave hull	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Concave hull	`OUTPUT`	[Vektor: Polygon]	The output vector layer

Pythoncode

Algorithm ID: `native:concavehull`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Convert geometry type

Generates a new layer based on an existing one, with a different type of geometry.

The attribute table of the output layer is the same as the one of the input layer.

Not all conversions are possible. For instance, a line layer can be converted to a point layer, but a point layer cannot be converted to a line layer.

Siehe auch

Polygonize, *Lines to polygons*, *Polygons to lines*, *Points to path*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
New geometry type	`TYPE`	[Aufzählung] Standard: 0	Geometry type to apply to the output features. One of: • 0 — Centroids • 1 — Nodes • 2 — Linestrings • 3 — Multilinestrings • 4 — Polygons
Konvertiert	`OUTPUT`	[Vektor: Geometrie] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Konvertiert	`OUTPUT`	[Vektor: Geometrie]	Output vector layer - the type depends on the parameters

Pythoncode

Algorithm ID: `native:convertgeometrytype`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Convert to curved geometries

Converts a geometry into its curved geometry equivalent.

Already curved geometries will be retained without change.

Allows *features in-place modification* of line and polygon features

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line or polygon]	Eingabe-Vektorlayer
Maximum distance tolerance	`DISTANCE`	[numerisch: Double] Default: 0.000001	The maximum distance allowed between the original location of vertices and where they would fall on the converted curved geometries
Maximum angle tolerance	`ANGLE`	[numerisch: Double] Default: 0.000001	Segments are considered as suitable for replacing with an arc if the points are all regularly spaced on the candidate arc. This parameter specifies the maximum angular deviation (in degrees) allowed when testing for regular point spacing. Between 0 and 45°.
Curves	`OUTPUT`	[vector: compoundcurve or curvepolygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Curves	`OUTPUT`	[vector: compoundcurve or curvepolygon]	Output vector layer with curved geometries

Pythoncode

Algorithm ID: `native:converttocurves`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Convex hull

Calculates the convex hull for each feature in an input layer.

See the ‚Minimum bounding geometry' algorithm for a convex hull calculation which covers the whole layer or grouped subsets of features.

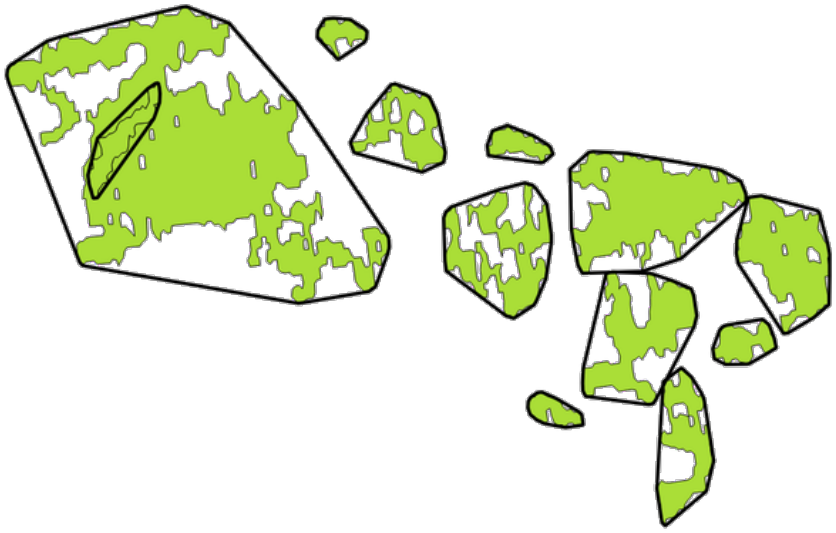


Abb. 24.102: Black lines identify the convex hull for each layer feature

Allows *features in-place modification* of polygon features

Default menu: *Vector* ► *Geoprocessing Tools*

Siehe auch

Minimum bounding geometry, *Concave hull*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Convex hull	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Convex hull	`OUTPUT`	[Vektor: Polygon]	The output (convex hull) vector layer. Other than the input attributes, the output layer also contains the following fields: `area` and `perimeter` of the generated polygon.

Pythoncode

Algorithm ID: `native:convexhull`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Create wedge buffers

Creates wedge shaped buffers from input points.

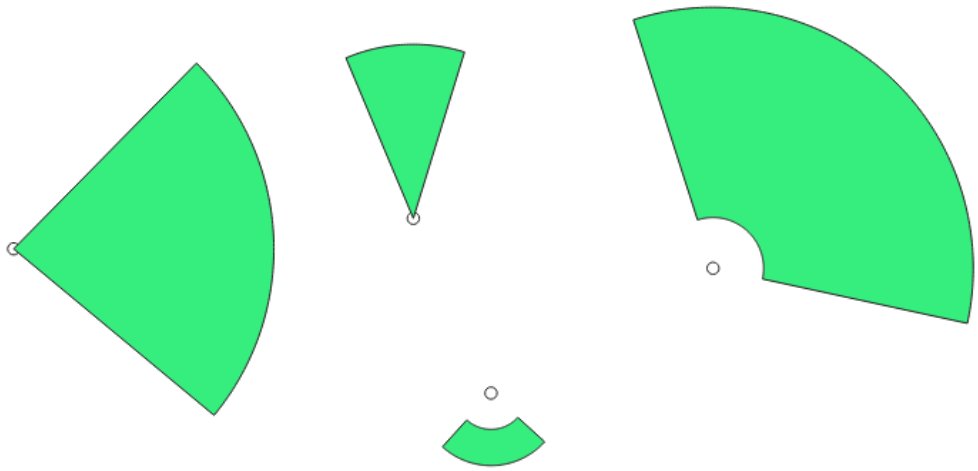


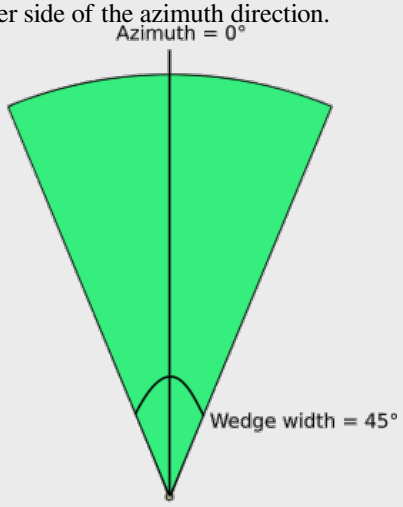
Abb. 24.103: Wedge buffers

The native output from this algorithm are CurvePolygon geometries, but these may be automatically segmentized to Polygons depending on the output format.

Siehe auch

Puffer, *Variable width buffer (by M value)*, *Tapered buffers*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Azimuth (degrees from North)	`AZIMUTH`	[numeric: double] Default: 0.0	Angle (in degrees) as the middle value of the wedge
Wedge width (in degrees)	`WIDTH`	[numeric: double] Default: 45.0	Width (in degrees) of the buffer. The wedge will extend to half of the angular width either side of the azimuth direction.  Abb. 24.104: Azimuth and width values of the wedge buffer
Outer radius	`OUTER_RADIUS`	[numeric: double] Standard: 1,0	The outer *size* (length) of the wedge: the size is meant from the source point to the edge of the wedge shape.
Inner radius Optional	`INNER_RADIUS`	[numeric: double] Default: 0.0	Inner radius value. If 0 the wedge will begin from the source point.
Buffers	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Buffers	`OUTPUT`	[Vektor: Polygon]	The output (wedge buffer) vector layer

Pythoncode

Algorithm ID: `native:wedgebuffers`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Delaunay triangulation

Creates a polygon layer with the Delaunay triangulation corresponding to the input point layer.

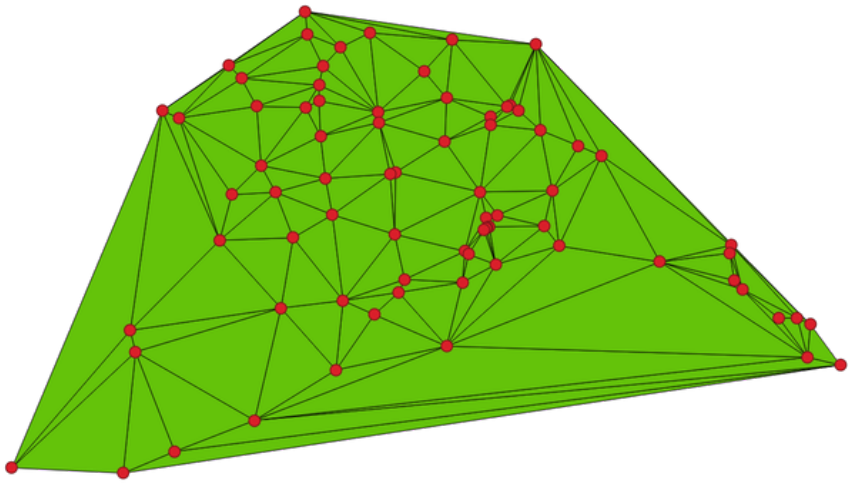


Abb. 24.105: Delaunay triangulation on points

Default menu: *Vector ► Geometry Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Tolerance Optional	`TOLERANCE`	[numerisch: Double] Default: 0.0	Specifies an optional snapping tolerance which can be used to improve the robustness of the triangulation.

Fortsetzung auf der nächsten Seite

Tab. 24.629 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Add point IDs to output	`ADD_ATTRIBUTES`	[Boolean] Standard: True	Specifies whether fields storing involved point features ID should be added to the output. If False, an `id` field is used to identify the polygons.
Delaunay triangulation	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Delaunay triangulation	`OUTPUT`	[Vektor: Polygon]	The output (Delaunay triangulation) vector layer

Pythoncode

Algorithm ID: `native:delaunaytriangulation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Delete holes

Takes a polygon layer and removes holes in polygons. It creates a new vector layer in which polygons with holes have been replaced by polygons with only their external ring. Attributes are not modified.

An optional minimum area parameter allows removing only holes which are smaller than a specified area threshold. Leaving this parameter at `0.0` results in all holes being removed.

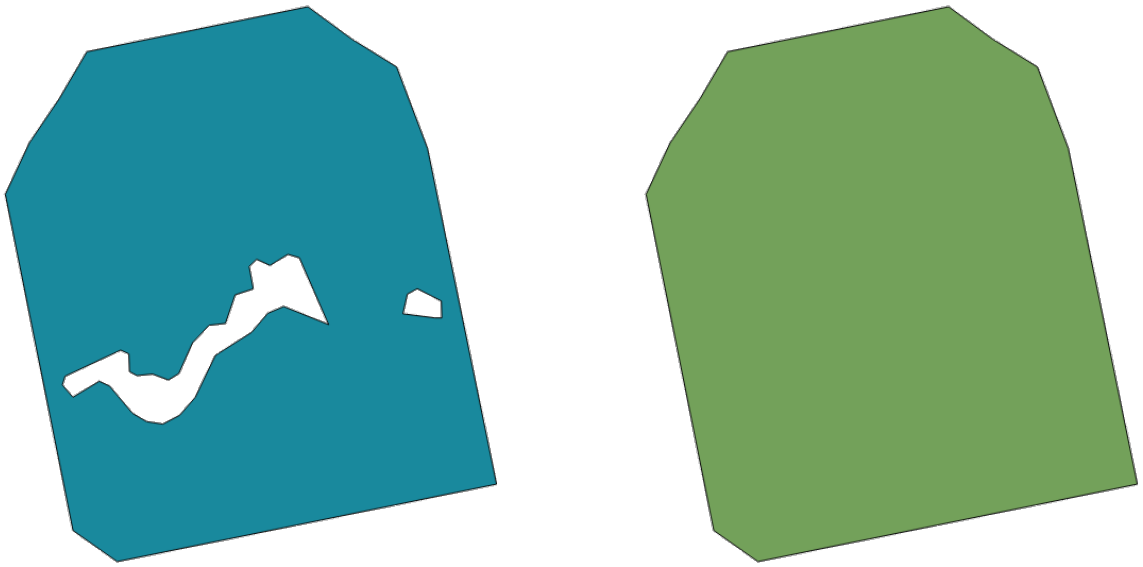


Abb. 24.106: Before and after the cleaning

Allows *features in-place modification* of polygon features

Siehe auch

qgisfixgeometryhole

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Input polygon vector layer
Remove holes with area less than Optional	`MIN_AREA`	[numeric: double] Default: 0.0	Only holes with an area less than this threshold will be deleted. With a value of `0.0`, **all** the holes will be deleted.
Cleaned	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Cleaned	`OUTPUT`	[wie Eingabe]	The output (cleaned) vector layer

Pythoncode

Algorithm ID: `native:deleteholes`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Densify by count

Takes a polygon or line layer and generates a new one in which the geometries have a larger number of vertices than the original one.

If the geometries have Z or M values present then these will be linearly interpolated at the added vertices.

The number of new vertices to add to each segment is specified as an input parameter.

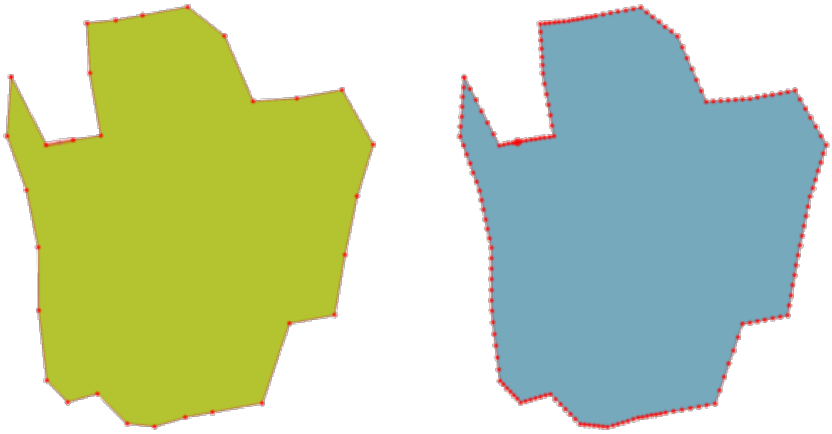


Abb. 24.107: Rote Punkte zeigen die Knotenpunkte vor und nach der Verdichtung

Allows *features in-place modification* of line and polygon features

Default menu: *Vector* ► *Geometry Tools*

Siehe auch

Densify by interval

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Vertices to add	`VERTICES`	[numerisch: Ganzzahl] Standard: 1	Number of vertices to add to each segment
Densified	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Densified	`OUTPUT`	[wie Eingabe]	The output (densified) vector layer

Pythoncode

Algorithm ID: `native:densifygeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Densify by interval

Takes a polygon or line layer and generates a new one in which the geometries have a larger number of vertices than the original one.

The geometries are densified by adding regularly placed extra vertices inside each segment so that the maximum distance between any two vertices does not exceed the specified distance.

If the geometries have Z or M values present then these will be linearly interpolated at the added vertices.

Beispiel

Specifying a distance of 3 would cause the segment `[0 0] -> [10 0]` to be converted to `[0 0] -> [2.5 0] -> [5 0] -> [7.5 0] -> [10 0]`, since 3 extra vertices are required on the segment and spacing these at 2.5 increments allows them to be evenly spaced over the segment.

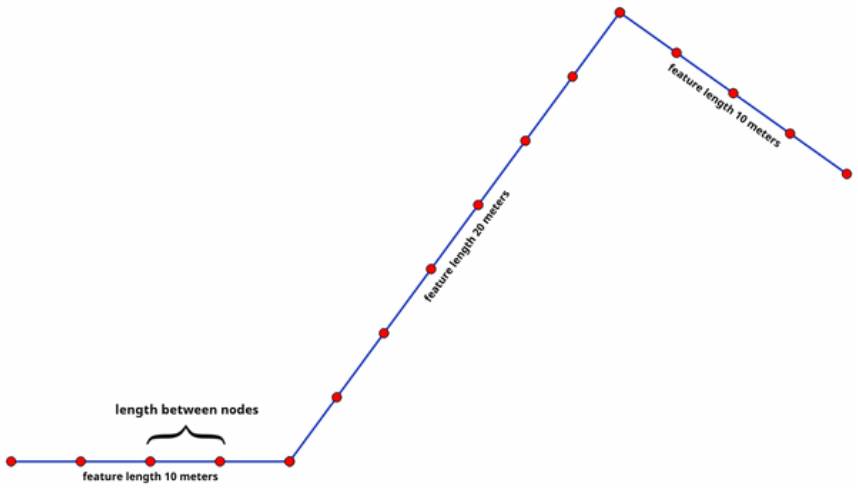


Abb. 24.108: Verdichtung der Geometrie in einem bestimmten Intervall

Allows *features in-place modification* of line and polygon features

Siehe auch

Densify by count

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Interval between vertices to add	`INTERVAL`	[numeric: double] Standard: 1,0	Maximum distance between two consecutive vertices
Densified	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Densified	`OUTPUT`	[wie Eingabe]	The output (densified) vector layer

Pythoncode

Algorithm ID: `native:densifygeometriesgivenaninterval`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Dissolve

Takes a vector layer and combines its features into new features. One or more attributes can be specified to dissolve features belonging to the same class (having the same value for the specified attributes), alternatively all features can be dissolved to a single feature.

All output geometries will be converted to multi geometries. In case the input is a polygon layer, common boundaries of adjacent polygons being dissolved will get erased. If enabled, the optional „Keep disjoint features separate“ setting will cause features and parts that do not overlap or touch to be exported as separate features (instead of parts of a single multipart feature).

The resulting attribute table will have the same fields as the input layer. The values in the output layer's fields are the ones of the first input feature that happens to be processed.

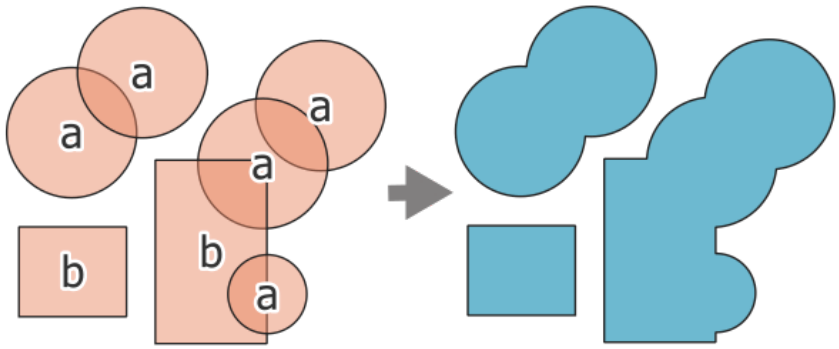


Abb. 24.109: Dissolving a layer into a single multipart feature

Default menu: *Vector* ► *Geoprocessing Tools*

Warnung

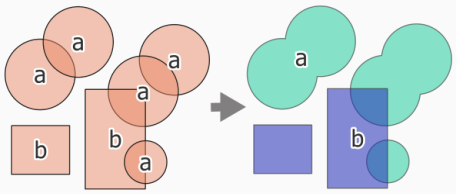
This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

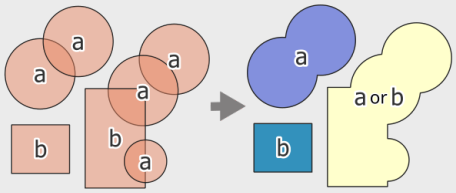
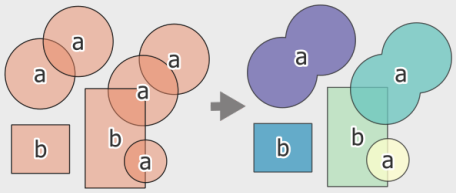
Dissolve coverage, *Aggregate*, *Collect geometries*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Dissolve field(s) Optional	`FIELD`	[tablefield: any] [list] Default: []	Features having the same value for the selected field(s) will be replaced with a single one and their geometries are merged. If no field is provided then all the features are dissolved, resulting in a single (multipart) feature.  Abb. 24.110: Dissolve the polygon layer on a common attribute (2 multipart features)
Dissolved	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Unzusammenhänge Ergebnisse getrennt halten	SEPARATE_DISJO	[Boolean] Standard: falsch	Parts of dissolved features are exported as separate features (instead of parts of a multipart feature).  Abb. 24.111: source (left), dissolve all (3 distinct features - right)  Abb. 24.112: source (left), dissolve on field (5 distinct features - right)

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Dissolved	OUTPUT	[wie Eingabe]	The output vector layer with dissolved geometries

Pythoncode

Algorithm ID: `native:dissolve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Drape (set Z value from raster)

Uses values sampled from a band within a raster layer to set the Z value for every overlapping vertex in the feature geometry. The raster values can optionally be scaled by a preset amount.

If Z values already exist in the layer, they will be overwritten with the new value. If no Z values exist, the geometry will be upgraded to include the Z dimension.

Allows *features in-place modification* of point, line, and polygon features with Z enabled

Siehe auch

Set M value from raster, *Set Z value*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Raster layer	`RASTER`	[Raster]	Raster layer with Z values
Band number	`BAND`	[raster band] Standard: 1	The raster band to take the Z values from
Value for NoData or non-intersecting vertices	`NODATA`	[numeric: double] Default: 0.0	Value to use in case the vertex does not intersect (a valid pixel of) the raster
Scale factor	`SCALE`	[numeric: double] Standard: 1,0	Scaling value: the band values are multiplied by this value.
Offset	`OFFSET`	[numeric: double] Default: 0.0	Offset value: it is algebraically added to the band values after applying the „Scale factor“.
Updated	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer (with Z values from the raster layer). *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Updated	`OUTPUT`	[wie Eingabe]	The output vector layer with Z values from the raster layer

Pythoncode

Algorithm ID: `native:setzfromraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Drop M/Z values

Removes M (measure) or Z (altitude) values from input geometries.

Siehe auch

M-Wert einstellen, *Set Z value*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Input vector layer with M or Z values
Drop M Values	`DROP_M_VALUES`	[Boolean] Standard: falsch	Removes the M values from the geometries
Drop Z Values	`DROP_Z_VALUES`	[Boolean] Standard: falsch	Removes the Z values from the geometries
Z/M Dropped	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Z/M Dropped	`OUTPUT`	[wie Eingabe]	The output vector layer (identical to the input layer, except that the M and/or Z dimensions have been removed from the geometries).

Pythoncode

Algorithm ID: `native:dropmzvalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Eliminate selected polygons

Combines selected polygons of the input layer with certain adjacent polygons by erasing their common boundary. The adjacent polygon can be either the one with the largest or smallest area or the one sharing the largest common boundary with the polygon to be eliminated.

Eliminate is normally used to get rid of sliver polygons, i.e. tiny polygons that are a result of polygon intersection processes where boundaries of the inputs are similar but not identical.

Default menu: *Vector* ► *Geoprocessing Tools*

Siehe auch

Fix geometries

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Input polygon vector layer
Merge selection with the neighboring polygon with the	`MODE`	[Aufzählung] Standard: Nicht gesetzt	Choose the parameter to use in order to get rid of the selected polygons: • 0 — Largest Area • 1 — Smallest Area • 2 — Largest Common Boundary
Eliminated	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Eliminated	`OUTPUT`	[Vektor: Polygon]	The output polygon vector layer.

Pythoncode

Algorithm ID: `qgis:eliminateselectedpolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Explode lines

Takes a lines layer and creates a new one in which each line layer is replaced by a set of lines representing the segments in the original line.

Each line in the resulting layer contains only a start and an end point, with no intermediate vertices between them.

If the input layer consists of CircularStrings or CompoundCurves, the output layer will be of the same type and contain only single curve segments.

Bemerkung

- This algorithm drops existing primary keys or FID values and regenerates them in output layers.
- This algorithm does not require valid geometries as input.

Abb. 24.113: The original line layer and the exploded one

Allows *features in-place modification* of line features

Siehe auch

Subdivide, *Line substring*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Exploded	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Exploded	`OUTPUT`	[vector: line]	The output line vector layer with features representing each segment of the input layer.

Pythoncode

Algorithm ID: `native:explodelines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extend lines

Extends line geometry by a specified amount at the start and end of the line.

Lines are extended using the bearing of the first and last segment in the line.

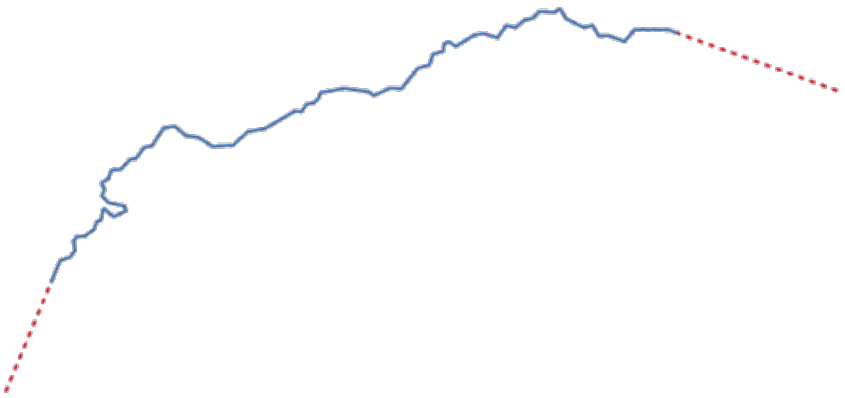


Abb. 24.114: Die roten Striche stellen die Verlängerung der ursprünglichen Geometrie dar

Allows *features in-place modification* of line features

Siehe auch

Line substring

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Start distance	`START_DISTANCE`	[numeric: double] Default: 0.0	Distance by which to extend the first segment of the line (starting point)
End distance	`END_DISTANCE`	[numeric: double] Default: 0.0	Distance by which to extend the last segment of the line (ending point)
Extended	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extended	`OUTPUT`	[vector: line]	The output (extended) line vector layer.

Pythoncode

Algorithm ID: `native:extendlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract M values

Extracts M values from geometries into feature attributes.

By default only the M value from the first vertex of each feature is extracted, however the algorithm can optionally calculate statistics on all of the geometry's M values, including sum, mean, minimum and maximum.

Siehe auch

Extract Z values, *M-Wert einstellen*, *Drop M/Z values*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Summaries to calculate	`SUMMARIES`	[Aufzählung] Default: [0]	Statistics on the M values of a geometry. One or more of: • 0 — First • 1 — Last • 2 — Count • 3 — Sum • 4 — Mean • 5 — Median • 6 — St.dev (pop) • 7 — Minimum • 8 — Maximum • 9 — Range • 10 — Minority • 11 — Majority • 12 — Variety • 13 — Q1 • 14 — Q3 • 15 — IQR
Output column prefix	`COLUMN_PREFIX`	[Text] Default: ‚m_‘	The prefix for the output (M) column
Extracted	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extracted	`OUTPUT`	[wie Eingabe]	The output vector layer (with M values)

Pythoncode

Algorithm ID: `native:extractmvalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract specific vertices

Takes a vector layer and generates a point layer with points representing specific vertices in the input geometries.

For instance, this algorithm can be used to extract the first or last vertices in the geometry. The attributes associated to each point are the same ones associated to the feature that the vertex belongs to.

The vertex indices parameter accepts a comma separated string specifying the indices of the vertices to extract. The first vertex corresponds to an index of 0, the second vertex has an index of 1, etc. Negative indices can be used to find vertices at the end of the geometry, e.g., an index of -1 corresponds to the last vertex, -2 corresponds to the second last vertex, etc.

Additional fields are added to the vertices indicating the specific vertex position (e.g., 0, -1, etc), the original vertex index, the vertex's part and its index within the part (as well as its ring for polygons), distance along the original geometry and bisector angle of vertex for the original geometry.

Erlaubt auch die *Insitu-Bearbeitung* von Punkt-Objekten

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Extract vertices, *Filter vertices by M value*, *Filter vertices by Z value*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Vertex indices	`VERTICES`	[Text] Default: ‚0'	Comma-separated string of the indices of the vertices to extract.
Vertices	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Vertices	`OUTPUT`	[Vektor: Punkt]	The output (point) vector layer containing the specified vertices from the input layer geometries. Other than the input attributes, the output layer also contains the following fields: • `vertex_pos`: the index of the vertex in the geometry, as requested by the user. • `vertex_index`: the index of the vertex within the geometry. The first vertex is indexed as 0. • `vertex_part`: the part number of the geometry that the vertex belongs to. For single-part geometries, this is always 0. For multi-part geometries (e.g. MultiPolygon or MultiLine), this indicates which part the vertex comes from. • `vertex_part_ring`: for polygon geometries, this indicates whether the vertex belongs to the outer ring (0) or one of the inner rings (1, 2, …). • `vertex_part_index`: the index of the vertex in its specific ring (outer or inner). • `distance`: the cumulative length of line segments from the first vertex of the geometry. • `angle`: the angle (in degrees) between the previous and the current vertex.

Pythoncode

Algorithm ID: `native:extractspecificvertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract vertices

Takes a vector layer and generates a point layer with points representing the vertices in the input geometries.

The attributes associated to each point are the same ones associated to the feature that the vertex belongs to.

Additional fields are added to the vertices indicating the vertex index (beginning at 0), the feature's part and its index within the part (as well as its ring for polygons), distance along original geometry and bisector angle of vertex for original geometry.

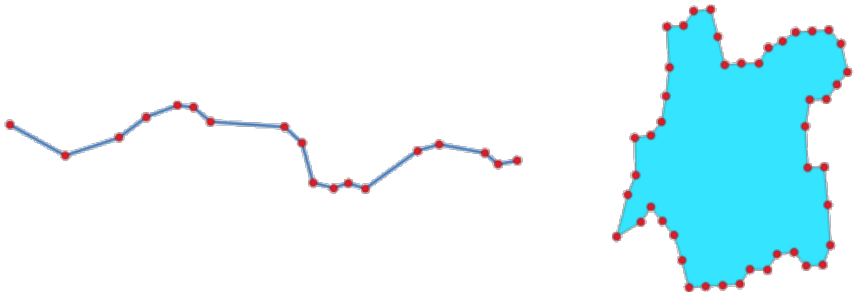


Abb. 24.115: Vertices extracted for line and polygon layer

Erlaubt auch die *Insitu-Bearbeitung* von Punkt-Objekten

Default menu: *Vector* ► *Geometry Tools*

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Extract specific vertices, *Filter vertices by M value*, *Filter vertices by Z value*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Vertices	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Vertices	`OUTPUT`	[Vektor: Punkt]	The output (point) vector layer containing the vertices from the input layer geometries. Other than the input attributes, the output layer also contains the following fields: • `vertex_index`: the index of the vertex within the geometry. The first vertex is indexed as 0. • `vertex_part`: the part number of the geometry that the vertex belongs to. For single-part geometries, this is always 0. For multi-part geometries (e.g. MultiPolygon or MultiLine), this indicates which part the vertex comes from. • `vertex_part_ring`: for polygon geometries, this indicates whether the vertex belongs to the outer ring (0) or one of the inner rings (1, 2, …). • `vertex_part_index`: the index of the vertex in its specific ring (outer or inner). • `distance`: the distance from the previous vertex. • `angle`: returns approximate angle at a vertex. Usually the average angle between adjacent segments, and can be pictured as the orientation of a line following the curvature of the geometry at the specified vertex.

Pythoncode

Algorithm ID: `native:extractvertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract Z values

Extracts Z values from geometries into feature attributes.

By default only the Z value from the first vertex of each feature is extracted, however the algorithm can optionally calculate statistics on all of the geometry's Z values, including sum, mean, minimum and maximum.

Siehe auch

Extract M values, *Set Z value*, *Drop M/Z values*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Summaries to calculate	`SUMMARIES`	[Aufzählung] Default: [0]	Statistics on the Z values of a geometry. One or more of: • 0 — First • 1 — Last • 2 — Count • 3 — Sum • 4 — Mean • 5 — Median • 6 — St.dev (pop) • 7 — Minimum • 8 — Maximum • 9 — Range • 10 — Minority • 11 — Majority • 12 — Variety • 13 — Q1 • 14 — Q3 • 15 — IQR
Output column prefix	`COLUMN_PREFIX`	[Text] Default: ‚z_'	The prefix for the output (Z) column
Extracted	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extracted	`OUTPUT`	[wie Eingabe]	The output vector layer (with Z values)

Pythoncode

Algorithm ID: `native:extractzvalues`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Filter vertices by M value

Filters away vertices based on their M value, returning geometries with only vertex points that have a M value greater than or equal to the specified minimum value and/or less than or equal to the maximum value.

If the minimum value is not specified then only the maximum value is tested, and similarly if the maximum value is not specified then only the minimum value is tested.

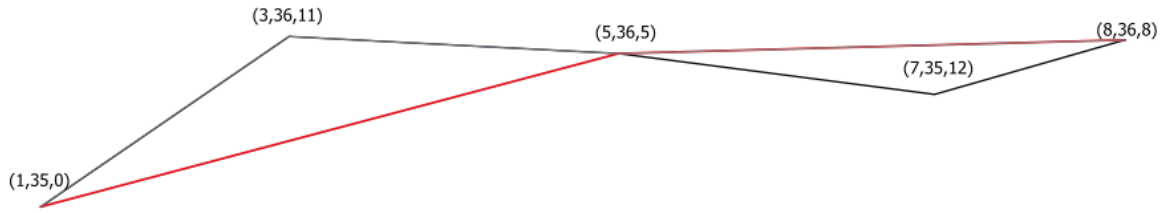


Abb. 24.116: The red line represents the black line with only vertices whose M value is <=10.

Allows *features in-place modification* of point, line and polygon features with M enabled

Warnung

Depending on the input geometry attributes and the filters used, the geometries created by this algorithm may no longer be valid.

Siehe auch

Filter vertices by Z value, *Extract vertices*, *Extract specific vertices*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Input vector layer to remove vertices from
Minimum Optional	`MIN`	[numeric: double] Standard: Nicht gesetzt	Minimum of M values allowed
Maximum Optional	`MAX`	[numeric: double] Standard: Nicht gesetzt	Maximum of M values allowed
Filtered	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Filtered	`OUTPUT`	[wie Eingabe]	The output vector layer of features with only the filtered vertices.

Pythoncode

Algorithm ID: `native:filterverticesbym`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Filter vertices by Z value

Filters away vertices based on their Z value, returning geometries with only vertex points that have a Z value greater than or equal to the specified minimum value and/or less than or equal to the maximum value.

If the minimum value is not specified then only the maximum value is tested, and similarly if the maximum value is not specified then only the minimum value is tested.

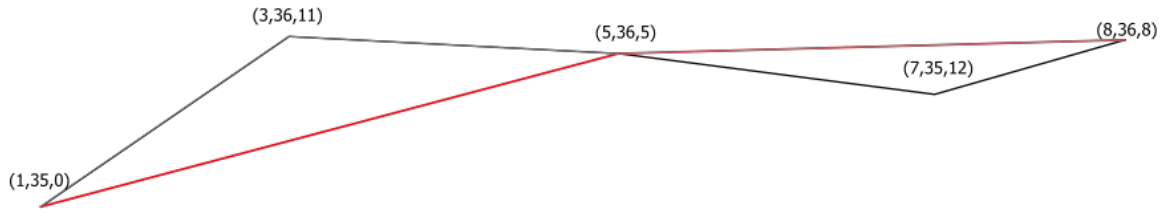


Abb. 24.117: The red line represents the black line with only vertices whose Z value is <=10.

Allows *features in-place modification* of point, line and polygon features with Z enabled

Warnung

Depending on the input geometry attributes and the filters used, the geometries created by this algorithm may no longer be valid. You may need to run the *Fix geometries* algorithm to ensure their validity.

Siehe auch

Filter vertices by M value, *Extract vertices*, *Extract specific vertices*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Input vector layer to remove vertices from
Minimum Optional	`MIN`	[numeric: double] Standard: Nicht gesetzt	Minimum of Z values allowed
Maximum Optional	`MAX`	[numeric: double] Standard: Nicht gesetzt	Maximum of Z values allowed

Fortsetzung auf der nächsten Seite

Tab. 24.660 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Filtered	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Filtered	`OUTPUT`	[wie Eingabe]	The output vector layer of features with only the filtered vertices.

Pythoncode

Algorithm ID: `native:filterverticesbyz`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Fix geometries

Attempts to create a valid representation of a given invalid geometry without losing any of the input vertices. Already valid geometries are returned without further intervention. Always outputs multi-geometry layer.

Allows *features in-place modification* of point, line, and polygon features without M enabled

Warnung

M values will be dropped from the output.

Siehe auch

Check validity, *Fix Geometry*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Repair method	`METHOD`	[Aufzählung] Standard: 1	Method used to repair the geometries. One of: • 0 — `Linework`: combines all rings into a set of noded lines and then extracts valid polygons from that 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. **Achtung** Using this method requires QGIS installed with GEOS >= 3.10 (see *Help* ► *About* menu).
Fixed geometries	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Fixed geometries	`OUTPUT`	[wie Eingabe]	The output vector layer with fixed geometries.

Pythoncode

Algorithm ID: `native:fixgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Force right-hand-rule

Forces polygon geometries to respect the Right-Hand-Rule, in which the area that is bounded by a polygon is to the right of the boundary. In particular, the exterior ring is oriented in a clockwise direction and any interior rings in a counter-clockwise direction.

Allows *features in-place modification* of polygon features

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Eingabe-Vektorlayer
Reoriented	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Reoriented	`OUTPUT`	[Vektor: Polygon]	The output vector layer with reoriented geometries.

Pythoncode

Algorithm ID: `native:forcerhr`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Geodesic line split at antimeridian

Splits a line into multiple geodesic segments, whenever the line crosses the antimeridian (±180 degrees longitude).

Splitting at the antimeridian helps the visual display of the lines in some projections. The returned geometry will always be a multi-part geometry.

Whenever line segments in the input geometry cross the antimeridian, they will be split into two segments, with the latitude of the breakpoint being determined using a geodesic line connecting the points either side of this segment. The current project ellipsoid setting will be used when calculating this breakpoint.

If the input geometry contains M or Z values, these will be linearly interpolated for the new vertices created at the antimeridian.

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Allows *features in-place modification* of line features

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Split	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Split	`OUTPUT`	[vector: line]	The output line vector layer split at the antimeridian.

Pythoncode

Algorithm ID: `native:antimeridiansplit`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Geometry by expression

Updates existing geometries (or creates new geometries) for input features by use of a QGIS expression.

This allows complex geometry modifications which can utilize all the flexibility of the QGIS expression engine to manipulate and create geometries for output features.

For help with QGIS expression functions, see the inbuilt help available in the *expression builder*.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer
Output geometry type	`OUTPUT_GEOMETR`	[Aufzählung] Standard: 0	The output geometry strongly depends on the expression: for instance, if you create a buffer the geometry type has to be polygon. One of: • 0 — Polygon • 1 — Line • 2 — Point
Output geometry has z values	`WITH_Z`	[Boolean] Standard: falsch	Choose if the output geometry should include the Z dimension
Output geometry has m values	`WITH_M`	[Boolean] Standard: falsch	Choose if the output geometry should include the M dimension
Geometry expression	`EXPRESSION`	[Ausdruck] Default: ‚$geometry‘	Add the geometry expression you want to use. You can use the button to open the Expression Dialog. The dialog lists all the relevant expressions, together with their help and guide.
Modified geometry	`OUTPUT`	[Vektor: Geometrie] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Modified geometry	`OUTPUT`	[Vektor: Geometrie]	The output vector layer

Pythoncode

Algorithm ID: `native:geometrybyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Interpolate point on line

Creates a point geometry interpolated at a set distance along line or curve geometries.

Z and M values are linearly interpolated from existing values.

If a multipart geometry is encountered, only the first part is considered when calculating the substring.

If the specified distance is greater than the input feature's length, the resultant feature will have a null geometry.

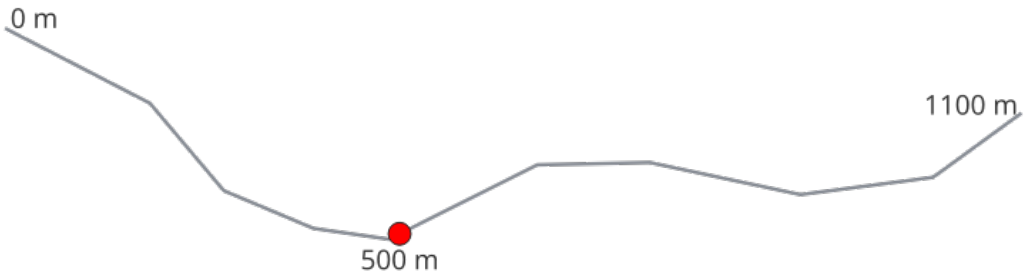


Abb. 24.118: Interpolierter Punkt in 500 m Entfernung vom Beginn der Strecke

Siehe auch

Points along geometry

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Abstand	`DISTANCE`	[numeric: double] Default: 0.0	Distance from the beginning of the line
Interpolated points	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Interpolated points	`OUTPUT`	[Vektor: Punkt]	The output point vector layer with features at a set distance along the line or polygon boundary

Pythoncode

Algorithm ID: `native:interpolatepoint`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Keep N biggest parts

Takes a layer with polygons or multipolygons and returns a new layer in which only the *n* largest polygons of each multipolygon feature are kept. If a feature has *n* or fewer parts, the feature will just be copied.

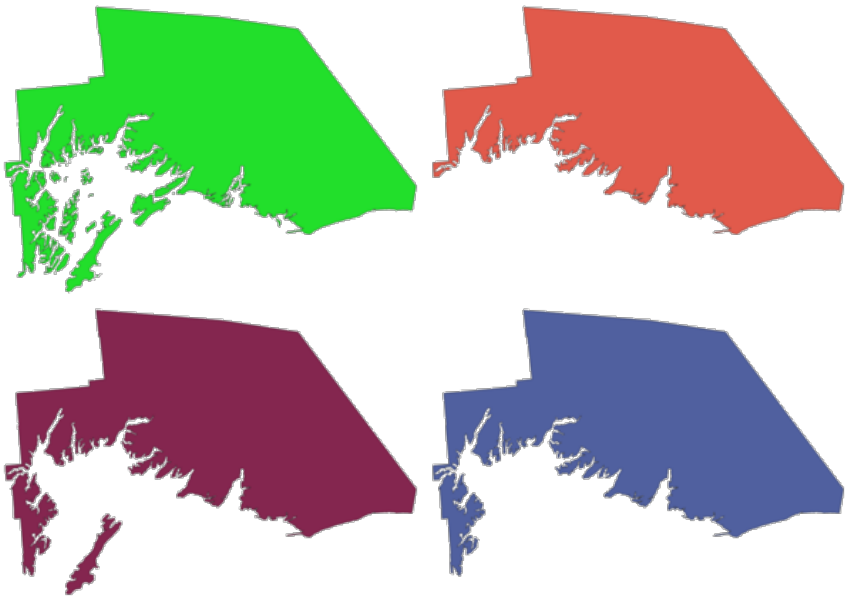


Abb. 24.119: Clockwise from top left: original multipart feature, one, two and three biggest parts kept

Parameter

Bezeichnung	Name	Typ	Beschreibung
Polygone	`INPUT`	[Vektor: Polygon]	Input polygon vector layer
Parts to keep	`PARTS`	[numeric: integer] Standard: 1	Number of parts to keep. If 1, only the biggest part of the feature will be kept.
Parts	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output polygon vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Parts	`OUTPUT`	[Vektor: Polygon]	The output polygon vector layer with the N biggest parts of each feature

Pythoncode

Algorithm ID: `native:keepnbiggestparts`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Line substring

Returns the portion of a line (or curve) which falls between the specified start and end distances (measured from the beginning of the line).

Z and M values are linearly interpolated from existing values.

If a multipart geometry is encountered, only the first part is considered when calculating the substring.

Abb. 24.120: Teilabschnitt mit einem Startabstand von 0 Metern und einem Endabstand von 250 Metern

Allows *features in-place modification* of line features

Siehe auch

Extend lines

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Start distance	`START_DISTANCE`	[numeric: double] Default: 0.0	Distance along the input line to the start point of the output feature
End distance	`END_DISTANCE`	[numeric: double] Standard: 1,0	Distance along the input line to the end point of the output feature
Substring	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Substring	`OUTPUT`	[vector: line]	The output line vector layer.

Pythoncode

Algorithm ID: `native:linesubstring`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Lines to polygons

Generates a polygon layer using as polygon rings the lines from an input line layer.

The attribute table of the output layer is the same as the one of the input layer.

Default menu: *Vector* ► *Geometry Tools*

Siehe auch

Polygons to lines, *Polygonize*, *Convert geometry type*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Polygone	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output polygon vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Polygone	`OUTPUT`	[Vektor: Polygon]	The output polygon vector layer.

Pythoncode

Algorithm ID: `qgis:linestopolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Merge lines

Joins all connected parts of MultiLineString geometries into single LineString geometries.

If any parts of the input MultiLineString geometries are not connected, the resultant geometry will be a MultiLineString containing any lines which could be merged and any non-connected line parts.

Allows *features in-place modification* of line features

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Zusammengeführt	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Zusammengeführt	`OUTPUT`	[vector: line]	The output (merged) line vector layer.

Pythoncode

Algorithm ID: `native:mergelines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Minimum bounding geometry

Creates geometries which enclose the features from an input layer. The features can be grouped by a field. The output layer will then contain one feature per group value with a geometry (MBB) that covers the geometries of the features with matching value.

The following enclosing geometry types are supported:

- bounding box (envelope)
- oriented rectangle
- circle
- convex hull

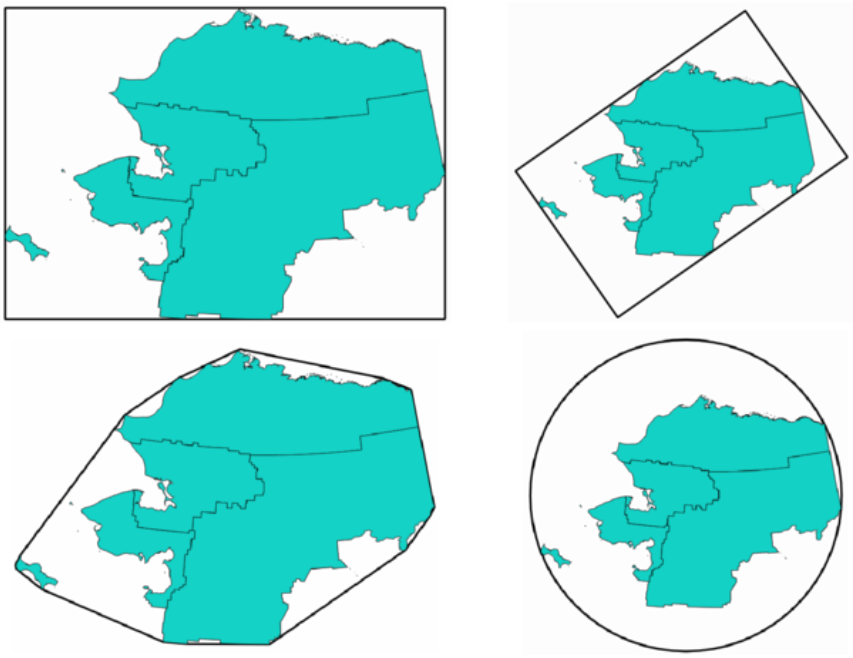


Abb. 24.121: Clockwise from top left: envelope, oriented rectangle, circle, convex hull

Siehe auch

Minimum enclosing circles

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Field Optional	`FIELD`	[tablefield: any]	Features can be grouped by a field. If set, this causes the output layer to contain one feature per grouped value with a minimal geometry covering only the features with matching values.
Geometry type	`TYPE`	[Aufzählung] Standard: 0	Enclosing geometry types. One of: • 0 — Envelope (Bounding Box) • 1 — Minimum Oriented Rectangle • 2 — Minimum Enclosing Circle • 3 — Convex Hull
Bounding geometry	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output polygon vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Bounding geometry	`OUTPUT`	[Vektor: Polygon]	The output (bounding) polygon vector layer. It has the fields from the input layer and, depending on the selected parameter `TYPE`, the output also has the attributes: • `width`, `height`, `area` and `perimeter` of the Envelope (Bounding Box) • `width`, `height`, `area`, `angle` and `perimeter` of the Minimum Oriented Rectangle • `radius` and `area` of the Minimum Enclosing Circle • `area` and `perimeter` of the Convex Hull

Pythoncode

Algorithm ID: `qgis:minimumboundinggeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Minimum enclosing circles

Calculates the minimum enclosing circles of the features in the input layer.

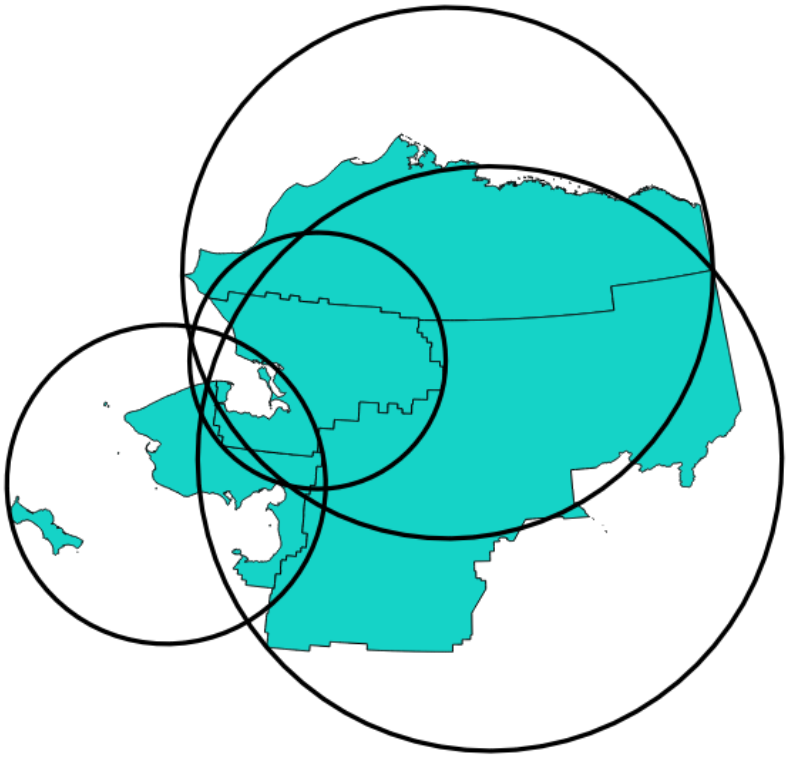


Abb. 24.122: Enclosing circles for each feature

Allows *features in-place modification* of polygon features

Siehe auch

Minimum bounding geometry

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Number of segments in circles	`SEGMENTS`	[numerisch: Ganzzahl] Default: 72	The number of segments used to approximate a circle. Minimum 8, maximum 100000.
Minimum enclosing circles	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output polygon vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Minimum enclosing circles	`OUTPUT`	[Vektor: Polygon]	The output polygon vector layer. Other than the input attributes, the output layer also contains the following fields: `area` and `radius` of the generated polygon.

Pythoncode

Algorithm ID: `native:minimumenclosingcircle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Multi-ring buffer (constant distance)

Computes multi-ring (*donut*) buffer for the features of the input layer, using a fixed or dynamic distance and number of rings.

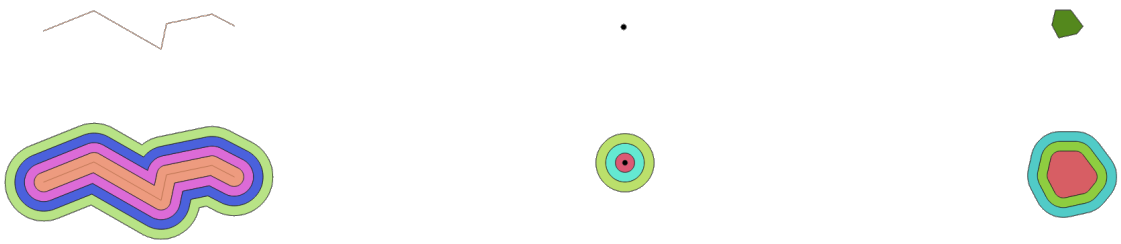


Abb. 24.123: Multi-ring buffer for a line, point and polygon layer

Allows *features in-place modification* of polygon features

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Puffer, *Variable distance buffer*, *Rectangles, ovals, diamonds*, *Single sided buffer*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Number of rings	`RINGS`	[numeric: integer] Standard: 1	The number of rings. It can be a unique value (same number of rings for all the features) or it can be taken from features data (the number of rings depends on feature values).
Distance between rings	`DISTANCE`	[numeric: double] Standard: 1,0	Distance between the rings. It can be a unique value (same distance for all the features) or it can be taken from features data (the distance depends on feature values).

Fortsetzung auf der nächsten Seite

Tab. 24.684 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Multi-ring buffer (constant distance)	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output polygon vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Multi-ring buffer (constant distance)	`OUTPUT`	[Vektor: Polygon]	The output polygon vector layer. Other than the input attributes, the output layer also contains the following fields: `ringId` and `distance` of the generated buffer polygon.

Pythoncode

Algorithm ID: `native:multiringconstantbuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Multipart to singleparts

Splits multipart features in the input layer into singlepart features.

The attributes of the output layer are the same as the original ones but divided into single features.

Abb. 24.124: Left the multipart source layer and right the single part output result

Allows *features in-place modification* of point, line, and polygon features

Default menu: *Vector* ► *Geometry Tools*

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Collect geometries, *Promote to multipart*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Single parts	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output polygon vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Single parts	`OUTPUT`	[wie Eingabe]	The output vector layer.

Pythoncode

Algorithm ID: `native:multiparttosingleparts`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Offset lines

Offsets lines by a specified distance. Positive distances will offset lines to the left, and negative distances will offset them to the right.

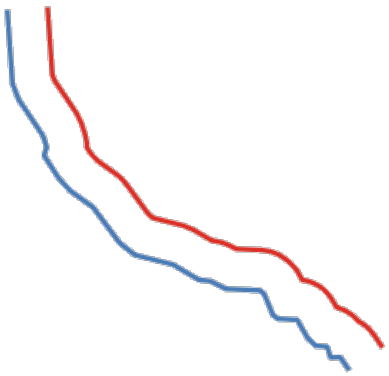


Abb. 24.125: In Blau der Quell-Layer, in Rot der Offset-Layer

Allows *features in-place modification* of line features

Siehe auch

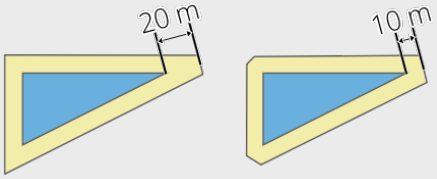
Array of offset (parallel) lines, *Translate*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Abstand	`DISTANCE`	[numeric: double] Default: 10.0	Offset distance. You can use the Data Defined button on the right to choose a field from which the radius will be calculated. This way you can have different radius for each feature (see *Variable distance buffer*).
Segmente	`SEGMENTS`	[numerisch: Ganzzahl] Default: 8	Steuert die Anzahl der Liniensegmente, die zur Annäherung an einen Viertelkreis beim Erstellen von abgerundeten Puffern verwendet werden sollen.
Verbindungsstil	`JOIN_STYLE`	[Aufzählung] Standard: 0	Legt fest, ob beim Puffern von Ecken in einer Linie runde, Gehrungs- oder abgeschrägte Verbindungen verwendet werden sollen. Mögliche Optionen: • 0 — Rund • 1 — Eckig • 2 — Abgefast Abb. 24.126: Runder, eckiger und abgefaster Verbindungsstil

Fortsetzung auf der nächsten Seite

Tab. 24.688 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Eckgrenze	`MITER_LIMIT`	[numerisch: Double] Default: 2.0	Sets the maximum distance from the offset geometry to use when creating a mitered join as a factor of the offset distance (only applicable for miter join styles). Minimum: 1.0  Abb. 24.127: A 10m buffer with a limit of 2 and a 10m buffer with a limit of 1
Offset	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output (offset) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Offset	`OUTPUT`	[vector: line]	Output (offset) line layer

Pythoncode

Algorithm ID: `native:offsetline`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Orientiertes minimales Begrenzungsrechteck

Calculates the minimum area rotated rectangle for each feature in the input layer.

Bemerkung

For singlepart point features, the output corresponds to the bounding box of the geometry.

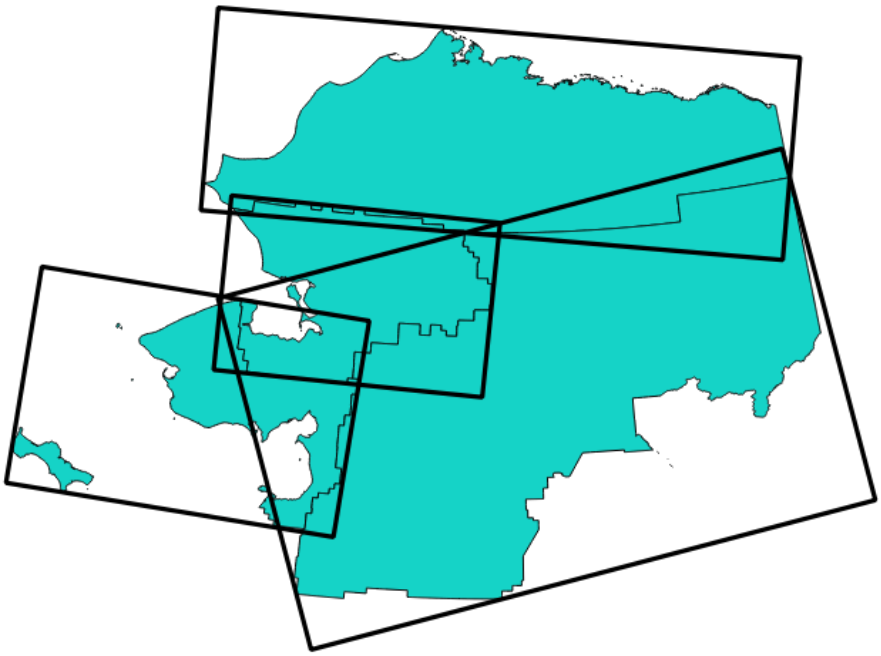


Abb. 24.128: Orientiertes minimales Begrenzungsrechteck

Allows *features in-place modification* of polygon features

Siehe auch

Minimum bounding geometry

Fortsetzung auf der nächsten Seite

Tab. 24.690 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Bounding boxes	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output polygon vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Bounding boxes	`OUTPUT`	[Vektor: Polygon]	The output polygon vector layer. Other than the input attributes, the output layer also contains the following fields: `width`, `height`, `angle`, `area` and `perimeter` of the generated polygon.

Pythoncode

Algorithm ID: `native:orientedminimumboundingbox`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Orthogonalize

Attempts to orthogonalize the geometries of the input line or polygon layer. This process shifts the vertices in the geometries to try to make every angle in the geometry either a right angle or a straight line.

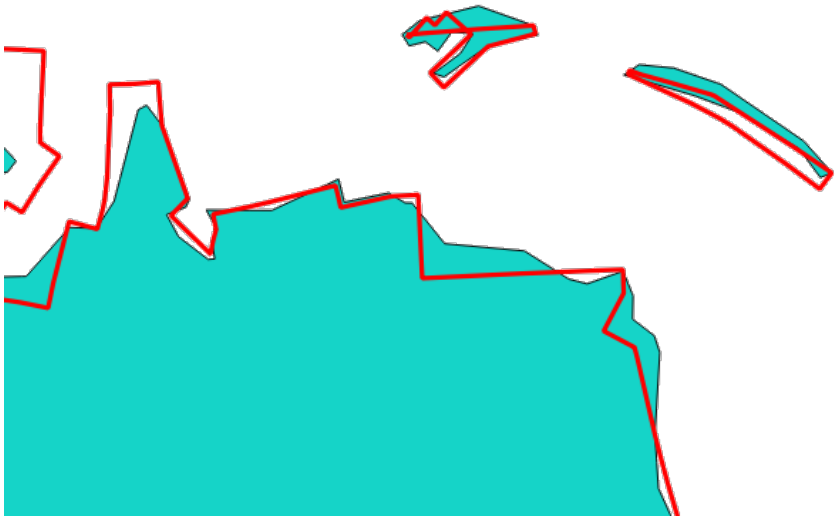


Abb. 24.129: In blue the source layer and in the red orthogonalized result

Allows *features in-place modification* of line and polygon features

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Maximum angle tolerance (degrees)	`ANGLE_TOLERANC`	[numerisch: Double] Default: 15.0	Specify the maximum deviation from a right angle or straight line a vertex can have for it to be adjusted. Smaller tolerances mean that only vertices which are already closer to right angles will be adjusted, and larger tolerances mean that vertices which deviate further from right angles will also be adjusted.
Maximum algorithm iterations	`MAX_ITERATIONS`	[numerisch: Ganzzahl] Standard: 1000	Setting a larger number for the maximum number of iterations will result in a more orthogonal geometry at the cost of extra processing time.
Orthogonalized	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output polygon vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Orthogonalized	`OUTPUT`	[wie Eingabe]	The output polygon vector layer with adjusted angles.

Pythoncode

Algorithm ID: `native:orthogonalize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Point on Surface

For each feature of the input layer, returns a point that is guaranteed to lie on the surface of the feature geometry.

Erlaubt auch die *Insitu-Bearbeitung* von Punkt-Objekten

Warnung

This algorithm may drop existing primary keys or FID values and regenerate them in output layers, depending on the input parameters.

Siehe auch

Centroids

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Create point on surface for each part	`ANGLE_TOLERANC`	[boolean]	If checked, a point will be created for each part of the geometry.

Fortsetzung auf der nächsten Seite

Tab. 24.694 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Point	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output point vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Point	`OUTPUT`	[Vektor: Punkt]	The output point vector layer.

Pythoncode

Algorithm ID: `native:pointonsurface`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Points along geometry

Creates points at regular intervals along line or polygon geometries. Created points will have new attributes added for the distance along the geometry and the angle of the line at the point.

An optional start and end offset can be specified, which controls how far from the start and end of the geometry the points should be created.

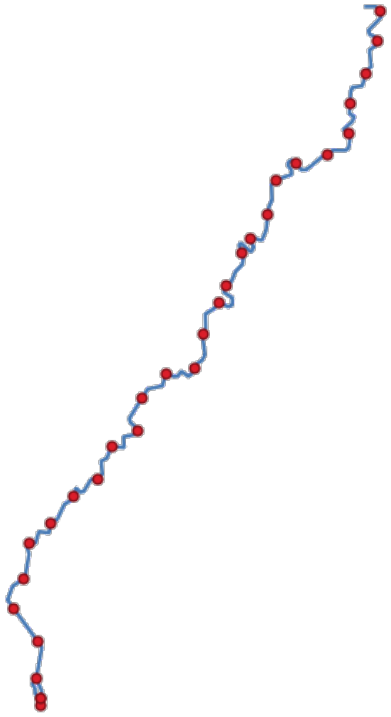


Abb. 24.130: Points created along the source line layer

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Interpolate point on line

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Abstand	`DISTANCE`	[numeric: double] Standard: 1,0	Distance between two consecutive points along the line
Start offset	`START_OFFSET`	[numeric: double] Default: 0.0	Distance from the beginning of the input line, representing the position of the first point.
End offset	`END_OFFSET`	[numeric: double] Default: 0.0	Distance from the end of the input line, representing the position beyond which no point feature shoud be created.

Fortsetzung auf der nächsten Seite

Tab. 24.696 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Interpolated points	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Interpolated points	`OUTPUT`	[Vektor: Punkt]	Point vector layer with features placed along lines or polygon boundaries of the input layer. Other than the input attributes on every related point, the output layer also contains the following fields: `distance` and `angle` of the generated point.

Pythoncode

Algorithm ID: `native:pointsalonglines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Points displacement

Given a distance of proximity, identifies nearby point features and radially distributes them over a circle whose center represents their barycenter. A convenient tool to scatter overlaid features.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Minimum distance to other points	`PROXIMITY`	[numerisch: Double] Standard: 1,0	Distance below which point features are considered close. Close features are distributed altogether.
Displacement distance	`DISTANCE`	[numerisch: Double] Standard: 1,0	Radius of the circle on which close features are placed

Fortsetzung auf der nächsten Seite

Tab. 24.698 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Horizontal distribution for two point case	`HORIZONTAL`	[Boolean] Standard: falsch	When only two points are identified as close, aligns them horizontally on the circle instead of vertically.
Displaced	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Displaced	`OUTPUT`	[Vektor: Punkt]	Output point vector layer

Pythoncode

Algorithm ID: `qgis:pointsdisplacement`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Pole of inaccessibility

Calculates the pole of inaccessibility for a polygon layer, which is the most distant internal point from the boundary of the surface.

This algorithm uses the ‚polylabel' algorithm (Vladimir Agafonkin, 2016), which is an iterative approach guaranteed to find the true pole of inaccessibility within a specified tolerance. A more precise tolerance (lower value) requires more iterations and will take longer to calculate.

Abb. 24.131: Pole of inaccessibility

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Eingabe-Vektorlayer
Tolerance	`TOLERANCE`	[numerisch: Double] Standard: 1,0	Set the tolerance for the calculation
Point	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output polygon vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Point	`OUTPUT`	[Vektor: Punkt]	The output point vector layer. It will contain a new attribute with the distance `dist_pole` from the calculated pole to the polygon boundary.

Pythoncode

Algorithm ID: `native:poleofinaccessibility`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Polygonize

Creates a polygon layer whose features boundaries are generated from a line layer of **closed** features.

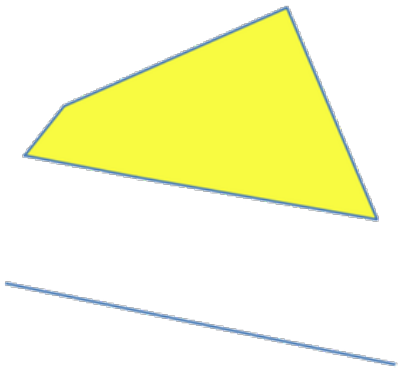


Abb. 24.132: The yellow polygons generated from the closed lines

Bemerkung

The line layer must have closed shapes in order to be transformed into a polygon.

Siehe auch

Polygons to lines, *Lines to polygons*, *Convert geometry type*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Keep fields from the input layer	`KEEP_FIELDS`	[Boolean] Standard: falsch	Check to keep the fields (only the table structure, not the values) of the input layer

Fortsetzung auf der nächsten Seite

Tab. 24.702 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Polygons from lines	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output polygon vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Polygons from lines	`OUTPUT`	[Vektor: Polygon]	The output polygon vector layer from lines

Pythoncode

Algorithm ID: `native:polygonize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Polygons to lines

Takes a polygon layer and creates a line layer, with lines representing the boundaries of the polygons in the input layer.

The attribute table of the output layer is the same as the one of the input layer.

Abb. 24.133: Black lines as the result of the algorithm

Default menu: *Vector ► Geometry Tools*

Siehe auch

Lines to polygons, *Polygonize*, *Convert geometry type*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Input polygon vector layer
Linien	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Linien	`OUTPUT`	[vector: line]	The output line vector layer from polygons

Pythoncode

Algorithm ID: `native:polygonstolines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Project points (Cartesian)

Projects point geometries by a specified distance and bearing (azimuth).

Erlaubt auch die *Insitu-Bearbeitung* von Punkt-Objekten

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Bearing (degrees from North)	`BEARING`	[numeric: double] Default: 0.0	Clockwise angle starting from North, in degree (°) unit
Abstand	`DISTANCE`	[numeric: double] Standard: 1,0	Distance to offset geometries, in layer units
Projected	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output point vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Projected	`OUTPUT`	[Vektor: Punkt]	The output (projected) point vector layer

Pythoncode

Algorithm ID: `native:projectpointcartesian`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Promote to multipart

Takes a vector layer with singlepart geometries and generates a new one in which all geometries are multipart.

Input features which are already multipart features will remain unchanged.

This algorithm can be used to force geometries to multipart types in order to be compatible with data providers that require multipart features.

Allows *features in-place modification* of point, line, and polygon features

Siehe auch

Aggregate, *Collect geometries*, *Convert to strictly multipart*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Multiparts	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output multipart vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Multiparts	`OUTPUT`	[wie Eingabe]	The output multipart vector layer

Pythoncode

Algorithm ID: `native:promotetomulti`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Rectangles, ovals, diamonds

Creates a buffer area with a rectangle, oval or diamond shape for each feature of the input point layer.

The shape parameters can be fixed for all features or dynamic using a field or an expression.

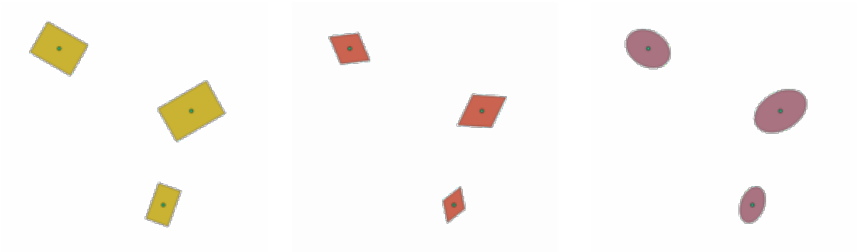


Abb. 24.134: Different buffer shapes with dynamic parameters

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Buffer shape	`SHAPE`	[Aufzählung]	The shape to use. One of: • 0 — Rectangles • 1 — Ovals • 2 — Diamonds
Width	`WIDTH`	[numeric: double] Standard: 1,0	Width of the buffer shape
Height	`HEIGHT`	[numeric: double] Standard: 1,0	Height of the buffer shape
Rotation Optional	`ROTATION`	[numeric: double] Default: 0.0	Rotation of the buffer shape
Number of segments	`SEGMENTS`	[numerisch: Ganzzahl] Default: 36	Number of segments for a full circle (*Ovals* shape)
Ergebnis	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ergebnis	`OUTPUT`	[Vektor: Polygon]	The output vector layer (with the buffer shapes)

Pythoncode

Algorithm ID: `native:rectanglesovalsdiamonds`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Remove duplicate vertices

Removes duplicate vertices from features, wherever removing the vertices does not result in a degenerate geometry.

The tolerance parameter specifies the tolerance for coordinates when determining whether vertices are identical.

By default, Z values are not considered when detecting duplicate vertices. E.g. two vertices with the same X and Y coordinate but different Z values will still be considered duplicate and one will be removed. If the *Use Z Value* parameter is true, then the Z values are also tested and vertices with the same X and Y but different Z will be maintained.

Allows *features in-place modification* of point, line, and polygon features

Bemerkung

Duplicate vertices are not tested between different parts of a multipart geometry, e.g. a multipoint geometry with overlapping points will not be changed by this method.

Siehe auch

Extract vertices, *Extract specific vertices*, *Delete duplicate geometries*, *Delete duplicated vertices*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Tolerance	`TOLERANCE`	[numeric: double] Default: 0.000001	Vertices closer than the specified distance are considered duplicates

Fortsetzung auf der nächsten Seite

Tab. 24.712 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Use Z value	`USE_Z_VALUE`	[boolean] Standard: falsch	If the *Use Z Value* parameter is true, then the Z values are also tested and vertices with the same X and Y but different Z will be maintained.
Cleaned	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Cleaned	`OUTPUT`	[wie Eingabe]	The output vector layer (without duplicate vertices)

Pythoncode

Algorithm ID: `native:removeduplicatevertices`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Remove null geometries

Removes any features which do not have a geometry from a vector layer. All other features will be copied unchanged.

The features with null geometries can be saved to a separate layer.

If *Also remove empty geometries* is checked, the algorithm removes features whose geometries have no coordinates, i.e., geometries that are empty. In that case, also the null output will reflect this option, containing both null and empty geometries.

Siehe auch

Delete duplicate geometries

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Input vector layer (with non-NULL geometries)
Also remove empty geometries	`REMOVE_EMPTY`	[Boolean]	
Non null geometries	`OUTPUT` Optional	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer for the non-NULL (and non-empty) geometries. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Null geometries Optional	`NULL_OUTPUT`	[wie Eingabe] Standard: `[Ausgabe überspringen]`	Specify the output vector layer for the NULL (and empty) geometries. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Null geometries	`NULL_OUTPUT`	[wie Eingabe]	Output vector layer (for NULL and, if chosen, empty geometries)
Non null geometries	`OUTPUT`	[wie Eingabe]	The output vector layer (without NULL and, if chosen, empty geometries)

Pythoncode

Algorithm ID: `native:removenullgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Reverse line direction

Inverts the direction of a line layer.

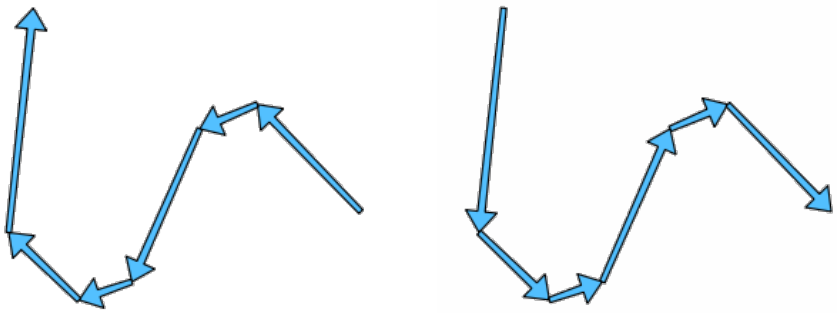


Abb. 24.135: Before and after the direction inversion

Allows *features in-place modification* of line features

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Reversed	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Reversed	`OUTPUT`	[vector: line]	The output line vector layer (with reversed lines)

Pythoncode

Algorithm ID: `native:reverselinedirection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Rotate

Rotates feature geometries by the specified angle clockwise. The rotation occurs around each feature's centroid, or optionally around a unique preset point.

Allows *features in-place modification* of point, line, and polygon features

Siehe auch

Translate, *Swap X and Y coordinates*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Rotation (degrees clockwise)	`ANGLE`	[numeric: double] Default: 0.0	Angle of the rotation in degrees
Rotation anchor point (x, y) Optional	`ANCHOR`	[coordinates] Standard: Nicht gesetzt	X,Y coordinates of the point to rotate the features around. If not set the rotation occurs around each feature's centroid.
Rotated	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer (with rotated geometries). *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Rotated	`OUTPUT`	[wie Eingabe]	The output vector layer with rotated geometries

Pythoncode

Algorithm ID: `native:rotatefeatures`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Roundness

Calculates the roundness of each feature and stores it as a new field `roundness`. The input vector layer must contain polygons.

The roundness of a polygon is defined as $4\pi \times$ polygon area / perimeter^2. The roundness value varies between 0 and 1. A perfect circle has a roundness of 1, while a completely flat polygon has a roundness of 0.

Bemerkung

The algorithm returns NULL for multipart polygon features.

Allows *features in-place modification* of polygon features

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Eingabe-Vektorlayer
Roundness	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output vector layer (with roundness field). *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Roundness	`OUTPUT`	[wie Eingabe]	The output vector layer with `roundness` value in a field

Pythoncode

Algorithm ID: `native:roundness`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Segmentize by maximum angle

Segmentizes a geometry by converting curved sections to linear sections.

The segmentization is performed by specifying the maximum allowed radius angle between vertices on the straightened geometry (e.g the angle of the arc created from the original arc center to consecutive output vertices on the linearized geometry). Non-curved geometries will be retained without change.

Siehe auch

Segmentize by maximum distance, *Simplify*, *Smooth*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Maximum angle between vertices (degrees)	`ANGLE`	[numeric: double] Default: 5.0	Maximum allowed radius angle between vertices on the straightened geometry
Segmentized	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer (with segmentized geometries). *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Segmentized	`OUTPUT`	[wie Eingabe]	The output vector layer with segmentized geometries

Pythoncode

Algorithm ID: `native:segmentizebymaxangle`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Segmentize by maximum distance

Segmentizes a geometry by converting curved sections to linear sections.

The segmentization is performed by specifying the maximum allowed offset distance between the original curve and the segmentized representation. Non-curved geometries will be retained without change.

Siehe auch

Segmentize by maximum angle, *Simplify*, *Smooth*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Maximum offset distance	`DISTANCE`	[numeric: double] Standard: 1,0	Maximum allowed offset distance between the original curve and the segmentized representation, in the layer units.
Segmentized	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer (with segmentized geometries). *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Segmentized	`OUTPUT`	[wie Eingabe]	The output vector layer with segmentized geometries

Pythoncode

Algorithm ID: `native:segmentizebymaxdistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

M-Wert einstellen

Sets the M value for geometries in a layer.

If M values already exist in the layer, they will be overwritten with the new value. If no M values exist, the geometry will be upgraded to include M values and the specified value used as the initial M value for all geometries.

Allows *features in-place modification* of point, line, and polygon features with M enabled

Tipp

Use the Identify Features button to check the added M value: the results are available in the *Identify Results* dialog.

Siehe auch

Set M value from raster, *Set Z value*, *Drop M/Z values*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
M Value	`M_VALUE`	[numeric: double] Default: 0.0	M value to assign to the feature geometries

Fortsetzung auf der nächsten Seite

Tab. 24.726 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
M Added	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
M Added	`OUTPUT`	[wie Eingabe]	The output vector layer (with M values assigned to the geometries)

Pythoncode

Algorithm ID: `native:setmvalue`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Set M value from raster

Uses values sampled from a band within a raster layer to set the M value for every overlapping vertex in the feature geometry. The raster values can optionally be scaled by a preset amount.

If M values already exist in the layer, they will be overwritten with the new value. If no M values exist, the geometry will be upgraded to include M values.

Allows *features in-place modification* of point, line, and polygon features with M enabled

Siehe auch

Drape (set Z value from raster), *M-Wert einstellen*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Raster layer	`RASTER`	[Raster]	Raster layer with M values
Band number	`BAND`	[raster band] Standard: 1	The raster band from which the M values are taken
Value for NoData or non-intersecting vertices	`NODATA`	[numeric: double] Default: 0.0	Value to use in case the vertex does not intersect (a valid pixel of) the raster
Scale factor	`SCALE`	[numeric: double] Standard: 1,0	Scaling value: the band values are multiplied by this value.
Offset	`OFFSET`	[numeric: double] Default: 0.0	Offset value: it is algebraically added to the band values after applying the „Scale factor“.
Updated	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer (with updated M values). *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Updated	`OUTPUT`	[wie Eingabe]	The output vector layer (with updated M values)

Pythoncode

Algorithm ID: `native:setmfromraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Set Z value

Sets the Z value for geometries in a layer.

If Z values already exist in the layer, they will be overwritten with the new value. If no Z values exist, the geometry will be upgraded to include Z values and the specified value used as the initial Z value for all geometries.

Allows *features in-place modification* of point, line, and polygon features with Z enabled

Tipp

Use the Identify Features button to check the added Z value: the results are available in the *Identify Results* dialog.

Siehe auch

Drape (set Z value from raster), *M-Wert einstellen*, *Drop M/Z values*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Z Value	`Z_VALUE`	[numeric: double] Default: 0.0	Z value to assign to the feature geometries
Z Added	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Z Added	`OUTPUT`	[wie Eingabe]	The output vector layer (with Z values assigned)

Pythoncode

Algorithm ID: `native:setzvalue`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Simplify

Simplifies the geometries in a line or polygon layer. It creates a new layer with the same features as the ones in the input layer, but with geometries containing a lower number of vertices.

The algorithm gives a choice of simplification methods, including distance based (the „Douglas-Peucker“ algorithm), area based („Visvalingam“ algorithm) and snapping geometries to grid.

Abb. 24.136: Von links nach rechts: Quell-Layer und zunehmende Vereinfachungs-Toleranzen

Allows *features in-place modification* of line and polygon features

Default menu: *Vector* ► *Geometry Tools*

Siehe auch

Simplify coverage, *Smooth*, *Densify by count*, *Densify by interval*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Simplification method	`METHOD`	[Aufzählung] Standard: 0	Simplification method. One of: • 0 — Distance (Douglas-Peucker) • 1 — Snap to grid • 2 — Area (Visvalingam)

Fortsetzung auf der nächsten Seite

Tab. 24.732 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Tolerance	`TOLERANCE`	[numeric: double] Standard: 1,0	Threshold tolerance (in units of the layer): if the distance between two nodes is smaller than the tolerance value, the segment will be simplified and vertices will be removed.
Simplified	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output (simplified) vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Simplified	`OUTPUT`	[wie Eingabe]	The output (simplified) vector layer

Pythoncode

Algorithm ID: `native:simplifygeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Single sided buffer

Computes a buffer on lines by a specified distance on one side of the line only.

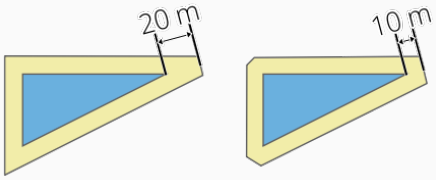
Buffer always results in a polygon layer.

Abb. 24.137: Linke und rechte Pufferung der gleichen Linie

Siehe auch

Puffer

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Abstand	`DISTANCE`	[numeric: double] Default: 10.0	Buffer distance.
Side	`SIDE`	[Aufzählung] Standard: 0	Which side to create the buffer on. One of: • 0 – Left • 1 – Right
Segmente	`SEGMENTS`	[numerisch: Ganzzahl] Default: 8	Steuert die Anzahl der Liniensegmente, die zur Annäherung an einen Viertelkreis beim Erstellen von abgerundeten Puffern verwendet werden sollen.
Verbindungsstil	`JOIN_STYLE`	[Aufzählung] Standard: 0	Legt fest, ob beim Puffern von Ecken in einer Linie runde, Gehrungs- oder abgeschrägte Verbindungen verwendet werden sollen. Mögliche Optionen: • 0 — Rund • 1 — Eckig • 2 — Abgefast Abb. 24.138: Runder, eckiger und abgefaster Verbindungsstil
Eckgrenze	`MITER_LIMIT`	[numerisch: Double] Default: 2.0	Sets the maximum distance from the offset geometry to use when creating a mitered join as a factor of the offset distance (only applicable for miter join styles). Minimum: 1.0  Abb. 24.139: A 10m buffer with a limit of 2 and a 10m buffer with a limit of 1

Fortsetzung auf der nächsten Seite

Tab. 24.734 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Buffer	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output (buffer) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Buffer	`OUTPUT`	[Vektor: Polygon]	Output (buffer) polygon layer

Pythoncode

Algorithm ID: `native:singlesidedbuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Smooth

Smooths the geometries in a line or polygon layer by adding more **vertices and corners** to the feature geometries.

The iterations parameter dictates how many smoothing iterations will be applied to each geometry. A higher number of iterations results in smoother geometries with the cost of greater number of nodes in the geometries.

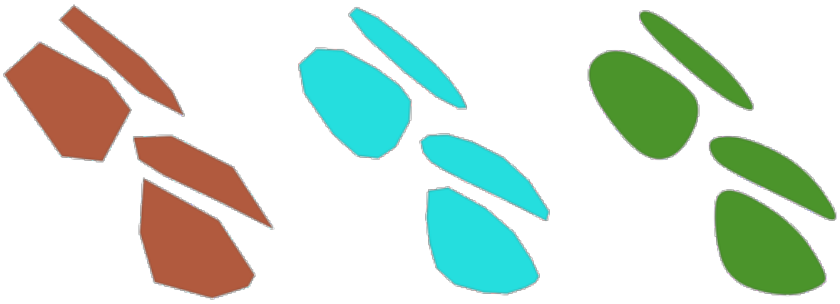


Abb. 24.140: Mit zunehmender Anzahl von Iterationen werden die Geometrien glatter

The offset parameter controls how „tightly“ the smoothed geometries follow the original geometries. Smaller values results in a tighter fit, and larger values will create a looser fit.

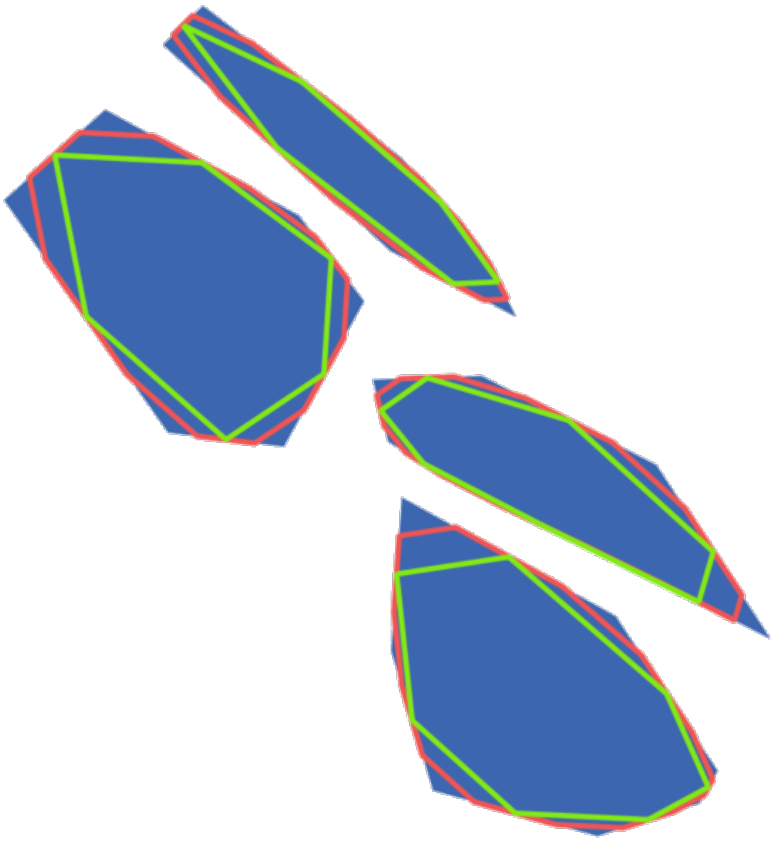


Abb. 24.141: Blue: the input layer. Offset 0.25 gives the red line, while offset 0.50 gives the green line.

The maximum angle parameter can be used to prevent smoothing of nodes with large angles. Any node where the angle of the segments to either side is larger than this will not be smoothed. For example, setting the maximum angle to 90 degrees or lower would preserve right angles in the geometry.

Allows *features in-place modification* of line and polygon features

Siehe auch

Simplify, *Simplify coverage*, *Densify by count*, *Densify by interval*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Input line or polygon vector layer
Iterations	`ITERATIONS`	[numeric: integer] Standard: 1	Increasing the number of iterations will give smoother geometries (and more vertices).
Offset	`OFFSET`	[numeric: double] Default: 0.25	Increasing values will *move* the smoothed lines / boundaries further away from the input lines / boundaries.

Fortsetzung auf der nächsten Seite

Tab. 24.736 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Maximum node angle to smooth	`MAX_ANGLE`	[numeric: double] Default: 180.0	Every node below this value will be smoothed
Smoothed	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output (smoothed) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Smoothed	`OUTPUT`	[wie Eingabe]	Output (smoothed) vector layer

Pythoncode

Algorithm ID: `native:smoothgeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Snap geometries to layer

Snaps the geometries in a layer either to the geometries from another layer, or to geometries within the same layer.

Matching is done based on a tolerance distance, and vertices will be inserted or removed as required to make the geometries match the reference geometries.

Allows *features in-place modification* of point, line, and polygon features

Siehe auch

Snap points to grid

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Referenz-Layer	`REFERENCE_LAYE`	[Vektor: Geometrie]	Vector layer to snap to
Tolerance	`TOLERANCE`	[numerisch: Double] Default: 10.0	Control how close input vertices need to be to the reference layer geometries before they are snapped.

Fortsetzung auf der nächsten Seite

Tab. 24.738 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Behavior	`BEHAVIOR`	[Aufzählung] Standard: 0	Snapping can be done to an existing node or a segment (its closest point to the vertex to move). Available snapping options: • 0 — Prefer aligning nodes, insert extra vertices where required Prefer to snap to nodes, even when a segment may be closer than a node. New nodes will be inserted to make geometries follow each other exactly when inside allowable tolerance. • 1 — Prefer closest point, insert extra vertices where required Snap to closest point, regardless of it is a node or a segment. New nodes will be inserted to make geometries follow each other exactly when inside allowable tolerance. • 2 — Prefer aligning nodes, don't insert new vertices Prefer to snap to nodes, even when a segment may be closer than a node. No new nodes will be inserted. • 3 — Prefer closest point, don't insert new vertices Snap to closest point, regardless of it is a node or a segment. No new nodes will be inserted. • 4 — Move end points only, prefer aligning nodes Only snap start/end points of lines (point features will also be snapped, polygon features will not be modified), prefer to snap to nodes. • 5 — Move end points only, prefer closest point Only snap start/end points of lines (point features will also be snapped, polygon features will not be modified), snap to closest point • 6 — Snap end points to end points only Only snap the start/end points of lines to other start/end points of lines • 7 — Snap to anchor nodes (single layer only)
Snapped geometry	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output (snapped) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Snapped geometry	`OUTPUT`	[wie Eingabe]	Output (snapped) vector layer

Pythoncode

Algorithm ID: `native:snapgeometries`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Snap points to grid

Modifies the coordinates of geometries in a vector layer, so that all points or vertices are snapped to the closest point of a grid.

If the snapped geometry cannot be calculated (or is totally collapsed) the feature's geometry will be cleared.

Snapping can be performed on the X, Y, Z or M axis. A grid spacing of 0 for any axis will disable snapping for that axis.

Allows *features in-place modification* of point, line, and polygon features

Warnung

Snapping to grid may generate an invalid geometry in some corner cases.

Siehe auch

Snap geometries to layer

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
X Grid Spacing	`HSPACING`	[numeric: double] Standard: 1,0	Grid spacing on the X axis
Y Grid Spacing	`VSPACING`	[numeric: double] Standard: 1,0	Grid spacing on the Y axis

Fortsetzung auf der nächsten Seite

Tab. 24.740 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Z Grid Spacing	`ZSPACING`	[numeric: double] Default: 0.0	Grid spacing on the Z axis
M Grid Spacing	`MSPACING`	[numeric: double] Default: 0.0	Grid spacing on the M axis
Snapped	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output (snapped) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Snapped	`OUTPUT`	[wie Eingabe]	Output (snapped) vector layer

Pythoncode

Algorithm ID: `native:snappointstogrid`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Split lines by maximum length

Takes a line (or curve) layer and splits each feature into multiple parts, where each part is of a specified maximum length. Z and M values at the start and end of the new line substrings are linearly interpolated from existing values.

Allows *features in-place modification* of line features

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	The input line vector layer
Maximum line length	`LENGTH`	[numeric: double] Default: 10.0	The maximum length of a line in the output.
Split	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Split	`OUTPUT`	[vector: line]	The new line vector layer. The length of the feature geometries is less than or equal to the length specified in the LENGTH parameter. An additional field containing the `order` will be added.

Pythoncode

Algorithm ID: `native:splitlinesbylength`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Subdivide

Subdivides the geometry. The returned geometry will be a collection containing subdivided parts from the original geometry, where no part has more than the specified maximum number of nodes.

This is useful for dividing a complex geometry into less complex parts, easier to spatially index and faster to perform spatial operations. Curved geometries will be segmentized before subdivision.

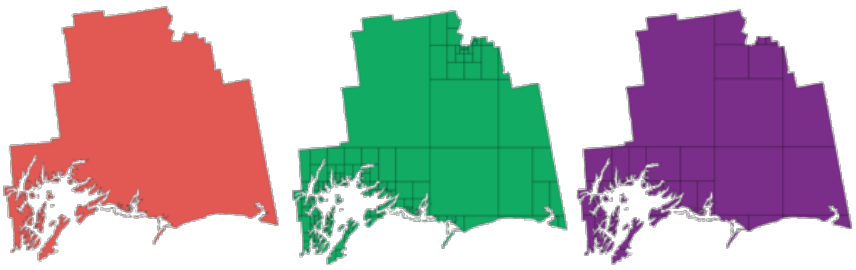


Abb. 24.142: Left the input layer, middle maximum nodes value is 100 and right maximum value is 200

Allows *features in-place modification* of point, line, and polygon features

Bemerkung

Subdividing a geometry can generate geometry parts that may not be valid and may contain self-intersections.

Siehe auch

Explode lines, *Line substring*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	The input vector layer
Maximum nodes in parts	`MAX_NODES`	[numeric: integer] Default: 256	Maximum number of vertices each new geometry part is allowed to have. Fewer *sub-parts* for higher values.
Subdivided	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output (subdivided) vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Subdivided	`OUTPUT`	[wie Eingabe]	Output vector layer

Pythoncode

Algorithm ID: `native:subdivide`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Swap X and Y coordinates

Switches the X and Y coordinate values in input geometries.

It can be used to repair geometries which have accidentally had their latitude and longitude values reversed.

Allows *features in-place modification* of point, line, and polygon features

Siehe auch

Translate, *Rotate*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	The input vector layer
Swapped	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Swapped	`OUTPUT`	[wie Eingabe]	Output (swapped) vector layer

Pythoncode

Algorithm ID: `native:swapxy`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Tapered buffers

Creates tapered buffer along line geometries, using a specified start and end buffer diameter.

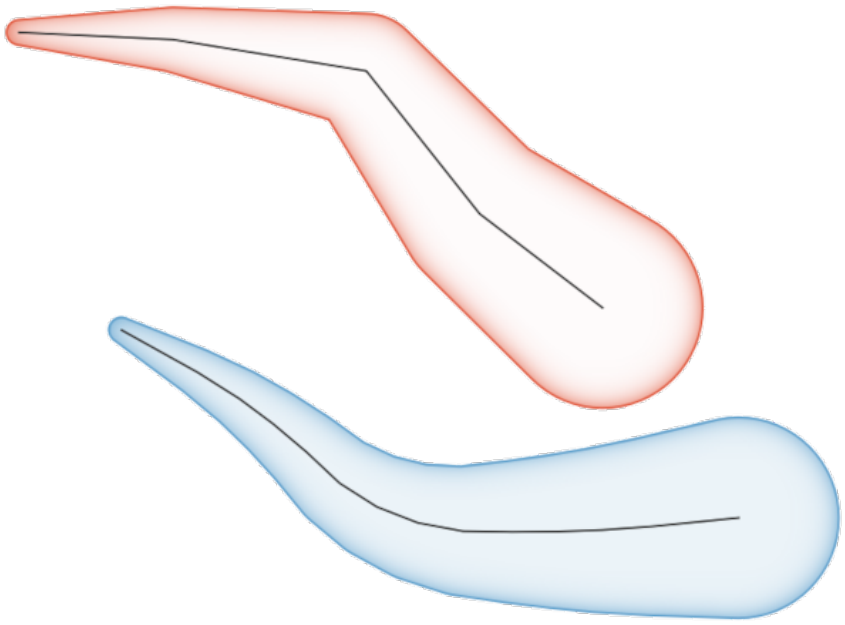


Abb. 24.143: Tapered buffer example

Siehe auch

Variable width buffer (by M value), *Puffer*, *Create wedge buffers*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Start width	`START_WIDTH`	[numeric: double] Default: 0.0	Represents the radius of the buffer applied at the start point of the line feature
End width	`END_WIDTH`	[numeric: double] Default: 0.0	Represents the radius of the buffer applied at the end point of the line feature.
Segmente	`SEGMENTS`	[numeric: integer] Default: 16	Steuert die Anzahl der Liniensegmente, die zur Annäherung an einen Viertelkreis beim Erstellen von abgerundeten Puffern verwendet werden sollen.
Gepuffert	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output (buffer) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Gepuffert	`OUTPUT`	[Vektor: Polygon]	Output (buffer) polygon layer

Pythoncode

Algorithm ID: `native:taperedbuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Tessellate

Tessellates a polygon geometry layer, dividing the geometries into triangular components.

The output layer consists of multipolygon geometries for each input feature, with each multipolygon consisting of multiple triangle component polygons.

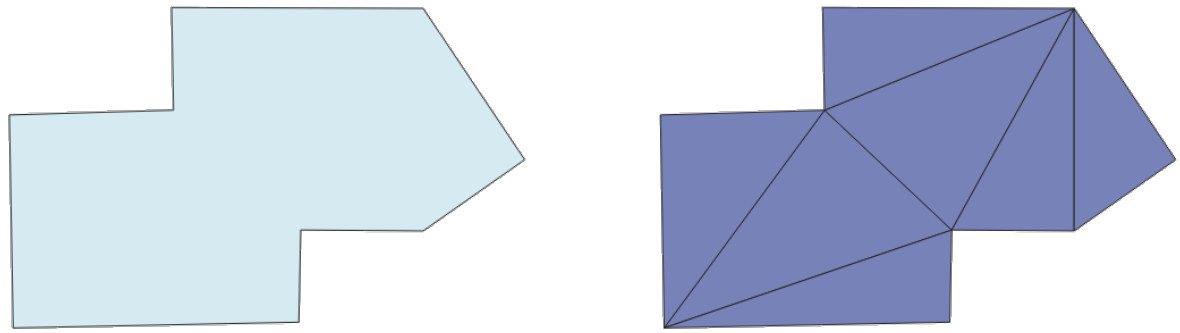


Abb. 24.144: Tessellated polygon (right)

Allows *features in-place modification* of polygon features

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Polygon]	Input polygon vector layer
Tesselated	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Tesselated	`OUTPUT`	[Vektor: Polygon]	Output multipolygonZ layer

Pythoncode

Algorithm ID: `3d:tessellate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Transect

Creates transects on vertices for (multi)linestring.

A transect is a line oriented from an angle (by default perpendicular) to the input polylines (at vertices).

Field(s) from feature(s) are returned in the transect with these new fields:

- TR_FID: ID of the original feature
- TR_ID: ID of the transect. Each transect have an unique ID
- TR_SEGMENT: ID of the segment of the linestring
- TR_ANGLE: Angle in degrees from the original line at the vertex
- TR_LENGTH: Total length of the transect returned
- TR_ORIENT: Side of the transect (only on the left or right of the line, or both side)

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

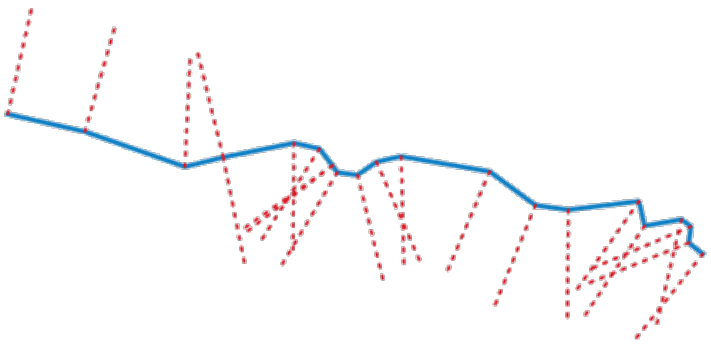


Abb. 24.145: Dashed red lines represent the transect of the input line layer

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Length of the transect	`LENGTH`	[numeric: double] Default: 5.0	Length in map unit of the transect
Angle in degrees from the original line at the vertices	`ANGLE`	[numeric: double] Default: 90.0	Change the angle of the transect
Side to create the transect	`SIDE`	[Aufzählung]	Choose the side of the transect. Available options are: • 0 — Left • 1 — Right • 2 — Both
Transect	`OUTPUT`	[vector: line] Standard: `[Create temporary layer]`	Specify the output line layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Transect	`OUTPUT`	[vector: line]	Output line layer

Pythoncode

Algorithm ID: `native:transect`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Translate

Moves the geometries within a layer, by offsetting with a predefined X and Y displacement.

Z and M values present in the geometry can also be translated.

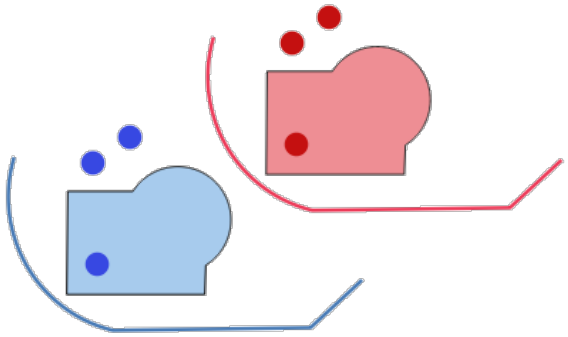


Abb. 24.146: Features in red represent the translated geometry of the input (blue) layer

Allows *features in-place modification* of point, line, and polygon features

Siehe auch

Array of translated features, *Offset lines*, *Rotate*, *Swap X and Y coordinates*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Offset distance (x-axis)	`DELTA_X`	[numeric: double] Default: 0.0	Displacement to apply on the X axis
Offset distance (y-axis)	`DELTA_Y`	[numeric: double] Default: 0.0	Displacement to apply on the Y axis
Offset distance (z-axis)	`DELTA_Z`	[numeric: double] Default: 0.0	Displacement to apply on the Z axis
Offset distance (m values)	`DELTA_M`	[numeric: double] Default: 0.0	Displacement to apply on the M axis

Fortsetzung auf der nächsten Seite

Tab. 24.754 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Translated	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Translated	`OUTPUT`	[wie Eingabe]	Output vector layer

Pythoncode

Algorithm ID: `native:translategeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Variable width buffer (by M value)

Creates variable width buffers along lines, using the M value of the line geometries as the diameter of the buffer at each vertex.

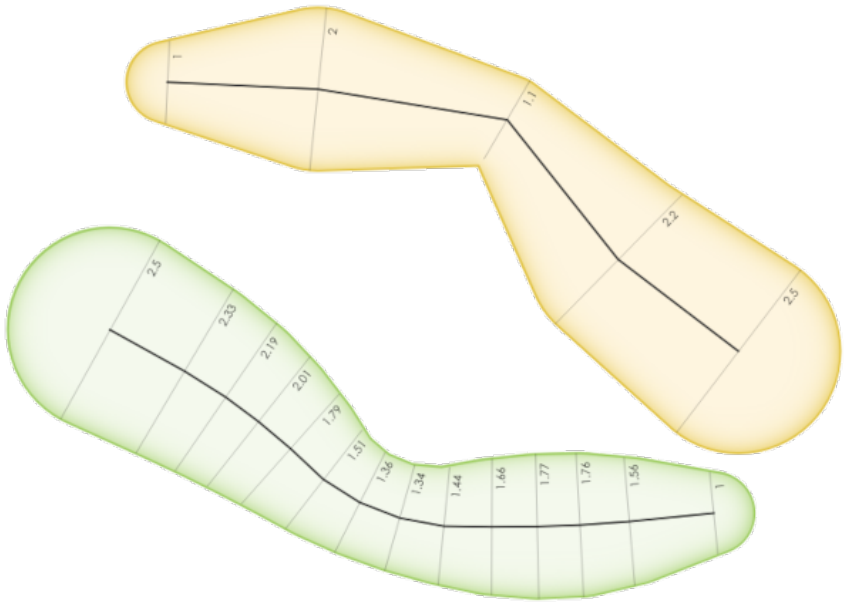


Abb. 24.147: Variable buffer example

Siehe auch

Tapered buffers, *Puffer*, *M-Wert einstellen*, *Variable distance buffer*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line vector layer
Segmente	`SEGMENTS`	[numeric: integer] Default: 16	Number of the buffer segments per quarter circle. It can be a unique value (same value for all the features), or it can be taken from features data (the value can depend on feature attributes).
Gepuffert	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output (buffer) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Gepuffert	`OUTPUT`	[Vektor: Polygon]	Variable buffer polygon layer

Pythoncode

Algorithm ID: `native:bufferbym`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Voronoi polygons

Takes a point layer and generates a polygon layer containing the Voronoi polygons (known also as Thiessen polygons) corresponding to those input points.

Any location within a Voronoi polygon is closer to the associated point than to any other point.

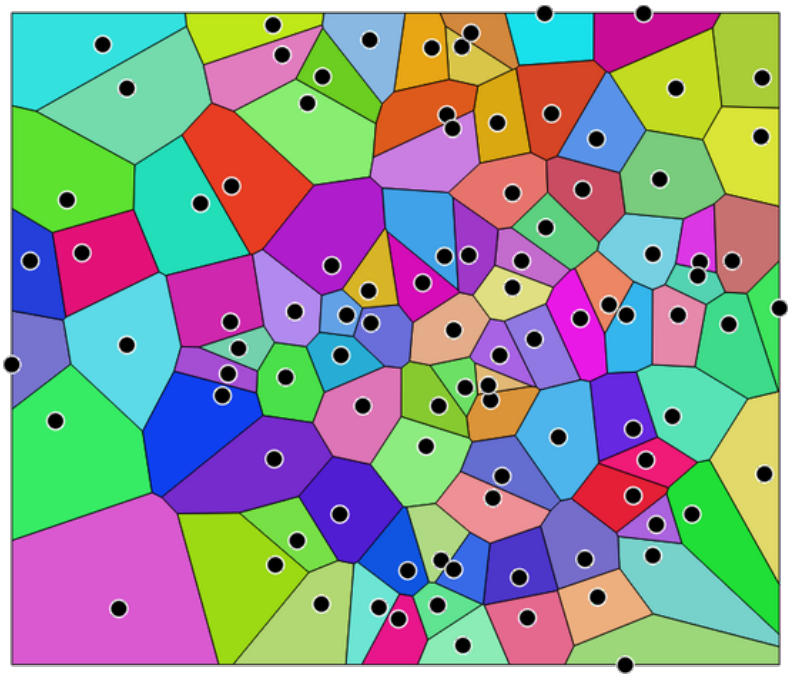


Abb. 24.148: Voronoi polygons

Default menu: *Vector ► Geometry Tools*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Buffer region (% of extent)	`BUFFER`	[numerisch: Double] Default: 0.0	The extent of the output layer will be this much bigger than the extent of the input layer
Tolerance Optional	`TOLERANCE`	[numerisch: Double] Default: 0.0	Specifies an optional snapping tolerance which can be used to improve the robustness of the voronoi.
Copy attributes from input features	`COPY_ATTRIBUTE`	[Boolean] Standard: True	Specifies whether fields storing involved point features ID should be added to the output. If False, an `id` field is created to identify the polygons.
Voronoi (Thiessen) Polygone	`OUTPUT`	[Vektor: Polygon] Standard: `[Create temporary layer]`	Specify the output layer (with the Voronoi polygons). *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Voronoi (Thiessen) Polygone	`OUTPUT`	[Vektor: Polygon]	Voronoi polygons of the input point vector layer

Pythoncode

Algorithm ID: `native:voronoipolygons`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.27 Vector overlay

Zuschneiden

Clips a vector layer using the features of an additional polygon layer.

Only the parts of the features in the input layer that fall within the polygons of the overlay layer will be added to the resulting layer.

Warnung

Nur Geometrieänderung

Dieser Vorgang verändert nur die Geometrie der Features. Die Attributwerte der Features werden **nicht verändert**, obwohl Eigenschaften wie Fläche oder Länge der Features durch den Overlay-Vorgang verändert werden. Wenn solche Eigenschaften als Attribute gespeichert sind, müssen diese Attribute manuell aktualisiert werden.

This algorithm uses spatial indexes on the providers, prepared geometries and apply a clipping operation if the geometry isn't wholly contained by the mask geometry.

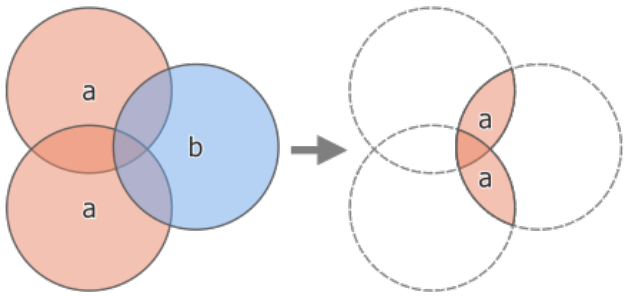


Abb. 24.149: Clipping operation between a two-feature input layer ‚a' and a single feature overlay layer ‚b' (left) - resulting in a new layer with the modified ‚a' features (right)

Allows *features in-place modification* of point, line, and polygon features

Default menu: *Vector* ► *Geoprocessing Tools*

Siehe auch

Intersection, *Difference*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Layer containing the features to be clipped
Overlay layer	`OVERLAY`	[Vektor: Polygon]	Layer containing the clipping features

Fortsetzung auf der nächsten Seite

Tab. 24.760 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Zugeschnitten	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the layer to contain the features from the input layer that are inside the overlay (clipping) layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Zugeschnitten	`OUTPUT`	[wie Eingabe]	Layer containing features from the input layer split by the overlay layer.

Python-Code

Algorithm ID: `native:clip`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Difference

Extracts features from the input layer that don't fall within the boundaries of the overlay layer.

Input layer features that partially overlap the overlay layer feature(s) are split along the boundary of those feature(s) and only the portions outside the overlay layer features are retained.

Warnung

Nur Geometrieänderung

Dieser Vorgang verändert nur die Geometrie der Features. Die Attributwerte der Features werden **nicht verändert**, obwohl Eigenschaften wie Fläche oder Länge der Features durch den Overlay-Vorgang verändert werden. Wenn solche Eigenschaften als Attribute gespeichert sind, müssen diese Attribute manuell aktualisiert werden.

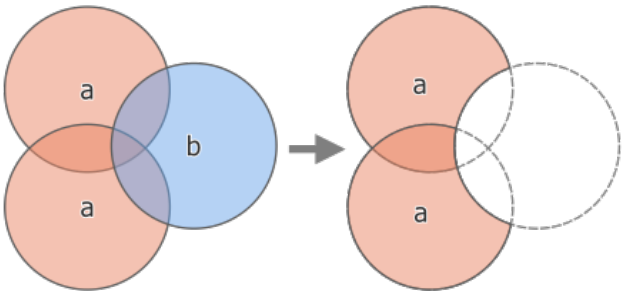


Abb. 24.150: Difference operation between a two-features input layer ‚a' and a single feature overlay layer ‚b' (left) - resulting in a new layer with the modified ‚a' features (right)

Allows *features in-place modification* of point, line, and polygon features

Default menu: *Vector* ► *Geoprocessing Tools*

Siehe auch

Difference (multiple), *Symmetrical difference*, *Zuschneiden*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Layer to extract (parts of) features from.
Overlay layer	`OVERLAY`	[Vektor: Geometrie]	Layer containing the geometries that will be subtracted from the input layer geometries. It is expected to have at least as many dimensions (point: 0D, line: 1D, polygon: 2D, volume: 3D) as the input layer geometries.
Difference	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the layer to contain the (parts of) features from the input layer that are not inside the overlay layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Grid size Optional	`GRID_SIZE`	[numerisch: Double] Standard: Nicht gesetzt	If provided, the input geometries are snapped to a grid of the given size, and the result vertices are computed on that same grid.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Difference	`OUTPUT`	[wie Eingabe]	Layer containing (parts of) features from the input layer not overlapping the overlay layer.

Python-Code

Algorithm ID: `native:difference`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Difference (multiple)

Extracts features from the input layer that fall completely outside or only partially overlap the features from any of the overlay layer(s).

For each overlay layer the difference is calculated between the result of all previous difference operations and this overlay layer. Input layer features that partially overlap feature(s) in the overlay layer are split along those features' boundary and only the portions outside the overlay layer features are retained.

Warnung

Nur Geometrieänderung

Dieser Vorgang verändert nur die Geometrie der Features. Die Attributwerte der Features werden **nicht verändert**, obwohl Eigenschaften wie Fläche oder Länge der Features durch den Overlay-Vorgang verändert werden. Wenn solche Eigenschaften als Attribute gespeichert sind, müssen diese Attribute manuell aktualisiert werden.

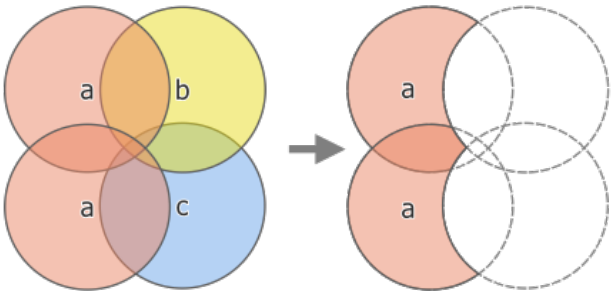


Abb. 24.151: Difference operation between a two-feature input layer ‚a' and single feature overlay layers ‚b' and ‚c' (left) - resulting in a new layer with the modified ‚a' features (right)

Siehe auch

Difference, *Symmetrical difference*, *Zuschneiden*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Layer to extract (parts of) features from.
Overlay layers	`OVERLAYS`	[vector: geometry] [list]	List of layers containing the geometries that will be subtracted from the input layer geometries. They are expected to have at least as many dimensions (point: 0D, line: 1D, polygon: 2D, volume: 3D) as the input layer geometries.
Difference	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the layer to contain the (parts of) features from the input layer that do not overlap features of the overlay layers. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Difference	`OUTPUT`	[wie Eingabe]	Layer containing (parts of) features from the input layer not overlapping features from the overlay layers.

Python-Code

Algorithm ID: `native:multidifference`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract/clip by extent

Creates a new vector layer that only contains features which fall within a specified extent. Any features which intersect the extent will be included by default.

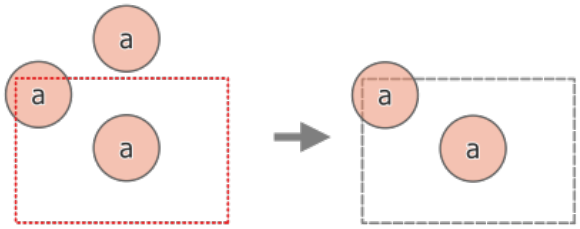


Abb. 24.152: Extract operation between a three-feature input layer ‚a' and a dashed extent (left) - resulting features with dashed extent for reference (right)

Optionally, feature geometries can also be clipped to the extent.

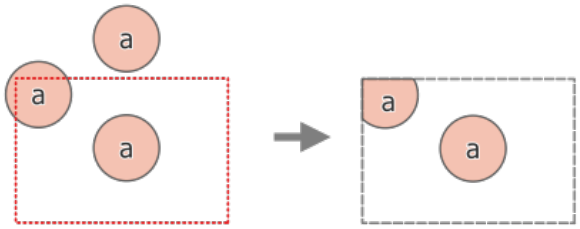


Abb. 24.153: Extract operation between a three-feature input layer ‚a' and a dashed extent (left) - resulting features with dashed extent for reference (right)

Siehe auch

Zuschneiden

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Layer to extract (parts of) features from.
Extent (xmin, xmax, ymin, ymax)	`EXTENT`	[Ausdehnung]	Extent for clipping. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Aktuelles Kartenansichtsausmaß verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Clip features to extent	`CLIP`	[Boolean] Standard: falsch	If checked, the geometries will be clipped to the extent chosen instead of taking the whole geometry as output. Moreover, output geometries will be automatically converted to multi geometries to ensure uniform output types.
Extracted	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the layer to contain the features from the input layer that are inside the clip extent. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extracted	`OUTPUT`	[wie Eingabe]	Layer containing the clipped features.

Python-Code

Algorithm ID: `native:extractbyextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Intersection

Extracts the portions of features from the input layer that overlap features in the overlay layer.

Features in the intersection layer are assigned the attributes of the overlapping features from both the input and overlay layers.

Warnung

Nur Geometrieänderung

Dieser Vorgang verändert nur die Geometrie der Features. Die Attributwerte der Features werden **nicht verändert**, obwohl Eigenschaften wie Fläche oder Länge der Features durch den Overlay-Vorgang verändert werden. Wenn solche Eigenschaften als Attribute gespeichert sind, müssen diese Attribute manuell aktualisiert werden.

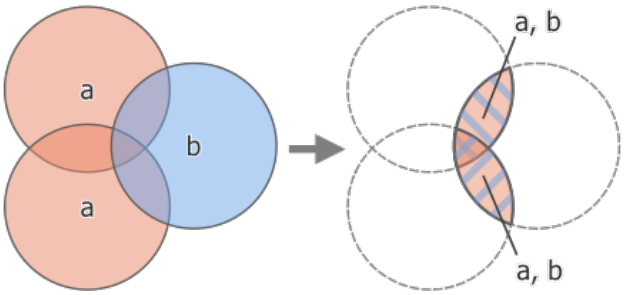


Abb. 24.154: Intersection operation between a two-feature input layer ‚a' and a single feature overlay layer ‚b' (left) - overlapping areas become a new two-feature layer with both layers' attributes (right)

Default menu: *Vector ► Geoprocessing Tools*

Warnung

This algorithm drops existing primary keys or FID values and regenerates them in output layers.

Siehe auch

Intersection (multiple), *Zuschneiden*, *Difference*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Layer to extract (parts of) features from.
Overlay layer	`OVERLAY`	[Vektor: Geometrie]	Layer containing the features to check for overlap. Its features' geometry is expected to have at least as many dimensions (point: 0D, line: 1D, polygon: 2D, volume: 3D) as the input layer's.
Input fields to keep (leave empty to keep all fields) Optional	`INPUT_FIELDS`	[tablefield: any] [list] Standard: Nicht gesetzt	Field(s) of the input layer to keep in the output. If no fields are chosen all fields are taken.
Overlay fields to keep (leave empty to keep all fields) Optional	`OVERLAY_FIELDS`	[tablefield: any] [list] Standard: Nicht gesetzt	Field(s) of the overlay layer to keep in the output. If no fields are chosen all fields are taken. Duplicate field names will be appended a count suffix to avoid collision.
Intersection	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the layer to contain (the parts of) the features from the input layer that overlap one or more features from the overlay layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Overlay fields prefix Optional	`OVERLAY_FIELDS_`	[Text]	Add a prefix to identify fields of the overlay layer. Duplicate field names will be appended a count suffix to avoid collision.
Grid size Optional	`GRID_SIZE`	[numerisch: Double] Standard: Nicht gesetzt	If provided, the input geometries are snapped to a grid of the given size, and the result vertices are computed on that same grid.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Intersection	`OUTPUT`	[wie Eingabe]	Layer containing (parts of) features from the input layer that overlap the overlay layer.

Python-Code

Algorithm ID: `native:intersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Intersection (multiple)

Extracts the overlapping portions of features in the input and all overlay layers.

Features in the output layer are assigned the attributes of the overlapping features from both the input and overlay layers.

Warnung

Nur Geometrieänderung

Dieser Vorgang verändert nur die Geometrie der Features. Die Attributwerte der Features werden **nicht verändert**, obwohl Eigenschaften wie Fläche oder Länge der Features durch den Overlay-Vorgang verändert werden. Wenn solche Eigenschaften als Attribute gespeichert sind, müssen diese Attribute manuell aktualisiert werden.

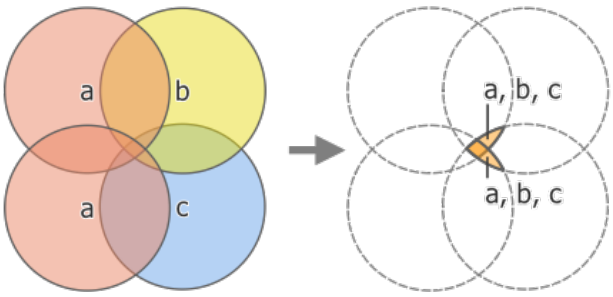


Abb. 24.155: Intersection operation between a two-feature input layer ‚a' and single feature overlay layers ‚b' and ‚c' (left) - overlapping areas become a new two-feature layer with all layers' attributes (right)

Warnung

This algorithm drops existing primary keys or FID values and regenerate them in output layers.

Siehe auch

Intersection, *Zuschneiden*, *Difference*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Layer to extract (parts of) features from.
Overlay layers	`OVERLAYS`	[vector: geometry] [list]	Layers containing the features to check for overlap. The features' geometry is expected to have at least as many dimensions (point: 0D, line: 1D, polygon: 2D, volume: 3D) as the input layer's.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Overlay fields prefix Optional	`OVERLAY_FIELDS_`	[Text]	Add a prefix to identify fields of the overlay layers. Duplicate field names will be appended a count suffix to avoid collision.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Intersection	`OUTPUT`	[wie Eingabe]	Layer containing (parts of) features from the input layer that overlap all the overlay layers.

Python-Code

Algorithm ID: `native:multiintersection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Line intersections

Creates point features where the lines from the two layers intersect.

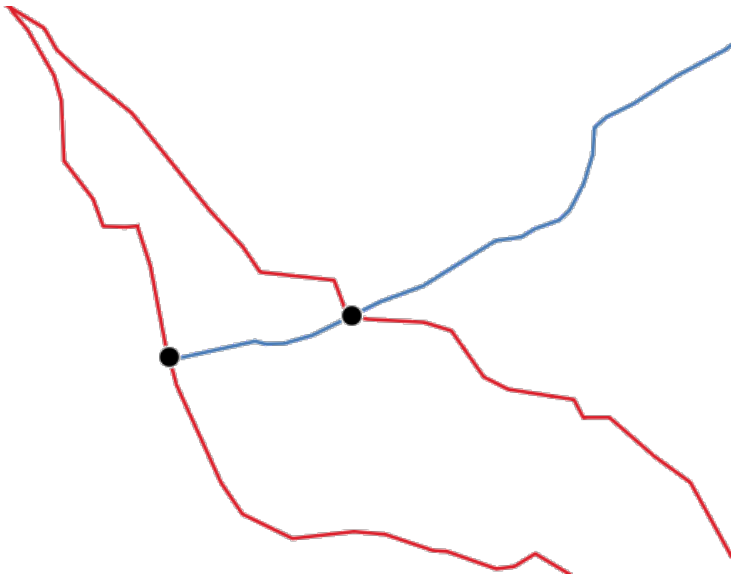


Abb. 24.156: Points of intersection

Default menu: *Vector* ► *Analysis Tools*

> **Warnung**
>
> This algorithm drops existing primary keys or FID values and regenerate them in output layers.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	Input line layer.
Intersect layer	`INTERSECT`	[vector: line]	Layer to use to find line intersections.
Input fields to keep (leave empty to keep all fields) Optional	`INPUT_FIELDS`	[tablefield: any] [list] Standard: Nicht gesetzt	Field(s) of the input layer to keep in the output. If no fields are chosen all fields are taken.
Intersect fields to keep (leave empty to keep all fields) Optional	`INTERSECT_FIEL`	[tablefield: any] [list] Standard: Nicht gesetzt	Field(s) of the intersect layer to keep in the output. If no fields are chosen all fields are taken. Duplicate field names will be appended a count suffix to avoid collision.

Fortsetzung auf der nächsten Seite

Tab. 24.775 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Intersection	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the layer to contain the intersection points of the lines from the input and overlay layers. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Intersect fields prefix Optional	`INTERSECT_FIEL`	[Text]	Add a prefix to identify fields of the intersect layer.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Intersections	`OUTPUT`	[Vektor: Punkt]	Point vector layer of the lines intersections, with both layers' attributes.

Python-Code

Algorithm ID: `native:lineintersections`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Split with lines

Splits the lines or polygons in one layer using the lines or polygon rings in another layer to define the breaking points. Intersection between geometries in both layers are considered as split points.

Output will contain multi geometries for split features.

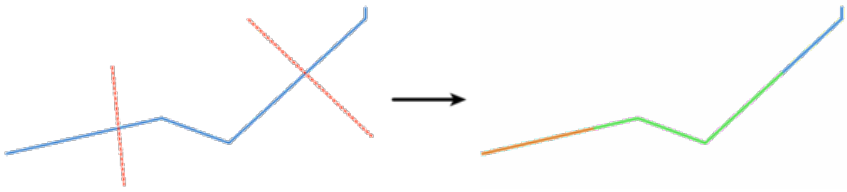


Abb. 24.157: Split lines

Allows *features in-place modification* of line and polygon features

Warnung

This algorithm drops existing primary keys or FID values and regenerate them in output layers.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line, polygon]	Layer containing the lines or polygons to split.
Split layer	`LINES`	[vector: line, polygon]	Layer whose lines or rings are used to define the breaking points.
Split	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the layer to contain the splitted (in case they are intersected by a line in the split layer) line/polygon features from the input layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Split	`OUTPUT`	[wie Eingabe]	Output vector layer with split lines or polygons from input layer.

Python-Code

Algorithm ID: `native:splitwithlines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Symmetrical difference

Creates a layer containing features from both the input and overlay layers but with the overlapping areas between the two layers removed.

The attribute table of the symmetrical difference layer contains attributes and fields from both the input and overlay layers.

Warnung

Nur Geometrieänderung

Dieser Vorgang verändert nur die Geometrie der Features. Die Attributwerte der Features werden **nicht verändert**, obwohl Eigenschaften wie Fläche oder Länge der Features durch den Overlay-Vorgang verändert werden. Wenn solche Eigenschaften als Attribute gespeichert sind, müssen diese Attribute manuell aktualisiert werden.

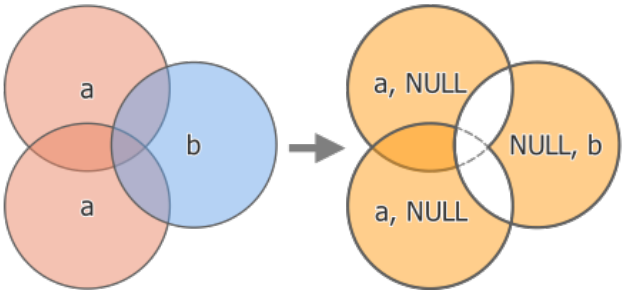


Abb. 24.158: Symmetrical difference operation between a two-features input layer ‚a' and a single feature overlay layer ‚b' (left) - resulting three-feature layer with both layers' attributes (right)

Default menu: *Vector* ► *Geoprocessing Tools*

Warnung

This algorithm drops existing primary keys or FID values and regenerate them in output layers.

Siehe auch

Difference, *Zuschneiden*, *Intersection*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	First layer to extract (parts of) features from.
Overlay layer	`OVERLAY`	[Vektor: Geometrie]	Second layer to extract (parts of) features from. Ideally the geometry type should be the same as input layer.
Symmetrical difference	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the layer to contain (the parts of) the features from the input and overlay layers that do not overlap features from the other layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Overlay fields prefix Optional	`OVERLAY_FIELDS_`	[Text]	Add a prefix to identify fields of the overlay layer. Duplicate field names will be appended a count suffix to avoid collision.
Grid size Optional	`GRID_SIZE`	[numerisch: Double] Standard: Nicht gesetzt	If provided, the input geometries are snapped to a grid of the given size, and the result vertices are computed on that same grid.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Symmetrical difference	`OUTPUT`	[wie Eingabe]	Layer containing (parts of) features from each layer not overlapping the other layer, with both layers‘ attributes.

Python-Code

Algorithm ID: `native:symmetricaldifference`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Union

Checks overlaps between features within the input layer and creates separate features for overlapping and non-overlapping parts. The area of overlap will create as many identical overlapping features as there are features that participate in that overlap.

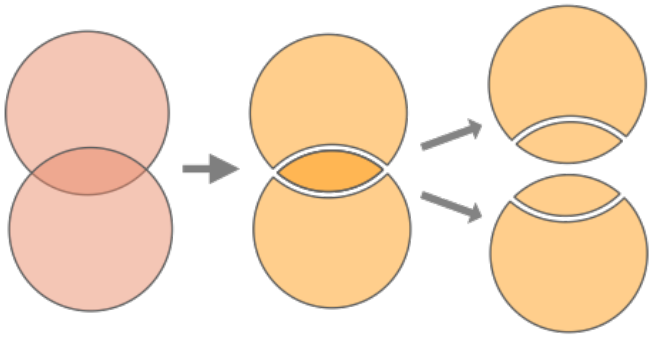


Abb. 24.159: Union operation with a single input layer with two overlapping features (left) - resulting in four features (middle) - features moved for clarity (right)

An overlay layer can also be used, in which case features from each layer are split at their overlap with features from the other one, creating a layer containing all the portions from both input and overlay layers. Features on the same layer will not split each other. The attribute table of the union layer is filled with attribute values from the respective original layer for non-overlapping features, and attribute values from both layers for overlapping features.

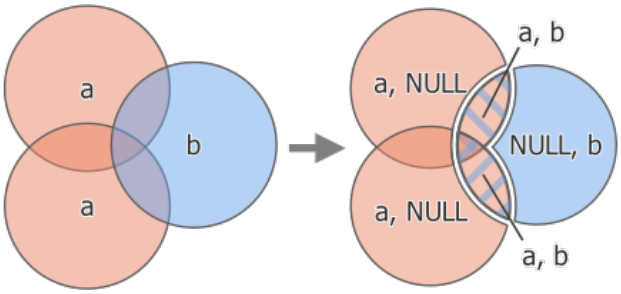


Abb. 24.160: Union operation between a two-feature input layer ‚a' and a single feature overlay layer ‚b' (left) - resulting five-feature layer with attributes from both layers (right)

Bemerkung

With an overlay layer, features on the same layer will not split each other. If you want to split overlaps on the same layer as well as other layers, first run the algorithm with multiple layers then run the algorithm again with

only the previous output.

Default menu: *Vector ► Geoprocessing Tools*

Warnung

This algorithm drops existing primary keys or FID values and regenerate them in output layers.

Siehe auch

Union (multiple), *Zuschneiden*, *Difference*, *Intersection*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Input vector layer to split at any intersections.
Overlay layer Optional	`OVERLAY`	[Vektor: Geometrie]	Layer that will be combined to the first one. Ideally the geometry type should be the same as input layer.
Union	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the layer to contain the (split and duplicated) features from the input layer and the overlay layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Overlay fields prefix Optional	`OVERLAY_FIELDS_`	[Text]	Add a prefix to identify fields of the overlay layer. Duplicate field names will be appended a count suffix to avoid collision.
Grid size Optional	`GRID_SIZE`	[numerisch: Double] Standard: Nicht gesetzt	If provided, the input geometries are snapped to a grid of the given size, and the result vertices are computed on that same grid.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Union	`OUTPUT`	[wie Eingabe]	Layer containing all the overlapping and non-overlapping parts from the processed layer(s).

Python-Code

Algorithm ID: `native:union`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Union (multiple)

Checks overlaps between features within the input layer and creates separate features for overlapping and non-overlapping parts. The area of overlap will create as many identical overlapping features as there are features that participate in that overlap.

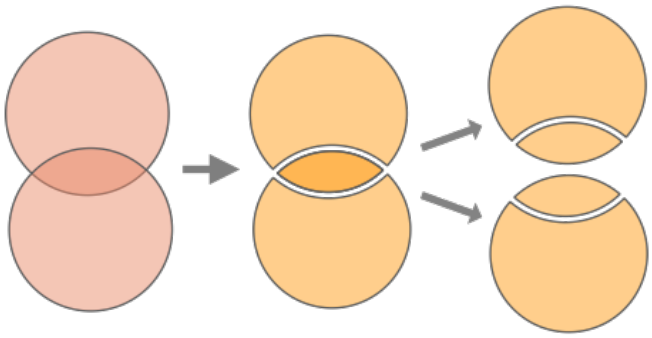


Abb. 24.161: Union operation with a single input layer with two overlapping features (left) - resulting in four features (middle) - features moved for clarity (right)

Multiple overlay layers can also be used, in which case features from each layer are split at their overlap with features from all other layers, creating a layer containing all the portions from both input and overlay layers. Features on the same layer will not split each other. The attribute table of the Union layer is filled with attribute values from the respective original layer for non-overlapping features, and attribute values from overlay layers for overlapping features.

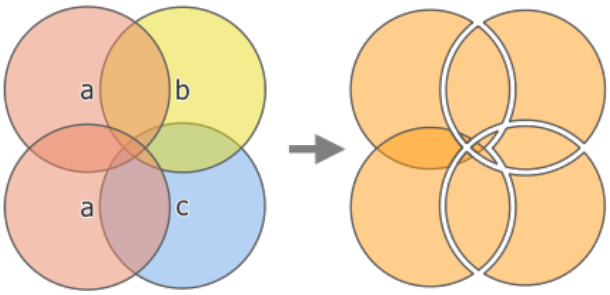


Abb. 24.162: Union operation between a two-feature input layer ‚a' and single feature overlay layers ‚b' and ‚c' (left) - resulting eleven-feature layer with attributes from all layers (right)

Bemerkung

With an overlay layer, features on the same layer will not split each other. If you want to split overlaps on the same layer as well as other layers, first run the algorithm with multiple layers then run the algorithm again with only the previous output.

Warnung

This algorithm drops existing primary keys or FID values and regenerate them in output layers.

Siehe auch

Union, *Zuschneiden*, *Difference*, *Intersection*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Input vector layer to split at any intersections.
Overlay layers Optional	`OVERLAYS`	[vector: geometry] [list]	Layers that will be combined to the first one. Ideally the geometry type should be the same as input layer.
Union	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the layer to contain the (split and duplicated) features from the input layer and the overlay layers. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Overlay fields prefix Optional	`OVERLAY_FIELDS_`	[Text]	Add a prefix to identify fields of the overlay layers. Duplicate field names will be appended a count suffix to avoid collision.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Union	`OUTPUT`	[wie Eingabe]	Layer containing all the overlapping and non-overlapping parts from the processed layer(s), with all layers' attributes.

Python-Code

Algorithm ID: `native:multiunion`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.28 Vector selection

Extract by attribute

Creates two vector layers from an input layer: one will contain only matching features while the second will contain all the non-matching features.

The criteria for adding features to the resulting layer is based on the values of an attribute from the input layer.

Siehe auch

Select by attribute

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Layer to extract features from.
Selection attribute	`FIELD`	[tablefield: any]	Filtering field of the layer
Operator	`OPERATOR`	[Aufzählung] Standard: 0	Many different operators are available: • 0 — = • 1 — ≠ • 2 — >> • 3 — >= • 4 — < • 5 — <= • 6 — begins with • 7 — contains • 8 — is null • 9 — is not null • 10 — does not contain
Value Optional	`VALUE`	[Text]	Value to be evaluated
Extracted (attribute)	`OUTPUT`	[wie Eingabe] Default: `[Create Temporary Layer]`	Specify the output vector layer for matching features. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Extracted (non-matching)	`FAIL_OUTPUT`	[wie Eingabe] Standard: `[Ausgabe überspringen]`	Specify the output vector layer for non-matching features. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extracted (attribute)	`OUTPUT`	[wie Eingabe]	Vector layer with matching features from the input layer
Extracted (non-matching)	`FAIL_OUTPUT`	[wie Eingabe]	Vector layer with non-matching features from the input layer

Pythoncode

Algorithm ID: `native:extractbyattribute`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract by expression

Creates two vector layers from an input layer: one will contain only matching features while the second will contain all the non-matching features.

The criteria for adding features to the resulting layer is based on a QGIS expression. For more information about expressions see the *Ausdrücke*.

Siehe auch

Select by expression

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer
Expression	`EXPRESSION`	[Ausdruck]	Expression to filter the vector layer
Matching features	`OUTPUT`	[wie Eingabe] Default: `[Create Temporary Layer]`	Specify the output vector layer for matching features. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.
Non-matching	`FAIL_OUTPUT`	[wie Eingabe] Standard: `[Ausgabe überspringen]`	Specify the output vector layer for non-matching features. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Matching features	`OUTPUT`	[wie Eingabe]	Vector layer with matching features from the input layer
Non-matching	`FAIL_OUTPUT`	[wie Eingabe]	Vector layer with non-matching features from the input layer

Pythoncode

Algorithm ID: `native:extractbyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract by location

Creates a new vector layer that only contains matching features from an input layer.

The criteria for adding features to the resulting layer is based on the spatial relationship between each feature and the features in an additional layer.

Siehe auch

Auswahl nach der Lage, *Extract within distance*

Exploring spatial relations

Geometrische Prädikate sind boolesche Funktionen, die verwendet werden, um die räumliche Beziehung eines Features zu einem anderen zu bestimmen, indem verglichen wird, ob und wie ihre Geometrien einen Teil des Raums teilen.

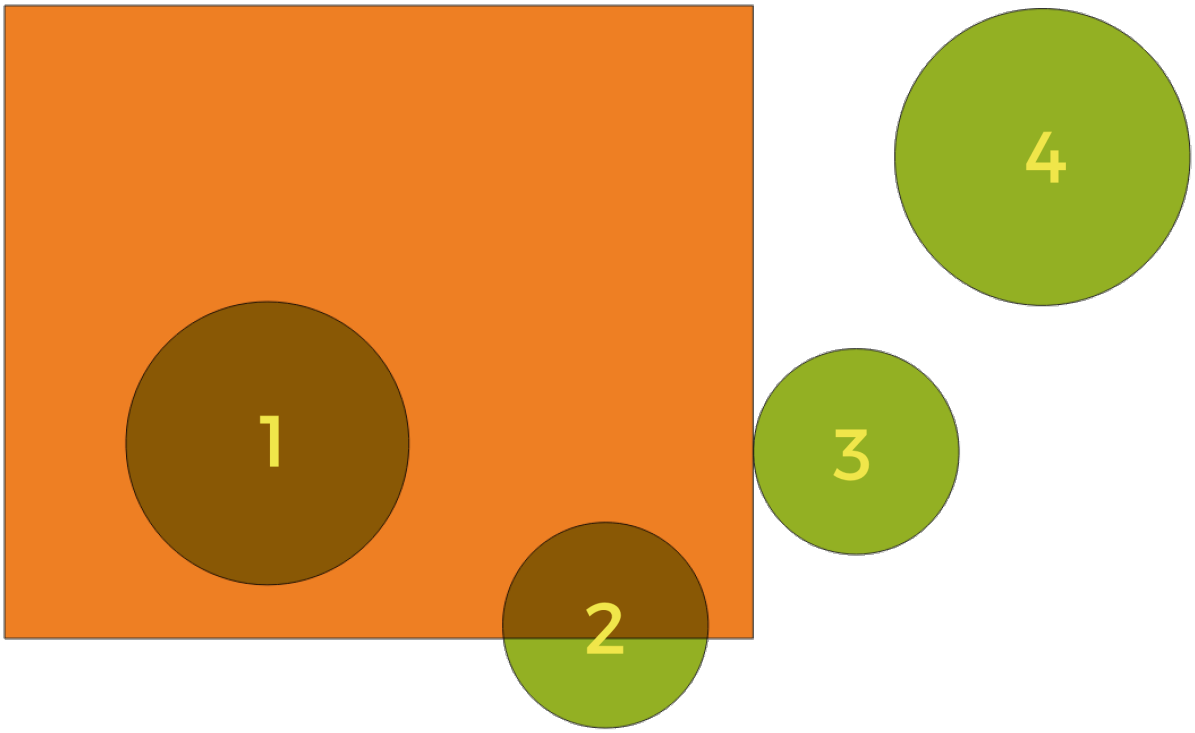


Abb. 24.163: Suche nach räumlichen Beziehungen zwischen Layern

Unter Verwendung der obigen Abbildung suchen wir nach den grünen Kreisen, indem wir sie räumlich mit dem orangefarbenen Rechteck-Feature vergleichen. Verfügbare geometrische Prädikate sind:

Schneidet
Testet, ob eine Geometrie eine andere schneidet. Gibt 1 (wahr) zurück, wenn sich die Geometrien räumlich überschneiden (einen Teil des Raums teilen – überlappen oder berühren) und 0, wenn dies nicht der Fall ist. Im obigen Bild werden die Kreise 1, 2 und 3 zurückgegeben.

Enthält
Gibt 1 (wahr) zurück, wenn und nur wenn keine Punkte von b außerhalb von a liegen und mindestens ein Punkt des Inneren von b innerhalb von a liegt. Im Bild wird kein Kreis zurückgegeben, sondern das Rechteck wäre es, wenn man es umgekehrt suchen würde, da es den Kreis 1 komplett enthält. Das ist das Gegenteil von *sind innerhalb*.

Getrennt
Gibt 1 (wahr) zurück, wenn die Geometrien keinen Teil des Raums teilen (keine Überlappung, keine Berührung). Nur Kreis 4 wird zurückgegeben.

Gleich
Gibt 1 (wahr) zurück, wenn und nur wenn die Geometrien genau gleich sind. Es werden keine Kreise zurückgegeben.

Berührt
Testet, ob eine Geometrie eine andere berührt. Gibt 1 (wahr) zurück, wenn die Geometrien mindestens einen Punkt gemeinsam haben, aber ihre Innenräume sich nicht schneiden. Nur Kreis 3 wird zurückgegeben.

Überlappt
Testet, ob eine Geometrie eine andere überlappt. Gibt 1 (wahr) zurück, wenn die Geometrien sich den Raum teilen, dieselbe Dimension haben, aber nicht vollständig ineinander liegen. Nur Kreis 2 wird zurückgegeben.

Sind innerhalb
Testet, ob sich eine Geometrie in einer anderen befindet. Gibt 1 (wahr) zurück, wenn sich Geometrie a vollständig innerhalb von Geometrie b befindet. Nur Kreis 1 wird zurückgegeben.

Kreuzen

Gibt 1 (wahr) zurück, wenn die bereitgestellten Geometrien einige, aber nicht alle inneren Punkte gemeinsam haben und die tatsächliche Kreuzung eine geringere Dimension als die höchste bereitgestellte Geometrie hat. Beispielsweise wird eine Linie, die ein Polygon kreuzt, als Linie gekreuzt (wahr). Zwei Linien, die sich kreuzen, kreuzen sich als Punkt (wahr). Zwei Polygone kreuzen sich als Polygon (falsch). Im Bild werden keine Kreise zurückgegeben.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Extract features from	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Where the features (geometric predicate)	`PREDICATE`	[Aufzählung] [Liste] Default: [0]	Type of spatial relation the input feature should have with an intersect feature so that it could be selected. One or more of: • 0 — intersect • 1 — contain • 2 — disjoint • 3 — equal • 4 — touch • 5 — overlap • 6 — are within • 7 — cross If more than one condition is chosen, at least one of them (OR operation) has to be met for a feature to be extracted.
By comparing to the features from	`INTERSECT`	[Vektor: Geometrie]	Intersection vector layer
Extracted (location)	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer for the features that have the chosen spatial relationship(s) with one or more features in the comparison layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extracted (location)	`OUTPUT`	[wie Eingabe]	Vector layer with features from the input layer that have the chosen spatial relationship(s) with one or more features in the comparison layer.

Pythoncode

Algorithm ID: `native:extractbylocation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract within distance

Creates a new vector layer that only contains matching features from an input layer. Features are copied wherever they are within the specified maximum distance from the features in an additional reference layer.

Siehe auch

Select within distance, *Extract by location*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Extract features from	`INPUT`	[Vektor: Geometrie]	Input vector layer to copy features from
By comparing to the features from	`REFERENCE`	[Vektor: Geometrie]	Vector layer whose features closeness is used
Where the features are within	`DISTANCE`	[numeric: double] Default: 100.0	The maximum distance around reference features to select input features within
Modify current selection by	`METHOD`	[Aufzählung] Standard: 0	How the selection of the algorithm should be managed. One of: • 0 — creating new selection • 1 — adding to current selection • 2 — selecting within current selection • 3 — removing from current selection
Extracted (location)	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer for the features that are within the set distance from reference features. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extracted (location)	`OUTPUT`	[wie Eingabe]	Vector layer with features from the input layer matching the condition of distance from reference features

Pythoncode

Algorithm ID: `native:extractwithindistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Filter by geometry type

Filters features by their geometry type. Incoming features will be directed to different outputs based on whether they have a point, line or polygon geometry.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Layer to evaluate

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Point features Optional	`POINTS`	[Vektor: Punkt]	Layer with points
Line features Optional	`LINES`	[vector: line]	Layer with lines
Polygon features Optional	`POLYGONS`	[Vektor: Polygon]	Layer with polygons
Features with no geometry Optional	`NO_GEOMETRY`	[vector: table]	Geometry-less vector layer

Pythoncode

Algorithm ID: `native:filterbygeometry`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Random extract

Takes a vector layer and generates a new one that contains only a subset of the features in the input layer.

The subset is defined randomly, based on feature IDs, using a percentage or count value to define the total number of features in the subset.

Warnung

This algorithm drops existing primary keys or FID values and regenerate them in output layers.

Siehe auch

Random selection

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Source vector layer to select the features from
Methode	`METHOD`	[Aufzählung] Standard: 0	Random selection methods. One of: • 0 — Number of selected features • 1 — Percentage of selected features
Number/percentage of selected features	`NUMBER`	[numerisch: Ganzzahl] Default: 10	Number or percentage of features to select
Extracted (random)	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer for the randomly selected features. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extracted (random)	`OUTPUT`	[wie Eingabe]	Vector layer containing randomly selected features from the input layer

Pythoncode

Algorithm ID: `native:randomextract`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Random extract within subsets

Takes a vector layer and generates a new one that contains only a subset of the features in the input layer.

The subset is defined randomly, based on feature IDs, using a percentage or count value to define the total number of features in the subset. The percentage/count value is not applied to the whole layer, but instead to each category. Categories are defined according to a given attribute.

Siehe auch

Random selection within subsets

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Vector layer to select the features from
ID field	`FIELD`	[tablefield: any]	Category of the source vector layer to select the features from
Methode	`METHOD`	[Aufzählung] Standard: 0	Random selection method. One of: • 0 — Number of selected features • 1 — Percentage of selected features
Number/percentage of selected features	`NUMBER`	[numerisch: Ganzzahl] Default: 10	Number or percentage of features to select

Fortsetzung auf der nächsten Seite

Tab. 24.801 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Extracted (random stratified)	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer for the randomly selected features. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Extracted (random stratified)	`OUTPUT`	[wie Eingabe]	Vector layer containing randomly selected features from the input layer

Pythoncode

Algorithm ID: `qgis:randomextractwithinsubsets`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Random selection

Takes a vector layer and selects a subset of its features. No new layer is generated by this algorithm.

The subset is defined randomly, based on feature IDs, using a percentage or count value to define the total number of features in the subset.

Default menu: *Vector ► Research Tools*

Siehe auch

Random extract

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Vector layer for the selection
Methode	`METHOD`	[Aufzählung] Standard: 0	Random selection method. One of: • 0 — Number of selected features • 1 — Percentage of selected features
Number/percentage of selected features	`NUMBER`	[numerisch: Ganzzahl] Default: 10	Number or percentage of features to select

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[wie Eingabe]	The input layer with features selected

Pythoncode

Algorithm ID: `qgis:randomselection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Random selection within subsets

Takes a vector layer and selects a subset of its features. No new layer is generated by this algorithm.

The subset is defined randomly, based on feature IDs, using a percentage or count value to define the total number of features in the subset.

The percentage/count value is not applied to the whole layer, but instead to each category.

Categories are defined according to a given attribute, which is also specified as an input parameter for the algorithm.

No new outputs are created.

Default menu: *Vector ► Research Tools*

Siehe auch

Random extract within subsets

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Vector layer to select features in
ID field	`FIELD`	[tablefield: any]	Category of the input layer to select the features from
Methode	`METHOD`	[Aufzählung] Standard: 0	Random selection method. One of: • 0 — Number of selected features • 1 — Percentage of selected features
Number/percentage of selected features	`NUMBER`	[numerisch: Ganzzahl] Default: 10	Number or percentage of features to select

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[wie Eingabe]	The input layer with features selected

Pythoncode

Algorithm ID: `qgis:randomselectionwithinsubsets`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Select by attribute

Erstellt eine Auswahl in einem Vektorlayer.

The criteria for selecting features is based on the values of an attribute from the input layer.

Siehe auch

Extract by attribute

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Vector layer to select features in
Selection attribute	`FIELD`	[tablefield: any]	Filtering field of the layer
Operator	`OPERATOR`	[Aufzählung] Standard: 0	Many different operators are available: • 0 — = • 1 — ≠ • 2 — >> • 3 — >= • 4 — < • 5 — <= • 6 — begins with • 7 — contains • 8 — is null • 9 — is not null • 10 — does not contain
Value Optional	`VALUE`	[Text]	Value to be evaluated
Modify current selection by	`METHOD`	[Aufzählung] Standard: 0	How the selection of the algorithm should be managed. One of: • 0 — creating new selection • 1 — adding to current selection • 2 — removing from current selection • 3 — selecting within current selection

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[wie Eingabe]	The input layer with features selected

Pythoncode

Algorithm ID: `qgis:selectbyattribute`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Select by expression

Erstellt eine Auswahl in einem Vektorlayer.

The criteria for selecting features is based on a QGIS expression. For more information about expressions see the *Ausdrücke*.

Siehe auch

Extract by expression

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer
Expression	`EXPRESSION`	[Ausdruck]	Expression to filter the input layer
Modify current selection by	`METHOD`	[Aufzählung] Standard: 0	How the selection of the algorithm should be managed. One of: • 0 — creating new selection • 1 — adding to current selection • 2 — removing from current selection • 3 — selecting within current selection

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[wie Eingabe]	The input layer with features selected

Pythoncode

Algorithm ID: `qgis:selectbyexpression`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Auswahl nach der Lage

Erstellt eine Auswahl in einem Vektorlayer.

Die Kriterien für die Auswahl von Features basieren auf der räumlichen Beziehung zwischen jedem Feature und den Features in eines zusätzlichen Layers.

Default menu: *Vector* ► *Research Tools*

Siehe auch

Extract by location, *Select within distance*

Exploring spatial relations

Geometrische Prädikate sind boolesche Funktionen, die verwendet werden, um die räumliche Beziehung eines Features zu einem anderen zu bestimmen, indem verglichen wird, ob und wie ihre Geometrien einen Teil des Raums teilen.

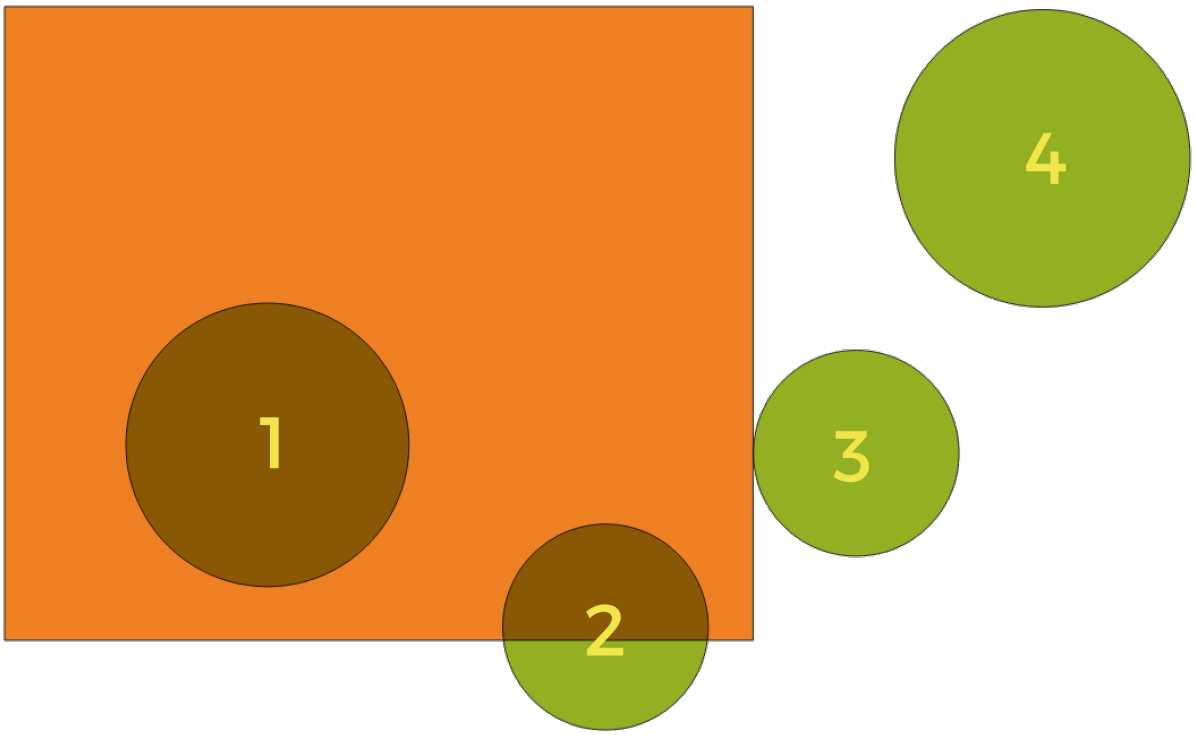


Abb. 24.164: Suche nach räumlichen Beziehungen zwischen Layern

Unter Verwendung der obigen Abbildung suchen wir nach den grünen Kreisen, indem wir sie räumlich mit dem orangefarbenen Rechteck-Feature vergleichen. Verfügbare geometrische Prädikate sind:

Schneidet
: Testet, ob eine Geometrie eine andere schneidet. Gibt 1 (wahr) zurück, wenn sich die Geometrien räumlich überschneiden (einen Teil des Raums teilen – überlappen oder berühren) und 0, wenn dies nicht der Fall ist. Im obigen Bild werden die Kreise 1, 2 und 3 zurückgegeben.

Enthält
: Gibt 1 (wahr) zurück, wenn und nur wenn keine Punkte von b außerhalb von a liegen und mindestens ein Punkt des Inneren von b innerhalb von a liegt. Im Bild wird kein Kreis zurückgegeben, sondern das Rechteck wäre

es, wenn man es umgekehrt suchen würde, da es den Kreis 1 komplett enthält. Das ist das Gegenteil von *sind innerhalb*.

Getrennt
Gibt 1 (wahr) zurück, wenn die Geometrien keinen Teil des Raums teilen (keine Überlappung, keine Berührung). Nur Kreis 4 wird zurückgegeben.

Gleich
Gibt 1 (wahr) zurück, wenn und nur wenn die Geometrien genau gleich sind. Es werden keine Kreise zurückgegeben.

Berührt
Testet, ob eine Geometrie eine andere berührt. Gibt 1 (wahr) zurück, wenn die Geometrien mindestens einen Punkt gemeinsam haben, aber ihre Innenräume sich nicht schneiden. Nur Kreis 3 wird zurückgegeben.

Überlappt
Testet, ob eine Geometrie eine andere überlappt. Gibt 1 (wahr) zurück, wenn die Geometrien sich den Raum teilen, dieselbe Dimension haben, aber nicht vollständig ineinander liegen. Nur Kreis 2 wird zurückgegeben.

Sind innerhalb
Testet, ob sich eine Geometrie in einer anderen befindet. Gibt 1 (wahr) zurück, wenn sich Geometrie a vollständig innerhalb von Geometrie b befindet. Nur Kreis 1 wird zurückgegeben.

Kreuzen
Gibt 1 (wahr) zurück, wenn die bereitgestellten Geometrien einige, aber nicht alle inneren Punkte gemeinsam haben und die tatsächliche Kreuzung eine geringere Dimension als die höchste bereitgestellte Geometrie hat. Beispielsweise wird eine Linie, die ein Polygon kreuzt, als Linie gekreuzt (wahr). Zwei Linien, die sich kreuzen, kreuzen sich als Punkt (wahr). Zwei Polygone kreuzen sich als Polygon (falsch). Im Bild werden keine Kreise zurückgegeben.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Select features from	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Where the features (geometric predicate)	`PREDICATE`	[Aufzählung] [Liste] Default: [0]	Type of spatial relation the input feature should have with an intersect feature so that it could be selected. One or more of: • 0 — intersect • 1 — contain • 2 — disjoint • 3 — equal • 4 — touch • 5 — overlap • 6 — are within • 7 — cross If more than one condition is chosen, at least one of them (OR operation) has to be met for a feature to be extracted.
By comparing to the features from	`INTERSECT`	[Vektor: Geometrie]	Intersection vector layer

Fortsetzung auf der nächsten Seite

Tab. 24.811 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Modify current selection by	METHOD	[Aufzählung] Standard: 0	How the selection of the algorithm should be managed. One of: • 0 — creating new selection • 1 — adding to current selection • 2 — selecting within current selection • 3 — removing from current selection

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	INPUT	[wie Eingabe]	The input layer with features selected

Pythoncode

Algorithm ID: `native:selectbylocation`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Select within distance

creates a selection in a vector layer. Features are selected wherever they are within the specified maximum distance from the features in an additional reference layer.

Siehe auch

Extract within distance, *Auswahl nach der Lage*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Select features from	INPUT	[Vektor: Geometrie]	Input vector layer to select features from
By comparing to the features from	REFERENCE	[Vektor: Geometrie]	Vector layer whose features closeness is used
Where the features are within	DISTANCE	[numeric: double] Default: 100.0	The maximum distance around reference features to select input features

Fortsetzung auf der nächsten Seite

Tab. 24.813 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Modify current selection by	METHOD	[Aufzählung] Standard: 0	How the selection of the algorithm should be managed. One of: • 0 — creating new selection • 1 — adding to current selection • 2 — selecting within current selection • 3 — removing from current selection

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	INPUT	[wie Eingabe]	The input layer with features selected

Pythoncode

Algorithm ID: `native:selectwithindistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.29 Vector table

Add autoincremental field

Adds a new integer field to a vector layer, with a sequential value for each feature.

This field can be used as a unique ID for features in the layer. The new attribute is not added to the input layer but a new layer is generated instead.

The initial starting value for the incremental series can be specified. Optionally, the incremental series can be based on grouping fields and a sort order for features can also be specified.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	INPUT	[Vektor: alle]	The input vector layer.
Field name	FIELD_NAME	[Text] Default: ‚AUTO‘	Name of the field with autoincremental values

Fortsetzung auf der nächsten Seite

Tab. 24.815 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Start values at Optional	START	[numerisch: Ganzzahl] Standard: 0	Choose the initial number of the incremental count
Modulus value Optional	MODULUS	[numerisch: Ganzzahl] Standard: 0	Specifying an optional modulus value will restart the count to START whenever the field value reaches the modulus value. 0 means no restart.
Group values by Optional	GROUP_FIELDS	[tablefield: any] [list]	Select grouping field(s): instead of a single count run for the whole layer, a separate count is processed for each value returned by the combination of these fields.
Sort expression Optional	SORT_EXPRESSIO	[Ausdruck]	Use an expression to sort the features in the layer either globally or if set, based on group fields.
Sort ascending	SORT_ASCENDING	[Boolean] Standard: True	When a sort expression is set, use this option to control the order in which features are assigned values.
Sort nulls first	SORT_NULLS_FIR	[Boolean] Standard: falsch	When a sort expression is set, use this option to set whether *Null* values are counted first or last.
Incremented	OUTPUT	[wie Eingabe] Standard: [Create temporary layer]	Specify the output vector layer with the auto increment field. *One of*: • Temporäre Ebene erstellen (TEMPORARY_OUTPUT) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Incremented	OUTPUT	[wie Eingabe]	Vector layer with auto incremental field

Pythoncode

Algorithm ID: native:addautoincrementalfield

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Add field to attributes table

Adds a new field to a vector layer.

The name and characteristics of the attribute are defined as parameters.

The new attribute is not added to the input layer but a new layer is generated instead.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	The input layer
Field name	`FIELD_NAME`	[Text]	Name of the new field
Field type	`FIELD_TYPE`	[Aufzählung] Standard: 0	Type of the new field. You can choose between: • 0 — Integer (32 bit) • 1 — Decimal (double) • 2 — Text (string) • 3 — Boolean • 4 — Date • 5 — Time • 6 — Date & Time • 7 — Binary Object (BLOB) • 8 — String List • 9 — Integer List • 10 — Decimal (double) List
Field length	`FIELD_LENGTH`	[numerisch: Ganzzahl] Default: 10	Length of the field
Field precision	`FIELD_PRECISIO`	[numerisch: Ganzzahl] Standard: 0	Precision of the field. Useful with Float field type.
Field alias Optional	`FIELD_ALIAS`	[Text]	Set a name to use as alias for the field. Not supported by all format types.
Field comment Optional	`FIELD_COMMENT`	[Text]	Store a comment describing the field. Not supported by all format types.
Added	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Added	`OUTPUT`	[wie Eingabe]	Vector layer with new field added

Pythoncode

Algorithm ID: `native:addfieldtoattributestable`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Add unique value index field

Takes a vector layer and an attribute and adds a new numeric field.

Values in this field correspond to values in the specified attribute, so features with the same value for the attribute will have the same value in the new numeric field.

This creates a numeric equivalent of the specified attribute, which defines the same classes.

The new attribute is not added to the input layer but a new layer is generated instead.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	The input layer.
Class field	`FIELD`	[tablefield: any]	Features that have the same value for this field will get the same index.
Output field name	`FIELD_NAME`	[Text] Default: ‚NUM_FIELD‘	Name of the new field containing the indexes.
Layer with index field	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Vector layer with the numeric field containing indexes. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Fortsetzung auf der nächsten Seite

Tab. 24.819 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Class summary	`SUMMARY_OUTPUT`	[vector: table] Standard: `[Ausgabe überspringen]`	Specify the table to contain the summary of the class field mapped to the corresponding unique value. *One of*: • Ausgabe übersrpingen • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Layer with index field	`OUTPUT`	[wie Eingabe]	Vector layer with the numeric field containing indexes.
Class summary	`SUMMARY_OUTPUT`	[vector: table]	Table with summary of the class field mapped to the corresponding unique value.

Pythoncode

Algorithm ID: `native:adduniquevalueindexfield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Add X/Y fields to layer

Adds X and Y (or latitude/longitude) fields to a point layer. The X/Y fields can be calculated in a different CRS to the layer (e.g. creating latitude/longitude fields for a layer in a projected CRS).

Erlaubt auch die *Insitu-Bearbeitung* von Punkt-Objekten

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Punkt]	The input layer.
Coordinate system	`CRS`	[KBS] Default: „EPSG:4326“	Coordinate reference system to use for the generated x and y fields.

Fortsetzung auf der nächsten Seite

Tab. 24.821 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Field prefix Optional	`PREFIX`	[Text]	Prefix to add to the new field names to avoid name collisions with fields in the input layer.
Added fields	`OUTPUT`	[Vektor: Punkt] Standard: `[Create temporary layer]`	Specify the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Added fields	`OUTPUT`	[Vektor: Punkt]	The output layer - identical to the input layer but with two new double fields, `x` and `y`.

Pythoncode

Algorithm ID: `native:addxyfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Advanced Python field calculator

Adds a new attribute to a vector layer, with values resulting from applying an expression to each feature.

The expression is defined as a Python function.

Warnung

This algorithm is a potential security risk if executed with unchecked inputs, and may result in system damage or data leaks.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer
Result field name	`FIELD_NAME`	[Text] Default: ‚NewField‘	Name of the new field
Field type	`FIELD_TYPE`	[Aufzählung] Standard: 0	Type of the new field. One of: • 0 — Integer (32 bit) • 1 — Decimal (double) • 2 — Text (string) • 3 — Boolean • 4 — Date • 5 — Time • 6 — Date & Time • 7 — Binary Object (BLOB) • 8 — String List • 9 — Integer List • 10 — Decimal (double) List
Field length	`FIELD_LENGTH`	[numerisch: Ganzzahl] Default: 10	Length of the field
Field precision	`FIELD_PRECISIO`	[numerisch: Ganzzahl] Default: 3	Precision of the field. Useful with Float field type.
Global expression Optional	`GLOBAL`	[Text]	The code in the global expression section will be executed only once before the calculator starts iterating through all the features of the input layer. Therefore, this is the correct place to import necessary modules or to calculate variables that will be used in subsequent calculations.

Formula	`FORMULA`	[Text]	The Python formula to evaluate. Example: To calculate the area of an input polygon layer you can add:
			`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()`

Calculated	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the vector layer with the new calculated field. One of: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Calculated	`OUTPUT`	[wie Eingabe]	Vector layer with the new calculated field

Pythoncode

Algorithm ID: `qgis:advancedpythonfieldcalculator`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Drop field(s)

Takes a vector layer and generates a new one that has the same features but without the selected columns.

Siehe auch

Retain fields

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Input vector layer to drop field(s) from
Fields to drop	`COLUMN`	[tablefield: any] [list]	The field(s) to drop
Remaining fields	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer with the remaining fields. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Remaining fields	`OUTPUT`	[wie Eingabe]	Vector layer with the remaining fields

Pythoncode

Algorithm ID: `native:deletecolumn`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Explode HStore Field

Creates a copy of the input layer and adds a new field for every unique key in the HStore field.

The expected field list is an optional comma separated list. If this list is specified, only these fields are added and the HStore field is updated. By default, all unique keys are added.

The PostgreSQL HStore is a simple key-value store used in PostgreSQL and GDAL (when reading an OSM file with the `other_tags` field.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
HStore field	`FIELD`	[tablefield: any]	The field(s) to drop
Expected list of fields separated by a comma Optional	`EXPECTED_FIELD`	[Text] Default: ‚‘	Comma-separated list of fields to extract. The HStore field will be updated by removing these keys.
Exploded	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specify the output vector layer. One of: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Exploded	`OUTPUT`	[wie Eingabe]	Output vector layer

Pythoncode

Algorithm ID: `native:explodehstorefield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Extract binary field

Extracts contents from a binary field, saving them to individual files. Filenames can be generated using values taken from an attribute in the source table or based on a more complex expression.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Input vector layer containing the binary data
Binary field	`FIELD`	[tablefield: any]	Field containing the binary data
File name	`FILENAME`	[Ausdruck]	Field or expression-based text to name each output file
Destination folder	`FOLDER`	[Verzeichnis] Standard: `[In temporärem Verzeichnis speichern]`	Folder in which to store the output files. *One of*: • In einem temporären Verzeichnis speichern • In Verzeichnis speichern

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Folder	`FOLDER`	[Verzeichnis]	The folder that contains the output files.

Pythoncode

Algorithm ID: `native:extractbinary`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Field calculator

Opens the field calculator (see *Ausdrücke*). You can use all the supported expressions and functions.

A new layer is created with the result of the expression.

The field calculator is very useful when used in *Ein Modell erstellen*.

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	The layer to calculate on
Output field name	`FIELD_NAME`	[Text]	The name of the field for the results
Output field type	`FIELD_TYPE`	[Aufzählung] Standard: 0	The type of the field. One of: • 0 — Decimal (double) • 1 — Integer (32 bit) • 2 — Text (string) • 3 — Date • 4 — Time • 5 — Date & Time • 6 — Boolean • 7 — Binary Object (BLOB) • 8 — String List • 9 — Integer List • 10 — Decimal (double) List
Output field width	`FIELD_LENGTH`	[numerisch: Ganzzahl] Standard: 0	The length of the result field (minimum 0)
Field precision	`FIELD_PRECISIO`	[numerisch: Ganzzahl] Standard: 0	The precision of the result field (minimum 0, maximum 15)
Create new field	`NEW_FIELD`	[Boolean] Standard: True	Should the result field be a new field
Formula	`FORMULA`	[Ausdruck]	The formula to use to calculate the result

Fortsetzung auf der nächsten Seite

Tab. 24.831 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Calculated	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specification of the output layer. • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Calculated	`OUTPUT`	[wie Eingabe]	Output layer with the calculated field values

Pythoncode

Algorithm ID: `native:fieldcalculator`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Refactor fields

Allows editing the structure of the attribute table of a vector layer.

Fields can be modified in their type and name, using a fields mapping.

The original layer is not modified. A new layer is generated, which contains a modified attribute table, according to the provided fields mapping.

Bemerkung

When using a template layer with *constraints* on fields, the information is displayed in the widget with a coloured background and tooltip. Treat this information as a hint during configuration. No constraints will be added on an output layer nor will they be checked or enforced by the algorithm.

The Refactor fields algorithm allows to:

- Change field names and types
- Add and remove fields
- Reorder fields
- Calculate new fields based on expressions
- Load field list from another layer

Bemerkung

This algorithm uses ellipsoid based measurements and respects the current *ellipsoid settings*.

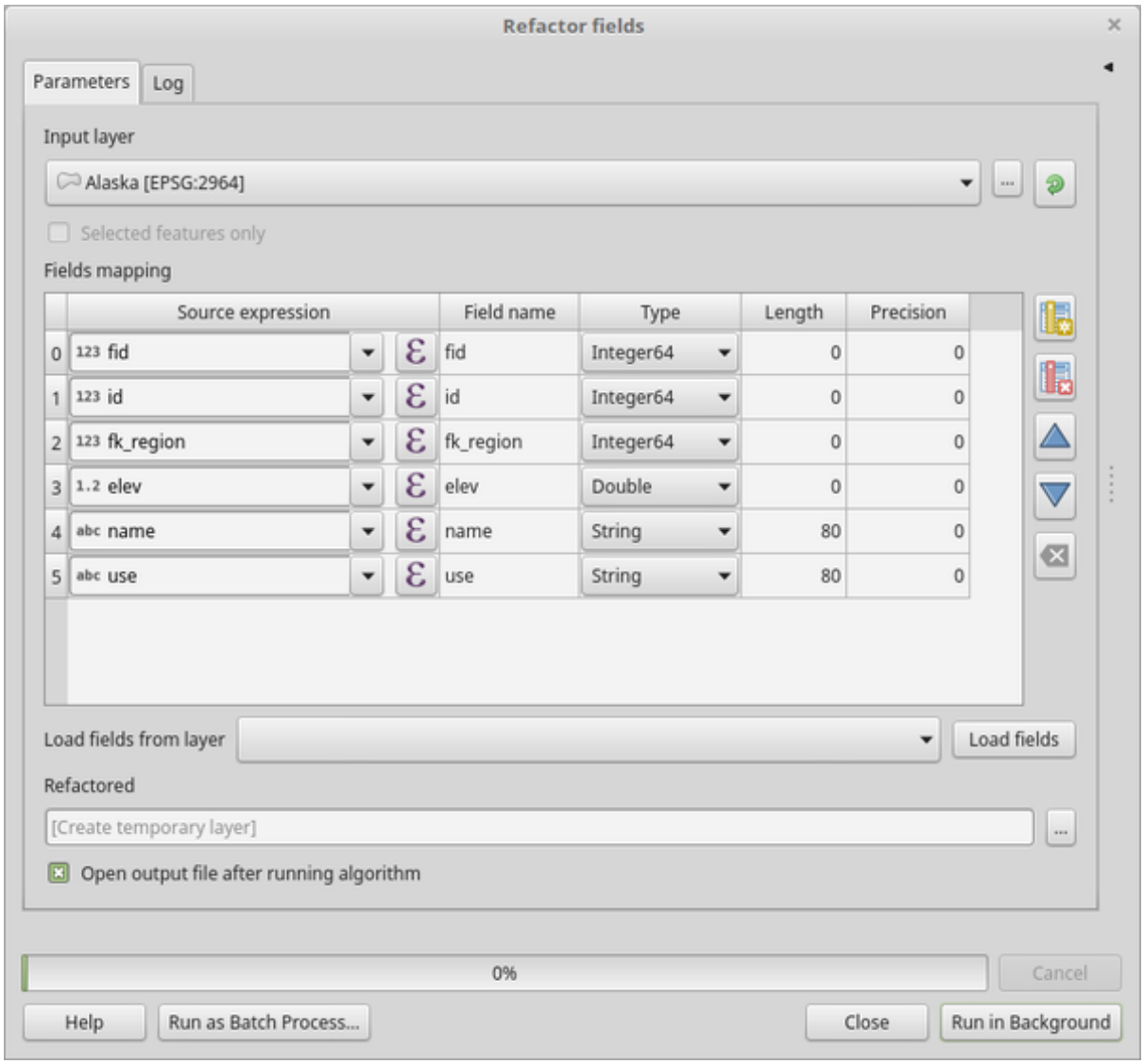


Abb. 24.165: Refactor fields dialog

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	The layer to modify

Fortsetzung auf der nächsten Seite

Tab. 24.833 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Fields mapping	`FIELDS_MAPPING`	[list]	List of output fields with their definitions. The embedded table lists all the fields of the source layer and allows you to edit them: • Click to create a new field. • Click to remove a field. • Use and to change the selected field order. • Click to reset to the default view. • Click to invert the selection in the fields list. For each of the fields you'd like to reuse, you need to fill the following options: ***Source expression*** **(`expression`) [expression]** Field or expression from the input layer. ***Name*** **(`name`) [string]** Name of the field in the output layer. By default input field name is kept. ***Type*** **(`type`) [enumeration]** Data type of the output field. Available types may not be compatible with the output layer provider. One of: **Achtung** For certain field types, e.g. lists, an extra `sub_type` parameter helps refine the specific type of the data. It is automatically set in the GUI but may be needed if you're running the algorithm in Python or from the command line. • 1 — Boolescher Wert • 2 — Ganzzahl (32bit) • 4 — Ganzzahl (64bit) • 6 — Dezimalzahl (double) • Liste von Ganzzahlen • 9 — Liste von Ganzzahlen (64bit) • 9 — Liste von Dezimalzahlen (double) • 10 — Zeichenkette (string) • 11 — Liste von Zeichenketten • 12 — Binär-Objekt (BLOB) • 14 — Datum • 15 — Uhrzeit • 16 — Datum & Uhrzeit ***Sub-type*** **(`sub_type`) [enumeration]** For certain field types, e.g. lists, this parameter helps refine the specific `type` of the data. It is automatically set in the GUI but may be needed if you're running the algorithm in Python or from the command line. *One of*: • 2 — Liste von Ganzzahlen • 4 — Liste von Ganzzahlen

Tab. 24.833 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Refactored	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specification of the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Refactored	`OUTPUT`	[wie Eingabe]	Output layer with refactored fields

Pythoncode

Algorithm ID: `native:refactorfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Rename field

Renames an existing field from a vector layer.

The original layer is not modified. A new layer is generated where the attribute table contains the renamed field.

Siehe auch

Refactor fields

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	The input vector layer
Field to rename	`FIELD`	[tablefield: any]	The field to be altered
New field name	`NEW_NAME`	[Text]	The new field name

Fortsetzung auf der nächsten Seite

Tab. 24.835 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Renamed	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specification of the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Renamed	`OUTPUT`	[wie Eingabe]	Output layer with the renamed field

Pythoncode

Algorithm ID: `native:renametablefield`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Retain fields

Takes a vector layer and generates a new one that retains only the selected fields. All other fields will be dropped.

Siehe auch

Drop field(s)

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	The input vector layer
Fields to retain	`FIELDS`	[tablefield: any] [list]	List of fields to keep in the layer

Fortsetzung auf der nächsten Seite

Tab. 24.837 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Retained fields	`OUTPUT`	[wie Eingabe] Standard: `[Create temporary layer]`	Specification of the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Retained fields	`OUTPUT`	[wie Eingabe]	Output layer with the retained fields

Pythoncode

Algorithm ID: `native:retainfields`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Text to float

Modifies the type of a given attribute in a vector layer, converting a text attribute containing numeric strings into a numeric attribute (e.g. ‚1‘ to `1.0`).

The algorithm creates a new vector layer so the source one is not modified.

If the conversion is not possible the selected column will have `NULL` values.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	The input vector layer.
Text attribute to convert to float	`FIELD`	[tablefield: string]	The string field for the input layer that is to be converted to a float field.

Fortsetzung auf der nächsten Seite

Tab. 24.839 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Float from text	`OUTPUT`	[wie Eingabe] Default: `[Create Temporary Layer]`	Specify the output layer. *One of*: • Temporäre Ebene erstellen (`TEMPORARY_OUTPUT`) • In Datei speichern… • In GeoPackage speichern… • In Datenbank speichern… • zu Layer hinzufügen… Die Kodierung der Datei kann ebenfalls hier geändert werden.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Float from text	`OUTPUT`	[wie Eingabe]	Output vector layer with the string field converted into a float field

Pythoncode

Algorithm ID: `qgis:texttofloat`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.1.30 Vector Tiles

Download vector tiles

Downloads vector tiles of an input vector tile layer and saves them in a local vector tile file.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector tiles]	A vector tile layer to extract some tiles from

Fortsetzung auf der nächsten Seite

Tab. 24.841 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Extent	`EXTENT`	[Ausdehnung]	Specify the spatial extent of the downloaded area. It will internally be extended to a multiple of the tile size. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Maximum zoom level to download	`MAX_ZOOM`	[numerisch: Ganzzahl] Default: 10	Defines how far to zoom in and fetch data from the tiles
Tile limit	`TILE_LIMIT`	[numerisch: Ganzzahl] Default: 100	Maximum number of tiles to download, considering the zoom levels and the extent.
Ergebnis	`OUTPUT`	[vector tiles] Default: [Save to temporary file]	Specification of the output vector tile file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ergebnis	`OUTPUT`	[vector tiles]	A local vector tile file storing the downloaded tiles.

Pythoncode

Algorithm ID: `native:downloadvectortiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Write vector tiles (MBTiles)

Exports one or more vector layers to vector tiles, a data format optimized for fast map rendering and small data size.

MBTiles is a specification for storing tiled map data in SQLite databases for immediate usage and for transfer. MBTiles files are known as tilesets.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: any] [list]	A list of layers to combine to generate the vector tiles
Minimum zoom level	`MIN_ZOOM`	[numerisch: Ganzzahl] Standard: 0	The lowest zoom level for which the tileset provides data. Set between 0 and 24.
Maximum zoom level	`MAX_ZOOM`	[numerisch: Ganzzahl] Default: 3	The highest zoom level for which the tileset provides data. Set between 0 and 24.
Extent Optional	`EXTENT`	[Ausdehnung] Standard: Nicht gesetzt	The maximum extent of the rendered map area. Bounds must define an area covered by all zoom levels.
Metadata: Name Optional	`META_NAME`	[Text]	Name of the tileset
Metadata: Description Optional	`META_DESCRIPTI`	[Text]	A description of the tileset's contents
Metadata: Attribution Optional	`META_ATTRIBUTI`	[Text]	An attribution string, which explains the sources of data and/or style for the map.
Metadata: Version Optional	`META_VERSION`	[Text]	The version of the tileset. This refers to a revision of the tileset itself, not of the MBTiles specification.
Metadata: Type Optional	`META_TYPE`	[Text]	Type of tileset. Possible values are `overlay` or `baselayer`.
Metadata: Center Optional	`META_CENTER`	[Text]	The center (string of comma-separated numbers: the longitude, latitude, and zoom level) of the default view of the map. Example: `-122.1906,37.7599,11`

Fortsetzung auf der nächsten Seite

Tab. 24.843 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Destination MBTiles	`OUTPUT`	[vector tiles] Default: [Save to temporary file]	Specification of the output MBTiles file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Destination MBTiles	`OUTPUT`	[Datei]	Output vector tiles `.mbtiles` file.

Pythoncode

Algorithm ID: `native:writevectortiles_mbtiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Write vector tiles (XYZ)

Exports one or more vector layers to vector tiles, a data format optimized for fast map rendering and small data size.

Parameter

Bezeichnung	Name	Typ	Beschreibung
File template	`XYZ_TEMPLATE`	[Text] Default: ‚{z}/{x}/{y}.pbf‘	Template to generate the vector tiles url
Eingabelayer	`INPUT`	[vector: any] [list]	A list of layers to combine to generate the vector tiles
Minimum zoom level	`MIN_ZOOM`	[numerisch: Ganzzahl] Standard: 0	The lowest zoom level for which the tileset provides data. Set between 0 and 24.
Maximum zoom level	`MAX_ZOOM`	[numerisch: Ganzzahl] Default: 3	The highest zoom level for which the tileset provides data. Set between 0 and 24.
Extent Optional	`EXTENT`	[Ausdehnung] Standard: Nicht gesetzt	The maximum extent of the rendered map area. Bounds must define an area covered by all zoom levels.

Fortsetzung auf der nächsten Seite

Tab. 24.845 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Ausgabeverzeichnis	`OUTPUT_DIRECTO`	[Verzeichnis] Default: [Save to temporary folder]	Specification of the output vector tiles folder. *One of*: • In einem temporären Verzeichnis speichern • In Verzeichnis speichern

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabeverzeichnis	`OUTPUT_DIRECTO`	[Verzeichnis]	A folder containing different subsets of the vector tiles files (`.pbf`) stored in subfolders corresponding to the zoom levels.

Pythoncode

Algorithm ID: `native:writevectortiles_xyz`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.2 GDAL Verarbeitungsalgorithmen

GDAL (Geospatial Data Abstraction Library) is a translator library for raster and vector geospatial data formats. Algorithms in the Processing Framework are derived from the GDAL raster programs and GDAL vector programs.

24.2.1 Raster analysis

Aspect

Generates an aspect map from any GDAL-supported elevation raster. Aspect is the compass direction that a slope faces. The pixels will have a value from 0-360° measured in degrees from north indicating the azimuth. On the northern hemisphere, the north side of slopes is often shaded (small azimuth from 0°-90°), while the southern side receives more solar radiation (higher azimuth from 180°-270°).

This algorithm is derived from the GDAL DEM utility.

Default menu: *Raster* ► *Analysis*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input elevation raster layer
Band number	`BAND`	[raster band] Standard: 1	The number of the band to use as elevation
Return trigonometric angle instead of azimuth	`TRIG_ANGLE`	[Boolean] Standard: falsch	Activating the trigonometric angle results in different categories: 0° (East), 90° (North), 180° (West), 270° (South).
Return 0 for flat instead of -9999	`ZERO_FLAT`	[Boolean] Standard: falsch	Activating this option will insert a 0-value for the value -9999 on flat areas.
Compute edges	`COMPUTE_EDGES`	[Boolean] Standard: falsch	Generates edges from the elevation raster
Use Zevenbergen&Thorn formula instead of the Horn's one	`ZEVENBERGEN`	[Boolean] Standard: falsch	Activates Zevenbergen&Thorne formula for smooth landscapes
Aspect	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Aspect	`OUTPUT`	[Raster]	Output raster with angle values in degrees

Pythoncode

Algorithm ID: `gdal:aspect`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Color relief

Generates a color relief map from any GDAL-supported elevation raster. Color reliefs can particularly be used to depict elevations. The Algorithm outputs a 4-band raster with values computed from the elevation and a text-based color configuration file. By default, the colors between the given elevation values are blended smoothly and the result is a nice colorized elevation raster.

This algorithm is derived from the GDAL DEM utility.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input elevation raster layer
Band number	`BAND`	[raster band] Standard: 1	The number of the band to use as elevation
Compute edges	`COMPUTE_EDGES`	[Boolean] Standard: falsch	Generates edges from the elevation raster
Color configuration file	`COLOR_TABLE`	[Datei]	A text-based color configuration file
Matching mode	`MATCH_MODE`	[Aufzählung] Default: 2	One of: • 0 — Use strict color matching • 1 — Use closest RGBA quadruples • 2 — Use smoothly blended colours
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚Äò	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options
Color relief	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Color relief	`OUTPUT`	[Raster]	A 4-band output raster

Pythoncode

Algorithm ID: `gdal:colorrelief`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Fill NoData

Fill raster regions with NoData values by interpolation from edges. The values for the NoData regions are calculated by the surrounding pixel values using inverse distance weighting. After the interpolation a smoothing of the results takes place. Input can be any GDAL-supported raster layer. This algorithm is generally suitable for interpolating missing regions of fairly continuously varying rasters (such as elevation models for instance). It is also suitable for filling small holes and cracks in more irregularly varying images (like airphotos). It is generally not so great for interpolating a raster from sparse point data.

This algorithm is derived from the GDAL fillnodata utility.

Default menu: *Raster* ► *Analysis*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer
Band number	`BAND`	[raster band] Standard: 1	The band to operate on. NoData values must be represented by the value 0.
Maximum distance (in pixels) to search out for values to interpolate	`DISTANCE`	[numerisch: Ganzzahl] Default: 10	The number of pixels to search in all directions to find values to interpolate from
Number of smoothing iterations to run after the interpolation	`ITERATIONS`	[numerisch: Ganzzahl] Standard: 0	The number of 3x3 filter passes to run (0 or more) to smoothen the results of the interpolation.
Validity mask	`MASK_LAYER`	[Raster]	A raster layer that defines the areas to fill.

Fortsetzung auf der nächsten Seite

Tab. 24.852 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Filled	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Filled	`OUTPUT`	[Raster]	Output raster

Pythoncode

Algorithm ID: `gdal:fillnodata`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Grid (Data metrics)

Computes some data metrics using the specified window and output grid geometry.

This algorithm is derived from the GDAL grid utility.

Default menu: *Raster* ► *Analysis*

Siehe auch

GDAL grid tutorial

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Point layer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Data metric to use	`METRIC`	[Aufzählung] Standard: 0	One of: • 0 — Minimum, minimum value found in grid node search ellipse • 1 — Maximum, maximum value found in grid node search ellipse • 2 — Range, a difference between the minimum and maximum values found in grid node search ellipse • 3 — Count, a number of data points found in grid node search ellipse • 4 — Average distance, an average distance between the grid node (center of the search ellipse) and all of the data points found in grid node search ellipse • 5 — Average distance between points, an average distance between the data points found in grid node search ellipse. The distance between each pair of points within ellipse is calculated and average of all distances is set as a grid node value
The first radius of search ellipse	`RADIUS_1`	[numerisch: Double] Default: 0.0	The first radius (X axis if rotation angle is 0) of the search ellipse
The second radius of search ellipse	`RADIUS_2`	[numerisch: Double] Default: 0.0	The second radius (Y axis if rotation angle is 0) of the search ellipse
Angle of search ellipse rotation in degrees (counter clockwise)	`ANGLE`	[numerisch: Double] Default: 0.0	Angle of ellipse rotation in degrees. Ellipse rotated counter clockwise.

Fortsetzung auf der nächsten Seite

Tab. 24.855 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Minimum number of data points to use	`MIN_POINTS`	[numerisch: Double] Default: 0.0	Minimum number of data points to average. If less amount of points found the grid node considered empty and will be filled with NoData marker.
NoData	`NODATA`	[numerisch: Double] Default: 0.0	NoData marker to fill empty points
Interpolated (data metrics)	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer with interpolated values. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Z value from field Optional	`Z_FIELD`	[tablefield: numeric]	Field for the interpolation
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ,‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Fortsetzung auf der nächsten Seite

Tab. 24.856 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the data type of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Interpolated (data metrics)	`OUTPUT`	[Raster]	Output raster with interpolated values

Pythoncode

Algorithm ID: `gdal:griddatametrics`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Grid (IDW with nearest neighbor searching)

Computes the Inverse Distance to a Power gridding combined to the nearest neighbor method. Ideal when a maximum number of data points to use is required.

This algorithm is derived from the GDAL grid utility.

Siehe auch

GDAL grid tutorial

Parameter

Bezeichnung	Name	Typ	Beschreibung
Point layer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Weighting power	`POWER`	[numerisch: Double] Default: 2.0	Weighting power
Smoothing	`SMOOTHING`	[numerisch: Double] Default: 0.0	Smoothing parameter
The radius of the search circle	`RADIUS`	[numerisch: Double] Standard: 1,0	The radius of the search circle
Maximum number of data points to use	`MAX_POINTS`	[numerisch: Ganzzahl] Default: 12	Do not search for more points than this number.
Minimum number of data points to use	`MIN_POINTS`	[numerisch: Ganzzahl] Standard: 0	Minimum number of data points to average. If less amount of points found the grid node considered empty and will be filled with NoData marker.
NoData	`NODATA`	[numerisch: Double] Default: 0.0	NoData marker to fill empty points
Interpolated (IDW with NN search)	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer with interpolated values. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Z value from field Optional	`Z_FIELD`	[tablefield: numeric]	Field for the interpolation

Fortsetzung auf der nächsten Seite

Tab. 24.859 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ,‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the data type of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Interpolated (IDW with NN search)	`OUTPUT`	[Raster]	Output raster with interpolated values

Pythoncode

Algorithm ID: `gdal:gridinversedistancenearestneighbor`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Grid (Inverse distance to a power)

The Inverse Distance to a Power gridding method is a weighted average interpolator.

You should supply the input arrays with the scattered data values including coordinates of every data point and output grid geometry. The function will compute interpolated value for the given position in output grid.

This algorithm is derived from the GDAL grid utility.

Default menu: *Raster* ► *Analysis*

Siehe auch

GDAL grid tutorial

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Point layer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Weighting power	`POWER`	[numerisch: Double] Default: 2.0	Weighting power
Smothing	`SMOOTHING`	[numerisch: Double] Default: 0.0	Smoothing parameter
The first radius of search ellipse	`RADIUS_1`	[numerisch: Double] Default: 0.0	The first radius (X axis if rotation angle is 0) of the search ellipse
The second radius of search ellipse	`RADIUS_2`	[numerisch: Double] Default: 0.0	The second radius (Y axis if rotation angle is 0) of the search ellipse
Angle of search ellipse rotation in degrees (counter clockwise)	`ANGLE`	[numerisch: Double] Default: 0.0	Angle of ellipse rotation in degrees. Ellipse rotated counter clockwise.
Maximum number of data points to use	`MAX_POINTS`	[numerisch: Ganzzahl] Standard: 0	Do not search for more points than this number.

Fortsetzung auf der nächsten Seite

Tab. 24.861 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Minimum number of data points to use	`MIN_POINTS`	[numerisch: Ganzzahl] Standard: 0	Minimum number of data points to average. If less amount of points found the grid node considered empty and will be filled with NoData marker.
NoData	`NODATA`	[numerisch: Double] Default: 0.0	NoData marker to fill empty points
Interpolated (IDW)	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer with interpolated values. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Z value from field Optional	`Z_FIELD`	[tablefield: numeric]	Field for the interpolation
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ,‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Fortsetzung auf der nächsten Seite

Tab. 24.862 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the data type of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Interpolated (IDW)	`OUTPUT`	[Raster]	Output raster with interpolated values

Pythoncode

Algorithm ID: `gdal:gridinversedistance`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Grid (Linear)

The Linear method perform linear interpolation by computing a Delaunay triangulation of the point cloud, finding in which triangle of the triangulation the point is, and by doing linear interpolation from its barycentric coordinates within the triangle. If the point is not in any triangle, depending on the radius, the algorithm will use the value of the nearest point or the NoData value.

This algorithm is derived from the GDAL grid utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Point layer	`INPUT`	[Vektor: Punkt]	Input point vector layer
Search distance	`RADIUS`	[numerisch: Double] Default: -1.0	In case the point to be interpolated does not fit into a triangle of the Delaunay triangulation, use that maximum distance to search a nearest neighbour, or use NoData otherwise. If set to `-1`, the search distance is infinite. If set to `0`, NoData value will be used.
NoData	`NODATA`	[numerisch: Double] Default: 0.0	NoData marker to fill empty points
Interpolated (Linear)	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer with interpolated values. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Z value from field Optional	`Z_FIELD`	[tablefield: numeric]	Field for the interpolation
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Fortsetzung auf der nächsten Seite

Tab. 24.865 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the data type of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Interpolated (Linear)	`OUTPUT`	[Raster]	Output raster with interpolated values

Pythoncode

Algorithm ID: `gdal:gridlinear`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Grid (Moving average)

The Moving Average is a simple data averaging algorithm. It uses a moving window of elliptic form to search values and averages all data points within the window. Search ellipse can be rotated by specified angle, the center of ellipse located at the grid node. Also the minimum number of data points to average can be set, if there are not enough points in window, the grid node considered empty and will be filled with specified NoData value.

This algorithm is derived from the GDAL grid utility.

Default menu: *Raster* ► *Analysis*

Siehe auch

GDAL grid tutorial

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Point layer	`INPUT`	[Vektor: Punkt]	Input point vector layer
The first radius of search ellipse	`RADIUS_1`	[numerisch: Double] Default: 0.0	The first radius (X axis if rotation angle is 0) of the search ellipse
The second radius of search ellipse	`RADIUS_2`	[numerisch: Double] Default: 0.0	The second radius (Y axis if rotation angle is 0) of the search ellipse
Angle of search ellipse rotation in degrees (counter clockwise)	`ANGLE`	[numerisch: Double] Default: 0.0	Angle of ellipse rotation in degrees. Ellipse rotated counter clockwise.
Minimum number of data points to use	`MIN_POINTS`	[numerisch: Ganzzahl] Standard: 0	Minimum number of data points to average. If less amount of points found the grid node considered empty and will be filled with NoData marker.
NoData	`NODATA`	[numerisch: Double] Default: 0.0	NoData marker to fill empty points
Interpolated (moving average)	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Z value from field Optional	`Z_FIELD`	[tablefield: numeric]	Field for the interpolation
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the data type of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Interpolated (moving average)	`OUTPUT`	[Raster]	Output raster with interpolated values

Pythoncode

Algorithm ID: `gdal:gridaverage`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Grid (Nearest neighbor)

The Nearest Neighbor method doesn't perform any interpolation or smoothing, it just takes the value of nearest point found in grid node search ellipse and returns it as a result. If there are no points found, the specified NoData value will be returned.

This algorithm is derived from the GDAL grid utility.

Default menu: *Raster* ► *Analysis*

Siehe auch

GDAL grid tutorial

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Point layer	`INPUT`	[Vektor: Punkt]	Input point vector layer
The first radius of search ellipse	`RADIUS_1`	[numerisch: Double] Default: 0.0	The first radius (X axis if rotation angle is 0) of the search ellipse
The second radius of search ellipse	`RADIUS_2`	[numerisch: Double] Default: 0.0	The second radius (Y axis if rotation angle is 0) of the search ellipse
Angle of search ellipse rotation in degrees (counter clockwise)	`ANGLE`	[numerisch: Double] Default: 0.0	Angle of ellipse rotation in degrees. Ellipse rotated counter clockwise.
NoData	`NODATA`	[numerisch: Double] Default: 0.0	NoData marker to fill empty points
Interpolated (Nearest neighbour)	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer with interpolated values. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Z value from field Optional	`Z_FIELD`	[tablefield: numeric]	Field for the interpolation
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the data type of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Interpolated (Nearest neighbour)	`OUTPUT`	[Raster]	Output raster with interpolated values

Pythoncode

Algorithm ID: `gdal:gridnearestneighbor`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Schummerung

Outputs a raster with a nice shaded relief effect. It's very useful for visualizing the terrain. You can optionally specify the azimuth and altitude of the light source, a vertical exaggeration factor and a scaling factor to account for differences between vertical and horizontal units.

This algorithm is derived from the GDAL DEM utility .

Default menu: *Raster* ► *Analysis*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input Elevation raster layer
Band number	`BAND`	[raster band] Standard: 1	Band containing the elevation information
Z factor (vertical exaggeration)	`Z_FACTOR`	[numerisch: Double] Standard: 1,0	The factor exaggerates the height of the output elevation raster
Scale (ratio of vertical units to horizontal)	`SCALE`	[numerisch: Double] Standard: 1,0	The ratio of vertical units to horizontal units
Azimuth of the light	`AZIMUTH`	[numerisch: Double] Default: 315.0	Defines the azimuth of the light shining on the elevation raster in degrees. If it comes from the top of the raster the value is 0, if it comes from the east it is 90 a.s.o.
Altitude of the light	`ALTITUDE`	[numerisch: Double] Default: 45.0	Defines the altitude of the light, in degrees. 90 if the light comes from above the elevation raster, 0 if it is raking light.
Compute edges	`COMPUTE_EDGES`	[Boolean] Standard: falsch	Generates edges from the elevation raster
Use Zevenbergen&Thorn formula (instead of the Horn's one)	`ZEVENBERGEN`	[Boolean] Standard: falsch	Activates Zevenbergen&Thorne formula for smooth landscapes
Combined shading	`COMBINED`	[Boolean] Standard: falsch	
Multidirectional shading	`MULTIDIRECTION`	[Boolean] Standard: falsch	

Fortsetzung auf der nächsten Seite

Tab. 24.873 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Hillshade	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer with interpolated values. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Hillshade	`OUTPUT`	[Raster]	Output raster with interpolated values

Pythoncode

Algorithm ID: `gdal:hillshade`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Near black

Converts nearly black/white borders to black.

This algorithm will scan an image and try to set all pixels that are nearly or exactly black, white or one or more custom colors around the collar to black or white. This is often used to „fix up“ lossy compressed airphotos so that color pixels can be treated as transparent when mosaicking.

This algorithm is derived from the GDAL nearblack utility.

Default menu: *Raster* ► *Analysis*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input Elevation raster layer
How far from black (white)	`NEAR`	[numerisch: Ganzzahl] Default: 15	Select how far from black, white or custom colors the pixel values can be and still considered near black, white or custom color.
Search for nearly white pixels instead of nearly black	`WHITE`	[Boolean] Standard: falsch	Search for nearly white (255) pixels instead of nearly black pixels
Nearblack	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Nearblack	`OUTPUT`	[Raster]	Output raster

Pythoncode

Algorithm ID: `gdal:nearblack`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Proximity (raster distance)

Generates a raster proximity map indicating the distance from the center of each pixel to the center of the nearest pixel identified as a target pixel. Target pixels are those in the source raster for which the raster pixel value is in the set of target pixel values.

This algorithm is derived from the GDAL proximity utility.

Default menu: *Raster* ► *Analysis*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input Elevation raster layer
Band number	`BAND`	[raster band] Standard: 1	Band containing the elevation information
List of target pixels Optional	`VALUES`	[Text] Default: ‚‘	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.
Distance units	`UNITS`	[Aufzählung] Standard: 1	Indicate whether distances generated should be in pixel or georeferenced coordinates. One of: • 0 — Georeferenced coordinates • 1 — Pixel coordinates
The maximum distance to generate Optional	`MAX_DISTANCE`	[numerisch: Double] Default: 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.

Fortsetzung auf der nächsten Seite

Tab. 24.879 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Value to apply to pixels within the maximum distance Optional	`REPLACE`	[numerisch: Double] Default: 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.
NoData value to use for the destination proximity raster Optional	`NODATA`	[numerisch: Double] Default: 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.
Proximity map	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ,‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Fortsetzung auf der nächsten Seite

Tab. 24.880 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the data type of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Proximity map	`OUTPUT`	[Raster]	Output raster

Pythoncode

Algorithm ID: `gdal:proximity`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Roughness

Outputs a single-band raster with values computed from the elevation. Roughness is the degree of irregularity of the surface. It's calculated by the largest inter-cell difference of a central pixel and its surrounding cell. The determination of the roughness plays a role in the analysis of terrain elevation data, it's useful for calculations of the river morphology, in climatology and physical geography in general.

This algorithm is derived from the GDAL DEM utility.

Default menu: *Raster* ► *Analysis*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input elevation raster layer
Band number	`BAND`	[raster band] Standard: 1	The number of the band to use as elevation
Compute edges	`COMPUTE_EDGES`	[Boolean] Standard: falsch	Generates edges from the elevation raster
Roughness	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Roughness	`OUTPUT`	[Raster]	Single-band output roughness raster. The value -9999 is used as NoData value.

Pythoncode

Algorithm ID: `gdal:roughness`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Sieve

Removes raster polygons smaller than a provided threshold size (in pixels) and replaces them with the pixel value of the largest neighbour polygon. It is useful if you have a large amount of small areas on your raster map.

This algorithm is derived from the GDAL sieve utility.

Default menu: *Raster* ► *Analysis*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input elevation raster layer
Threshold	`THRESHOLD`	[numerisch: Ganzzahl] Default: 10	Only raster polygons smaller than this size will be removed
Use 8-connectedness	`EIGHT_CONNECTE`	[Boolean] Standard: falsch	If False, consider as neighbours only the (4) pixels that connect the edges horizontally or vertically. If True, every pixel that touches an edge or a corner is considered as neighbour.
Do not use the default validity mask for the input band	`NO_MASK`	[Boolean] Standard: falsch	
Validity mask Optional	`MASK_LAYER`	[Raster]	Validity mask to use instead of the default
Sieved	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional command-line parameters Optional	EXTRA	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Sieved	OUTPUT	[Raster]	Output raster layer.

Pythoncode

Algorithm ID: `gdal:sieve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Slope

Generates a slope map from any GDAL-supported elevation raster. Slope is the angle of inclination to the horizontal. You have the option of specifying the type of slope value you want: degrees or percent slope.

This algorithm is derived from the GDAL DEM utility.

Default menu: *Raster* ► *Analysis*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	INPUT	[Raster]	Input Elevation raster layer
Band number	BAND	[raster band] Standard: 1	Band containing the elevation information
Ratio of vertical units to horizontal	SCALE	[numerisch: Double] Standard: 1,0	The ratio of vertical units to horizontal units
Express slope as percent (instead of degrees)	AS_PERCENT	[Boolean] Standard: falsch	Express slope as percent instead of degrees

Fortsetzung auf der nächsten Seite

Tab. 24.888 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Compute edges	`COMPUTE_EDGES`	[Boolean] Standard: falsch	Generates edges from the elevation raster
Use Zevenbergen&Thorne formula (instead of the Horn's one)	`ZEVENBERGEN`	[Boolean] Standard: falsch	Activates Zevenbergen&Thorne formula for smooth landscapes
Slope	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Slope	`OUTPUT`	[Raster]	Output raster

Pythoncode

Algorithm ID: `gdal:slope`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Terrain Ruggedness Index (TRI)

Outputs a single-band raster with values computed from the elevation. TRI stands for Terrain Ruggedness Index, which is defined as the mean difference between a central pixel and its surrounding cells.

This algorithm is derived from the GDAL DEM utility.

Default menu: *Raster* ► *Analysis*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input elevation raster layer
Band number	`BAND`	[raster band] Standard: 1	The number of the band to use as elevation
Compute edges	`COMPUTE_EDGES`	[Boolean] Standard: falsch	Generates edges from the elevation raster
Terrain Ruggedness Index	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Terrain Ruggedness Index	`OUTPUT`	[Raster]	Output ruggedness raster. The value -9999 is used as NoData value.

Pythoncode

Algorithm ID: `gdal:triterrainruggednessindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Topographic Position Index (TPI)

Outputs a single-band raster with values computed from the elevation. TPI stands for Topographic Position Index, which is defined as the difference between a central pixel and the mean of its surrounding cells.

This algorithm is derived from the GDAL DEM utility.

Default menu: *Raster* ► *Analysis*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input elevation raster layer
Band number	`BAND`	[raster band] Standard: 1	The number of the band to use for elevation values
Compute edges	`COMPUTE_EDGES`	[Boolean] Standard: falsch	Generates edges from the elevation raster
Terrain Ruggedness Index	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Terrain Ruggedness Index	`OUTPUT`	[Raster]	Output raster.

Pythoncode

Algorithm ID: `gdal:tpitopographicpositionindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.2.2 Raster conversion

gdal2xyz

Converts raster data to XYZ ASCII file format.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Raster layer to convert
Band number	`BAND`	[raster band] Default: The first band of the input layer	If the raster is multiband, choose the band you want to convert
Input pixel value to treat as NoData Optional	`NODATA_INPUT`	[numerisch: Ganzzahl] Standard: Nicht gesetzt	Input pixel value to treat as „NoData“ **Achtung** Using this parameter requires QGIS installed with GDAL >= 3.7 (see *Help* ► *About* menu).
Destination NoData Optional	`NODATA_OUTPUT`	[numerisch: Ganzzahl] Standard: Nicht gesetzt	Assign specified „NoData“ value to output. **Achtung** Using this parameter requires QGIS installed with GDAL >= 3.7 (see *Help* ► *About* menu).

Fortsetzung auf der nächsten Seite

Tab. 24.897 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Do not output NoData values	`SKIP_NODATA`	[Boolean] Standard: falsch	Do not output „NoData“ values. **Achtung** Using this parameter requires QGIS installed with GDAL >= 3.3 (see *Help* ► *About* menu).
Output comma-separated values	`CSV`	[Boolean] Standard: falsch	Sets whether the output file should be of type comma-separated values (csv).
XYZ ASCII file	`OUTPUT`	[Datei] Standard: `[In temporäre Datei speichern]`	Specification of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
XYZ ASCII file	`INPUT`	[table]	Table file containing the values exported from the raster band.

Pythoncode

Algorithm ID: `gdal:gdal2xyz`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

PCT to RGB

Converts an 8 bit paletted image to a 24 bit RGB. It will convert a pseudocolor band from the input file to an RGB file of the desired format.

This algorithm is derived from the GDAL pct2rgb utility.

Default menu: *Raster* ► *Conversion*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input 8 bit raster image
Band number	`BAND`	[raster band] Default: The first band of the input layer	If the raster is multiband, choose the band you want to convert
Generate a RGBA file	`RGBA`	[Boolean] Standard: falsch	Sets whether the output file should be of type RGBA.
PCT to RGB	`OUTPUT`	[Datei] Standard: `[In temporäre Datei speichern]`	Specification of the output file. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
PCT to RGB	`OUTPUT`	[Raster]	24 bit RGB raster image

Pythoncode

Algorithm ID: `gdal:pcttorgb`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Polygonize (raster to vector)

Creates vector polygons for all connected regions of pixels in the raster sharing a common pixel value. Each polygon is created with an attribute indicating the pixel value of that polygon.

This algorithm is derived from the GDAL polygonize utility.

Default menu: *Raster* ► *Conversion*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer
Band number	`BAND`	[raster band] Default: The first band of the input layer	If the raster is multiband, choose the band you want to use
Name of the field to create	`FIELD`	[Text] Default: ‚DN‘	Specify the field name for the attributes of the connected regions.
Use 8-connectedness	`EIGHT_CONNECTE`	[Boolean] Standard: falsch	If not set, raster cells must have a common border to be considered connected (*4-connected*). If set, touching raster cells are also considered connected (*8-connected*).
Vectorized	`OUTPUT`	[Vektor: Polygon] Standard: `[In temporäre Datei speichern]`	Specification of the output (polygon) vector layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Vectorized	`OUTPUT`	[Vektor: Polygon]	Output vector layer

Pythoncode

Algorithm ID: `gdal:polygonize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Rearrange bands

Creates a new raster using selected band(s) from a given raster layer. The algorithm also makes it possible to reorder the bands for the newly-created raster.

This algorithm is derived from the GDAL translate utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer
Selected band(s)	`BANDS`	[raster band] [list] Standard: Nicht gesetzt	Ordered list of the bands to use to create the new raster
Konvertiert	`OUTPUT`	[Raster] Default: `[Save to temporary file]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).

Fortsetzung auf der nächsten Seite

Tab. 24.905 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output data type	`DATA_TYPE`	[Aufzählung] Standard: 0	Defines the data type of the output raster file. Options: • 0 — Datentyp des Eingabelayers verwenden • 1 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 2 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (qint16)) • 3 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 4 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 5— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 6 — Float32 (32-Bit-Gleitkommazahl (Float)) • 7 — Float64 (64-Bit-Gleitkommazahl (Double)) • 8 — CInt16 (komplexe Int16) • 9 — CInt32 (komplexe Int32) • 10 — CFloat32 (komplexe Float32) • 11 — CFloat64 (komplexe Float64) • 12 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Konvertiert	`OUTPUT`	[Raster]	Output raster layer with rearranged bands.

Pythoncode

Algorithm ID: `gdal:rearrange_bands`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

RGB to PCT

Converts a 24 bit RGB image into a 8 bit paletted. Computes an optimal pseudo-color table for the given RGB-image using a median cut algorithm on a downsampled RGB histogram. Then it converts the image into a pseudo-colored image using the color table. This conversion utilizes Floyd-Steinberg dithering (error diffusion) to maximize output image visual quality.

If you want to classify a raster map and want to reduce the number of classes it can be helpful to downsample your image with this algorithm before.

This algorithm is derived from the GDAL rgb2pct utility.

Default menu: *Raster* ► *Conversion*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input (RGB) raster layer
Number of colors	`NCOLORS`	[numerisch: Ganzzahl] Default: 2	The number of colors the resulting image will contain. A value from 2-256 is possible.
RGB to PCT	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
RGB to PCT	`OUTPUT`	[Raster]	Output raster layer.

Pythoncode

Algorithm ID: `gdal:rgbtopct`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Translate (convert format)

Converts raster data between different formats.

This algorithm is derived from the GDAL translate utility.

Default menu: *Raster* ► *Conversion*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer
Override the projection of the output file Optional	`TARGET_CRS`	[KBS]	Specify a projection for the output file
Assign a specified NoData value to output bands Optional	`NODATA`	[numerisch: Double] Standard: Nicht gesetzt	Defines the value to use for NoData in the output raster
Copy all subdatasets of this file to individual output files	`COPY_SUBDATASE`	[Boolean] Standard: falsch	Create individual files for subdatasets
Konvertiert	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output (translated) raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Fortsetzung auf der nächsten Seite

Tab. 24.910 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output data type	`DATA_TYPE`	[Aufzählung] Standard: 0	Defines the data type of the output raster file. Options: • 0 — Datentyp des Eingabelayers verwenden • 1 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 2 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (qint16)) • 3 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 4 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 5— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 6 — Float32 (32-Bit-Gleitkommazahl (Float)) • 7 — Float64 (64-Bit-Gleitkommazahl (Double)) • 8 — CInt16 (komplexe Int16) • 9 — CInt32 (komplexe Int32) • 10 — CFloat32 (komplexe Float32) • 11 — CFloat64 (komplexe Float64) • 12 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Konvertiert	`OUTPUT`	[Raster]	Output (translated) raster layer.

Pythoncode

Algorithm ID: `gdal:translate`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.2.3 Raster extraction

Clip raster by extent

Clips any GDAL-supported raster file to a given extent.

This algorithm is derived from the GDAL translate utility.

Default menu: *Raster* ► *Extraction*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	The input raster
Clipping extent	`EXTENT`	[Ausdehnung]	Extent that should be used for the output raster. Only pixels within the specified bounding box will be included in the output. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Override the projection for the output file	`OVERCRS`	[Boolean] Standard: falsch	If checked, the output file is assigned the input layer CRS.
Assign a specified NoData value to output bands Optional	`NODATA`	[numerisch: Double] Standard: Nicht gesetzt	Defines a value that should be inserted for the NoData values in the output raster
Clipped (extent)	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).
Output data type	`DATA_TYPE`	[Aufzählung] Standard: 0	Defines the format of the output raster file. Optionen: • 0 — Datentyp des Eingabelayers verwenden • 1 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 2 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (qint16)) • 3 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 4 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 5— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 6 — Float32 (32-Bit-Gleitkommazahl (Float)) • 7 — Float64 (64-Bit-Gleitkommazahl (Double)) • 8 — CInt16 (komplexe Int16) • 9 — CInt32 (komplexe Int32) • 10 — CFloat32 (komplexe Float32) • 11 — CFloat64 (komplexe Float64) • 12 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Clipped (extent)	`OUTPUT`	[Raster]	Output raster layer clipped by the given extent

Pythoncode

Algorithm ID: `gdal:cliprasterbyextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Clip raster by mask layer

Clips any GDAL-supported raster by a vector mask layer.

This algorithm is derived from the GDAL warp utility.

Default menu: *Raster* ► *Extraction*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	The input raster
Mask layer	`MASK`	[Vektor: Polygon]	Vector mask for clipping the raster
Source CRS	`SOURCE_CRS`	[KBS]	Set the coordinate reference to use for the input raster
Ziel KBS	`TARGET_CRS`	[KBS]	Set the coordinate reference to use for the mask layer

Fortsetzung auf der nächsten Seite

Tab. 24.915 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Target extent Optional	TARGET_EXTENT	[Ausdehnung]	Extent of the output file to be created Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Assign a specified NoData value to output bands Optional	NODATA	[numerisch: Double] Standard: Nicht gesetzt	Defines a value that should be inserted for the NoData values in the output raster
Create an output alpha band	ALPHA_BAND	[Boolean] Standard: falsch	Creates an alpha band for the result. The alpha band then includes the transparency values of the pixels.
Match the extent of the clipped raster to the extent of the mask layer	CROP_TO_CUTLIN	[Boolean] Standard: True	Applies the vector layer extent to the output raster if checked.
Keep resolution of input raster	KEEP_RESOLUTIO	[Boolean] Standard: falsch	The resolution of the output raster will not be changed
Set output file resolution	SET_RESOLUTION	[Boolean] Standard: falsch	Shall the output resolution (cell size) be specified
X Resolution to output bands Optional	X_RESOLUTION	[numerisch: Double] Standard: Nicht gesetzt	The width of the cells in the output raster
Y Resolution to output band Optional	Y_RESOLUTION	[numerisch: Double] Standard: Nicht gesetzt	The height of the cells in the output raster

Fortsetzung auf der nächsten Seite

Tab. 24.915 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Use multithreaded warping implementation	`MULTITHREADING`	[Boolean] Standard: falsch	Two threads will be used to process chunks of image and perform input/output operation simultaneously. Note that computation is not multithreaded itself.
Clipped (mask)	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Output data type	`DATA_TYPE`	[Aufzählung] Standard: 0	Defines the format of the output raster file. Optionen: • 0 — Datentyp des Eingabelayers verwenden • 1 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 2 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (qint16)) • 3 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 4 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 5— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 6 — Float32 (32-Bit-Gleitkommazahl (Float)) • 7 — Float64 (64-Bit-Gleitkommazahl (Double)) • 8 — CInt16 (komplexe Int16) • 9 — CInt32 (komplexe Int32) • 10 — CFloat32 (komplexe Float32) • 11 — CFloat64 (komplexe Float64) • 12 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Fortsetzung auf der nächsten Seite

Tab. 24.916 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Additional command-line parameters Optional	EXTRA	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Clipped (mask)	OUTPUT	[Raster]	Output raster layer clipped by the vector layer

Pythoncode

Algorithm ID: `gdal:cliprasterbymasklayer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Contour

Extracts contour lines from any GDAL-supported elevation raster.

This algorithm is derived from the GDAL contour utility.

Default menu: *Raster* ► *Extraction*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	INPUT	[Raster]	Input raster
Band number	BAND	[raster band] Standard: 1	Raster band to create the contours from
Interval between contour lines	INTERVAL	[numerisch: Double] Default: 10.0	Defines the interval between the contour lines in the given units of the elevation raster (minimum value 0)
Attribute name (if not set, no elevation attribute is attached) Optional	FIELD_NAME	[Text] Default: ‚ELEV‘	Provides a name for the attribute in which to put the elevation.

Fortsetzung auf der nächsten Seite

Tab. 24.918 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Offset from zero relative to which to interpret intervals Optional	`OFFSET`	[numerisch: Double] Default: 0.0	
Contours	`OUTPUT`	[vector: line] Standard: `[In temporäre Datei speichern]`	Specification of the output vector layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Produce 3D vector	`CREATE_3D`	[Boolean] Standard: falsch	Forces production of 3D vectors instead of 2D. Includes elevation at every vertex.
Treat all raster values as valid	`IGNORE_NODATA`	[Boolean] Standard: falsch	Ignores any NoData values in the dataset.
Input pixel value to treat as „NoData“ Optional	`NODATA`	[numerisch: Double] Standard: Nicht gesetzt	Defines a value that should be inserted for the NoData values in the output raster
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options. Refer to the corresponding GDAL utility documentation.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Contours	`OUTPUT`	[vector: line]	Output vector layer with contour lines

Pythoncode

Algorithm ID: `gdal:contour`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Contour Polygons

Extracts contour polygons from any GDAL-supported elevation raster.

This algorithm is derived from the GDAL contour utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster
Band number	`BAND`	[raster band] Standard: 1	Raster band to create the contours from
Interval between contour lines	`INTERVAL`	[numerisch: Double] Default: 10.0	Defines the interval between the contour lines in the given units of the elevation raster (minimum value 0)
Offset from zero relative to which to interpret intervals Optional	`OFFSET`	[numerisch: Double] Default: 0.0	
Attribute name for minimum elevation of contour polygon Optional	`FIELD_NAME_MIN`	[Text] Default: ‚ELEV_MIN‘	Provides a name for the attribute in which to put the minimum elevation of contour polygon. If not provided no minimum elevation attribute is attached.
Attribute name for maximum elevation of contour polygon Optional	`FIELD_NAME_MAX`	[Text] Default: ‚ELEV_MAX‘	Provides a name for the attribute in which to put the maximum elevation of contour polygon. If not provided no maximum elevation attribute is attached.
Contours	`OUTPUT`	[Vektor: Polygon] Standard: `[In temporäre Datei speichern]`	Specification of the output vector layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Produce 3D vector	`CREATE_3D`	[Boolean] Standard: falsch	Forces production of 3D vectors instead of 2D. Includes elevation at every vertex.
Treat all raster values as valid	`IGNORE_NODATA`	[Boolean] Standard: falsch	Ignores any NoData values in the dataset.
Input pixel value to treat as „NoData“ Optional	`NODATA`	[numerisch: Double] Standard: Nicht gesetzt	Defines a value that should be inserted for the NoData values in the output raster

Fortsetzung auf der nächsten Seite

Tab. 24.922 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Additional command-line parameters Optional	EXTRA	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options. Refer to the corresponding GDAL utility documentation.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Contours	OUTPUT	[Vektor: Polygon]	Output vector layer with contour polygons

Pythoncode

Algorithm ID: `gdal:contour_polygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.2.4 Raster miscellaneous

Build overviews (pyramids)

To speed up rendering time of raster layers overviews (pyramids) can be created. Overviews are lower resolution copies of the data which QGIS uses depending of the level of zoom.

This algorithm is derived from the GDAL addo utility.

Default menu: *Raster* ► *Miscellaneous*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	INPUT	[Raster]	Input raster layer
Remove all existing overviews	CLEAN	[Boolean] Standard: falsch	Removes existing overviews from the raster. By default these are not removed.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Overview levels	`LEVELS`	[Text] Default: ‚2 4 8 16‘	Defines the number of overview levels calculated by the original resolution of the input raster layer. By default 4 levels will be taken into consideration.
Resampling method Optional	`RESAMPLING`	[Aufzählung] Standard: 0	Calculates the overviews with a defined resampling method. Possible resampling methods are: • 0 – Nearest Neighbour (`nearest`) • 1 – Average (`average`) • 2 – Gaussian (`gauss`) • 3 – Cubic (4x4 kernel) (`cubic`) • 4 – Cubic B-Spline (4x4 kernel) (`cubicspline`) • 5 – Lanczos (6x6 kernel) (`lanczos`) • 6 – Average MP (`average_mp`) • 7 – Average in Mag/Phase Space (`average_magphase`) • 8 – Mode (`mode`)
Overviews format Optional	`FORMAT`	[Aufzählung] Standard: 0	The overviews can be stored internally, or externally as GTiff or ERDAS Imagine file. By default the overviews are stored in the output raster. Possible formats methods are: • 0 – Internal (if possible) • 1 – External (GTiff .ovr) • 2 – External (ERDAS Imagine .aux)
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Pyramidized	`OUTPUT`	[Raster]	Output raster layer with overviews

Pythoncode

Algorithm ID: `gdal:overviews`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Build virtual raster

Builds a VRT (Virtual Dataset) that is a mosaic of the list of input GDAL-supported rasters. With a mosaic you can merge several raster files.

This algorithm is derived from the GDAL buildvrt utility.

Default menu: *Raster* ► *Miscellaneous*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[raster] [list]	GDAL-supported raster layers.
Resolution	`RESOLUTION`	[Aufzählung] Standard: 0	The output resolution of the mosaic. By default the average resolution of the raster files will be chosen. Optionen: • 0 — Average (`average`) • 1 — Highest (`highest`) • 2 — Lowest (`lowest`)
Place each input file into a separate band	`SEPARATE`	[Boolean] Standard: falsch	With ‚True' you can define that each raster file goes into a separated stacked band in the VRT band.
Allow projection difference	`PROJ_DIFFERENC`	[Boolean] Standard: falsch	Allows that the output bands have different projections derived from the projection of the input raster layers.
Virtual	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Add alpha mask band to VRT when source raster has none	`ADD_ALPHA`	[Boolean] Standard: falsch	Adds an alpha mask band to the VRT when the source raster has none.
Override projection for the output file Optional	`ASSIGN_CRS`	[KBS] Standard: Nicht gesetzt	Overrides the projection for the output file. No reprojection is done.

Fortsetzung auf der nächsten Seite

Tab. 24.928 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Resampling algorithm	`RESAMPLING`	[Aufzählung] Standard: 0	The resampling algorithm to use. Options: • 0 — Nearest Neighbour (`nearest`) • 1 — Bilinear (2x2 kernel) (`bilinear`) • 2 — Cubic (4x4 kernel) (`cubic`) • 3 — Cubic B-Spline (4x4 kernel) (`cubicspline`) • 4 — Lanczos (6x6 kernel) (`lanczos`) • 5 — Average (`average`) • 6 — Mode (`mode`)
NoData value(s) for input bands (space separated) Optional	`SRC_NODATA`	[Text] Standard: Nicht gesetzt	Space separated NoData value(s) for input band(s)
Additional command-line parameters	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Virtual	`OUTPUT`	[Raster]	Output raster layer

Pythoncode

Algorithm ID: `gdal:buildvirtualraster`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

gdal2tiles

Generates a directory with small tiles and metadata, following the OSGeo Tile Map Service Specification. See also the OpenGIS Web Map Tile Service Implementation Standard. Simple web pages with viewers based on Google Maps, OpenLayers and Leaflet are generated as well. To explore your maps on-line in the web browser, you only need to upload the generated directory onto a web server.

This algorithm also creates the necessary metadata for Google Earth (KML SuperOverlay), in case the supplied map uses `EPSG:4326` projection.

ESRI world files and embedded georeferencing is used during tile generation, but you can publish a picture without proper georeferencing too.

This algorithm is derived from the GDAL gdal2tiles utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	GDAL-supported raster layer.
Tile cutting profile	`PROFILE`	[Aufzählung] Standard: 0	One of: • 0 — Mercator (`mercator`) • 1 — Geodetic (`geodetic`) • 2 — Raster (`raster`)
Zoom levels to render Optional	`ZOOM`	[Text] Default: ‚‘	
Web viewer to generate	`VIEWER`	[Aufzählung] Standard: 0	One of: • 0 — All (`all`) • 1 — GoogleMaps (`google`) • 2 — OpenLayers (`openlayers`) • 3 — Leaflet (`leaflet`) • 4 — None (`none`)
Title of the map Optional	`TITLE`	[Text] Default: ‚‘	
Copyright of the map	`COPYRIGHT`	[Text] Default: ‚‘	
Ausgabeverzeichnis	`OUTPUT`	[Verzeichnis] Standard: `[In temporärem Verzeichnis speichern]`	Specify the output folder for the tiles. *One of*: • In einem temporären Verzeichnis speichern • In Verzeichnis speichern

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Resampling method	`RESAMPLING`	[Aufzählung] Standard: 0	The resampling algorithm to use. Options: • 0 — Average (`average`) • 1 — Nearest Neighbour (`near`) • 2 — Bilinear (2x2 kernel) (`bilinear`) • 3 — Cubic (4x4 kernel) (`cubic`) • 4 — Cubic B-Spline (4x4 kernel) (`cubicspline`) • 5 — Lanczos (6x6 kernel) (`lanczos`) • 6 — Antialias (`antialias`)
The spatial reference system used for the source input data Optional	`SOURCE_CRS`	[KBS] Standard: Nicht gesetzt	

Fortsetzung auf der nächsten Seite

Tab. 24.931 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Transparency value to assign to the input data Optional	NODATA	[numerisch: Double] Default: 0.0	
URL address where the generated tiles are going to be published Optional	URL	[Text] Default: ‚‘	
Google Maps API key (http://code.google.c Optional	GOOGLE_KEY	[Text] Default: ‚‘	Your Google maps API key.
Bing Maps API key (https://www.bingm Optional	BING_KEY	[Text] Default: ‚‘	Your Bing maps API key.
Generate only missing files	RESUME	[Boolean] Standard: falsch	
Generate KML for Google Earth	KML	[Boolean] Standard: falsch	
Avoid automatic generation of KML files for EPSG:4326	NO_KML	[Boolean] Standard: falsch	

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabeverzeichnis	OUTPUT	[Verzeichnis]	The output folder (for the tiles)

Pythoncode

Algorithm ID: `gdal:gdal2tiles`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Vereinigen

Merges raster files in a simple way. Here you can use a pseudocolor table from an input raster and define the output raster type. All the images must be in the same coordinate system.

This algorithm is derived from the GDAL merge utility.

Default menu: *Raster* ► *Miscellaneous*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[raster] [list]	Input raster layers
Grab pseudocolor table from first layer	`PCT`	[Boolean] Standard: falsch	The pseudocolor table from the first layer will be used for the coloring
Place each input file into a separate band	`SEPARATE`	[Boolean] Standard: falsch	Place each input file into a separate band
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the format of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).
Zusammengeführt	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Input pixel value to treat as „NoData“ Optional	NODATA_INPUT	[numerisch: Double] Standard: Nicht gesetzt	Ignores pixels from files being merged in with this pixel value
Assign specified „NoData“ value to output Optional	NODATA_OUTPUT	[numerisch: Double] Standard: Nicht gesetzt	Assigns the specified NoData value to output bands.
Additional creation options Optional	CREATION_OPTIO (for QGIS <= 3.42, this was OPTIONS)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).
Additional command-line parameters	EXTRA	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Zusammengeführt	OUTPUT	[Raster]	Output raster layer

Pythoncode

Algorithm ID: `gdal:merge`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Pansharpening

Performs a pan-sharpening operation. It can create a „classic“ output dataset (such as GeoTIFF), or a VRT dataset describing the pan-sharpening operation.

See GDAL Pansharpen.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Spectral dataset	`SPECTRAL`	[Raster]	Input (spectral) raster layer
Panchromatic dataset	`PANCHROMATIC`	[Raster]	Input (panchromatic) raster layer
Ergebnis	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output (sharpened) raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Resampling algorithm	`RESAMPLING`	[Aufzählung] Default: 2	The resampling algorithm to use. Options: • 0 — Nearest Neighbour (`nearest`) • 1 — Bilinear (2x2 kernel) (`bilinear`) • 2 — Cubic (4x4 kernel) (`cubic`) • 3 — Cubic B-Spline (4x4 kernel) (`cubicspline`) • 4 — Lanczos (6x6 kernel) (`lanczos`) • 5 — Average (`average`)
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ergebnis	`OUTPUT`	[Raster]	Output (sharpened) raster layer

Pythoncode

Algorithm ID: `gdal:pansharp`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster calculator

Command line raster calculator with numpy syntax. Use any basic arithmetic supported by numpy arrays, such as +, -, *, and / along with logical operators, such as >. Note that all input rasters must have the same dimensions, but no projection checking is performed.

See the GDAL Raster Calculator utility docs.

Siehe auch

Raster calculator, *Raster calculator (virtual)*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input layer A	`INPUT_A`	[Raster]	First input raster layer (mandatory)
Number of raster band for A	`BAND_A`	[raster band]	Band for input layer A (mandatory)
Input layer B Optional	`INPUT_B`	[Raster] Standard: Nicht gesetzt	Second input raster layer
Number of raster band for B Optional	`BAND_B`	[raster band]	Band for input layer B
Input layer C Optional	`INPUT_C`	[Raster] Standard: Nicht gesetzt	Third input raster layer
Number of raster band for C Optional	`BAND_C`	[raster band]	Band for input layer C
Input layer D Optional	`INPUT_D`	[Raster] Standard: Nicht gesetzt	Fourth input raster layer
Number of raster band for D Optional	`BAND_D`	[raster band]	Band for input layer D
Input layer E Optional	`INPUT_E`	[Raster] Standard: Nicht gesetzt	Fifth input raster layer

Fortsetzung auf der nächsten Seite

Tab. 24.939 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Number of raster band for E Optional	`BAND_E`	[raster band]	Band for input layer E
Input layer F Optional	`INPUT_F`	[Raster]	Sixth input raster layer
Number of raster band for F Optional	`BAND_F`	[raster band] Standard: Nicht gesetzt	Band for input layer F
Calculation in gdalnumeric syntax using +-/* or any numpy array functions (i.e. logical_and())	`FORMULA`	[Text] Default: ‚‘	The calculation formula. Examples: • `A*(A>0)` — outputs the value of the raster A if the value of A is greater than 0. If not, outputs 0. • `A*(A>0 and A>B)` — outputs the value of A if that value is bigger than 0 and bigger than the value of B. If not, outputs 0. • `A*logical_or(A<=177, A>=185)` — outputs the value of A if A <= 177 or A >= 185. If not, outputs 0. • `sqrt(A*A+B*B)` — Outputs the square root of the sum of the value of A squared and the value of B squared.
Set output NoData value Optional	`NO_DATA`	[numerisch: Double] Standard: Nicht gesetzt	Value to use for NoData
Handling of extent differences	`EXTENT_OPT`	[Aufzählung] Standard: 0	Determines how to handle rasters with different extents. Supported options are: • 0 — Ignore • 1 — Fail • 2 — Union • 3 — Intersect **Achtung** Using this parameter requires QGIS installed with GDAL >= 3.3 (see *Help* ► *About* menu).

Fortsetzung auf der nächsten Seite

Tab. 24.939 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output extent Optional	`INPUT`	[Ausdehnung]	Custom extent of the output raster. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein. **Achtung** Using this parameter requires QGIS installed with GDAL >= 3.3 (see *Help* ► *About* menu).
Output raster type	`RTYPE`	[Aufzählung] Default: 5	Defines the data type of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Fortsetzung auf der nächsten Seite

Tab. 24.939 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Calculated	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specify the output (calculated) raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Default: ‚‘	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Calculated	`OUTPUT`	[Raster]	Output (calculated) raster layer

Pythoncode

Algorithm ID: `gdal:rastercalculator`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Raster information

The gdalinfo program lists various information about a GDAL supported raster dataset.

This algorithm is derived from the GDAL info utility.

Default menu: *Raster* ► *Miscellaneous*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer
Force computation of the actual min/max values for each band	`MIN_MAX`	[Boolean] Standard: falsch	Forces computation of the actual min/max values for each band in the dataset
Read and display image statistics (force computation if necessary)	`STATS`	[Boolean] Standard: falsch	Reads and displays image statistics. Forces computation if no statistics are stored in an image.
Suppress GCP info	`NO_GCP`	[Boolean] Standard: falsch	Suppresses ground control points list printing. It may be useful for datasets with huge amount of GCPs, such as L1B AVHRR or HDF4 MODIS which contain thousands of them.
Suppress metadata info	`NO_METADATA`	[Boolean] Standard: falsch	Suppresses metadata printing. Some datasets may contain a lot of metadata strings.
Layer Information	`OUTPUT`	[html] Standard: `[In temporäre Datei speichern]`	Specify the HTML file for output. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Layer Information	`OUTPUT`	[html]	The HTML file containing information about the input raster layer

Pythoncode

Algorithm ID: `gdal:gdalinfo`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Retile

Retiles a set of input tiles. All the input tiles must be georeferenced in the same coordinate system and have a matching number of bands. Optionally pyramid levels are generated.

This algorithm is derived from the GDAL Retile utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input files	`INPUT`	[raster] [list]	The input raster files
Tile width	`TILE_SIZE_X`	[numerisch: Ganzzahl] Default: 256	Width of the tiles in pixels (minimum 0)
Tile height	`TILE_SIZE_Y`	[numerisch: Ganzzahl] Default: 256	Height of the tiles in pixels (minimum 0)
Overlap in pixels between consecutive tiles	`OVERLAP`	[numerisch: Ganzzahl] Standard: 0	
Number of pyramid levels to build	`LEVELS`	[numerisch: Ganzzahl] Standard: 1	Minimum: 0
Ausgabeverzeichnis	`OUTPUT`	[Verzeichnis] Standard: `[In temporärem Verzeichnis speichern]`	Specify the output folder for the tiles. *One of*: • In einem temporären Verzeichnis speichern • In Verzeichnis speichern

Fortsetzung auf der nächsten Seite

Tab. 24.945 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
CSV file containing the tile(s) georeferencing information	`OUTPUT_CSV`	[Datei] Standard: `[Ausgabe überspringen]`	Specify the output file for the tiles. *One of*: • Ausgabe übersrpingen • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Source coordinate reference system Optional	`SOURCE_CRS`	[KBS] Standard: Nicht gesetzt	
Resampling method	`RESAMPLING`	[Aufzählung] Standard: 0	The resampling algorithm to use. Options: • 0 — Nearest Neighbour (`nearest`) • 1 — Bilinear (2x2 kernel) (`bilinear`) • 2 — Cubic (4x4 kernel) (`cubic`) • 3 — Cubic B-Spline (4x4 kernel) (`cubicspline`) • 4 — Lanczos (6x6 kernel) (`lanczos`)
Column delimiter used in the CSV file Optional	`DELIMITER`	[Text] Default: „;“	Delimiter to use in the CSV file containing the tile(s) georeferencing information
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: „“	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters Optional	`EXTRA`	[Text] Default: „“	Add extra GDAL command line options

Fortsetzung auf der nächsten Seite

Tab. 24.946 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the format of the output raster file. Options: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).
Build only the pyramids	`ONLY_PYRAMIDS`	[Boolean] Standard: falsch	
Use separate directory for each tile row	`DIR_FOR_ROW`	[Boolean] Standard: falsch	

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ausgabeverzeichnis	`OUTPUT`	[Verzeichnis]	The output folder for the tiles.
CSV file containing the tile(s) georeferencing information	`OUTPUT_CSV`	[Datei]	The CSV file with georeferencing information for the tiles.

Pythoncode

Algorithm ID: `gdal:retile`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Tile index

Builds a vector layer with a record for each input raster file, an attribute containing the filename, and a polygon geometry outlining the raster. This output is suitable for use with MapServer as a raster tileindex.

This algorithm is derived from the GDAL Tile Index utility.

Default menu: *Raster* ► *Miscellaneous*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Input files	`LAYERS`	[raster] [list]	The input raster files. Can be multiple files.
Field name to hold the file path to the indexed rasters	`PATH_FIELD_NAM` Optional	[Text] Default: ‚location‘	The output field name to hold the file path/location to the indexed rasters.
Store absolute path to the indexed rasters	`ABSOLUTE_PATH`	[Boolean] Standard: falsch	Set whether the absolute path to the raster files is stored in the tile index file. By default the raster filenames will be put in the file exactly as they are specified in the command.
Skip files with different projection reference	`PROJ_DIFFERENC`	[Boolean] Standard: falsch	Only files with same projection as files already inserted in the tile index will be inserted. Default does not check projection and accepts all inputs.
Tile index	`OUTPUT`	[Vektor: Polygon] Standard: `[In temporäre Datei speichern]`	Specify the polygon vector layer to write the index to. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Transform geometries to the given CRS Optional	`TARGET_CRS`	[KBS]	Geometries of input files will be transformed to the specified target coordinate reference system. Default creates simple rectangular polygons in the same coordinate reference system as the input rasters.
The name of the field to store the SRS of each tile Optional	`CRS_FIELD_NAME`	[Text]	The name of the field to store the SRS of each tile
The format in which the CRS of each tile must be written	`CRS_FORMAT`	[Aufzählung] Standard: 0	Format for the CRS. One of: • 0 – Auto (`AUTO`) • 1 – Well-known text (`WKT`) • 2 – EPSG (`EPSG`) • 3 – Proj.4 (`PROJ`)

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Tile index	`OUTPUT`	[Vektor: Polygon]	The polygon vector layer with the tile index.

Pythoncode

Algorithm ID: `gdal:tileindex`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Viewshed

Calculates a viewshed raster from an input raster DEM using method defined in Wang2000 for a user defined point.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input elevation raster layer
Band number	`BAND`	[raster band] Standard: 1	The number of the band to use as elevation
Observer location	`OBSERVER`	[coordinates]	The location of the observer
Observer height	`OBSERVER_HEIGH`	[numerisch: Double] Standard: 1,0	The altitude of the observer, in the DEM units
Target height	`TARGET_HEIGHT`	[numerisch: Double] Standard: 1,0	The altitude of the target element, in the DEM units
Maximum distance from observer to compute visibility	`MAX_DISTANCE`	[numerisch: Double] Default: 100.0	Maximum distance from observer to compute visibility, in the DEM units
Ergebnis	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (`|`).
Additional command-line parameters	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Ergebnis	`OUTPUT`	[Raster]	The raster layer displaying the viewshed.

Pythoncode

Algorithm ID: `gdal:viewshed`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.2.5 Rasterprojektionen

Projektion zuweisen

Wendet ein Koordinatenbezugssystem auf ein Rasterdatensatz an.

This algorithm is derived from the GDAL edit utility.

Default menu: *Raster* ► *Projections*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT_LAYER`	[Raster]	Input raster layer
Gewünschte KBS	`CRS`	[KBS]	The projection (CRS) of the output layer

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Layer with projection	`OUTPUT`	[Raster]	The output raster layer (with the new projection information)

Python-Code

Algorithm ID: `gdal:assignprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Projektion extrahieren

Extracts the projection of a raster file and writes it into a *world* file with extension `.wld`.

This algorithm is derived from the GDAL srsinfo utility.

Default menu: *Raster* ► *Projections*

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input file	`INPUT_LAYER`	[Raster]	Input raster The raster layer has to be file based, as the algorithm uses the path to the raster file as the location of the generated `.wld` file. Using a non-file raster layer will lead to an error.
Create also .prj file	`PRJ_FILE_CREAT`	[Boolean] Standard: falsch	If this is activated a `.prj` file containing the projection information is also created.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
World file	`WORLD_FILE`	[Datei]	Text file with extension `.wld` containing transformation parameters for the raster file.
ESRI Shapefile prj file	`PRJ_FILE`	[Datei]	Text file with `.prj` extension that describes the CRS. Will be `None` if *Create also .prj file* is False.

Python-Code

Algorithm ID: `gdal:extractprojection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Warp (reproject)

Reprojects a raster layer into another Coordinate Reference System (CRS). The output file resolution and the resampling method can be chosen.

This algorithm is derived from the GDAL warp utility.

Default menu: *Raster* ► *Projections*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Raster]	Input raster layer to reproject
Source CRS Optional	`SOURCE_CRS`	[KBS]	Defines the CRS of the input raster layer
Ziel KBS Optional	`TARGET_CRS`	[KBS] Default: `EPSG:4326`	The CRS of the output layer
Resampling method to use	`RESAMPLING`	[Aufzählung] Standard: 0	Pixel value resampling method to use. Options: • 0 — Nächster Nachbar • 1 — Bilinear (2x2 Kern) • 2 — Kubisch (4x4 Kern) • 3 — Kubisch B-Spline (4x4 Kern) • 4 — Lanczos (6x6 Kern) • 5 — Durchschnitt • 6 — Modus • 7 — Maximum • 8 — Minimum • 9 — Median • 10 — erstes Quartil (Q1) • 11 — drittes Quartil (Q3)
NoData value for output bands Optional	`NODATA`	[numerisch: Double] Standard: Nicht gesetzt	Sets NoData value for output bands. If not provided, then NoData values will be copied from the source dataset.

Fortsetzung auf der nächsten Seite

Tab. 24.958 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output file resolution in target georeferenced units Optional	`TARGET_RESOLUT`	[numerisch: Double] Standard: Nicht gesetzt	Defines the output file resolution of reprojection result
Reprojiziert	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).
Output data type	`DATA_TYPE`	[Aufzählung] Standard: 0	Defines the format of the output raster file. Options: • 0 — Datentyp des Eingabelayers verwenden • 1 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 2 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (qint16)) • 3 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 4 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 5— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 6 — Float32 (32-Bit-Gleitkommazahl (Float)) • 7 — Float64 (64-Bit-Gleitkommazahl (Double)) • 8 — CInt16 (komplexe Int16) • 9 — CInt32 (komplexe Int32) • 10 — CFloat32 (komplexe Float32) • 11 — CFloat64 (komplexe Float64) • 12 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).

Fortsetzung auf der nächsten Seite

Tab. 24.959 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Georeferenced extents of output file to be created Optional	`TARGET_EXTENT`	[Ausdehnung]	Sets the georeferenced extent of the output file to be created (in the *Target CRS* by default. In the *CRS of the target raster extent*, if specified). Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Aktuelles Kartenansichtsausmaß verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
CRS of the target raster extent Optional	`TARGET_EXTENT_`	[KBS]	Specifies the CRS in which to interpret the coordinates given for the extent of the output file. This must not be confused with the target CRS of the output dataset. It is instead a convenience e.g. when knowing the output coordinates in a geodetic long/lat CRS, but wanting a result in a projected coordinate system.
Use multithreaded warping implementation	`MULTITHREADING`	[Boolean] Standard: falsch	Two threads will be used to process chunks of the image and perform input/output operations simultaneously. Note that the computation itself is not multithreaded.
Additional command-line parameters Optional	`EXTRA`	[Text] Standard: Nicht gesetzt	Add extra GDAL command line options.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Reprojiziert	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Reprojected output raster layer

Python-Code

Algorithm ID: `gdal:warpreproject`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.2.6 Vector conversion

Format konvertieren

Converts any OGR-supported vector layer into another OGR-supported format.

This algorithm is derived from the ogr2ogr utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	Eingabe-Vektorlayer
Convert all layers from dataset	`CONVERT_ALL_LA`	[Boolean] Standard: falsch	Converts the whole dataset. Supported output formats for this option are `GPKG` and `GML`.
Konvertiert	`OUTPUT`	[wie Eingabe]	Specification of the output vector layer. *One of*: • In temporärer Datei speichern • In Datei speichern… For `Save to File`, the output format has to be specified. All GDAL vector formats are supported. For `Save to a Temporary File` the QGIS default vector format will be used.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘ (no additional options)	Additional GDAL creation options.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Konvertiert	`OUTPUT`	[wie Eingabe]	The output vector layer

Pythoncode

Algorithm ID: `gdal:convertformat`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Rasterize (overwrite with attribute)

Overwrites a raster layer with values from a vector layer. New values are assigned based on the attribute value of the overlapping vector feature.

This algorithm is derived from the GDAL rasterize utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Input raster layer	`INPUT_RASTER`	[Raster]	Input raster layer
Field to use for a burn-in value Optional	`FIELD`	[tablefield: numeric]	Defines the attribute field to use to set the pixels values

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Add burn in values to existing raster values	`ADD`	[Boolean] Standard: falsch	If False, pixels are assigned the selected field's value. If True, the selected field's value is added to the value of the input raster layer.
Additional command-line parameters Optional	`EXTRA`	[Text] Default: ,‘	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Rasterized	`OUTPUT`	[Raster]	The overwritten input raster layer

Pythoncode

Algorithm ID: `gdal:rasterize_over`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Rasterize (overwrite with fixed value)

Overwrites parts of a raster layer with a fixed value. The pixels to overwrite are chosen based on the supplied (overlapping) vector layer.

This algorithm is derived from the GDAL rasterize utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Input raster layer	`INPUT_RASTER`	[Raster]	Input raster layer
A fixed value to burn	`BURN`	[numerisch: Double] Default: 0.0	The value to burn

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Add burn in values to existing raster values	`ADD`	[Boolean] Standard: falsch	If False, pixels are assigned the fixed value. If True, the fixed value is added to the value of the input raster layer.
Additional command-line parameters Optional	`EXTRA`	[Text] Default: ‚‘	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Rasterized	`OUTPUT`	[Raster]	The overwritten input raster layer

Pythoncode

Algorithm ID: `gdal:rasterize_over_fixed_value`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Rasterize (vector to raster)

Converts vector geometries (points, lines and polygons) into a raster image.

This algorithm is derived from the GDAL rasterize utility.

Default menu: *Raster* ► *Conversion*

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Field to use for a burn-in value Optional	`FIELD`	[tablefield: numeric]	Defines the attribute field from which the attributes for the pixels should be chosen
A fixed value to burn Optional	`BURN`	[numerisch: Double] Default: 0.0	A fixed value to burn into a band for all features.
Burn value extracted from the „Z“ values of the feature	`USE_Z`	[Boolean] Standard: falsch	Indicates that a burn value should be extracted from the „Z“ values of the feature. Works with points and lines (linear interpolation along each segment). For polygons, works properly only if they are flat (same Z value for all vertices)
Output raster size units	`UNITS`	[Aufzählung] Standard: 0	Units to use when defining the output raster size/resolution. One of: • 0 — Pixels • 1 — Georeferenced units

Fortsetzung auf der nächsten Seite

Tab. 24.970 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Width/Horizontal resolution	`WIDTH`	[numerisch: Double] Default: 0.0	Sets the width (if size units is „Pixels“) or horizontal resolution (if size units is „Georeferenced units“) of the output raster. Minimum value: 0.0.
Height/Vertical resolution	`HEIGHT`	[numerisch: Double] Default: 0.0	Sets the height (if size units is „Pixels“) or vertical resolution (if size units is „Georeferenced units“) of the output raster.
Output extent Optional	`EXTENT`	[Ausdehnung]	Extent of the output raster layer. If the extent is not specified, the minimum extent that covers the selected reference layer(s) will be used. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Assign a specified NoData value to output bands Optional	`NODATA`	[numerisch: Double] Default: 0.0	Assigns a specified NoData value to output bands
Rasterized	`OUTPUT`	[Raster] Standard: `[In temporäre Datei speichern]`	Specification of the output raster layer. *One of*: • In temporärer Datei speichern • In Datei speichern… For `Save to File`, the output format has to be specified. All GDAL raster formats are supported. For `Save to a Temporary File` the QGIS default raster format will be used.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘	For adding one or more creation options that control the raster to be created (colors, block size, file compression…). For convenience, you can rely on predefined profiles (see *GDAL driver options section*). Batch Process and Model Designer: separate multiple options with a pipe character (|).
Output data type	`DATA_TYPE`	[Aufzählung] Default: 5	Defines the format of the output raster file. Optionen: • 0 — Byte (8-Bit-Ganzzahl ohne Vorzeichen (quint8)) • 1 — Int16 (16-Bit-Ganzzahl mit Vorzeichen (quint6)) • 2 — UInt16 (16-Bit-Ganzzahl ohne Vorzeichen (quint6)) • 3 — UInt32 (32-Bit-Ganzzahl ohne Vorzeichen (quint32)) • 34— Int132 (32-Bit-Ganzzahl mit Vorzeichen (qint32)) • 5 — Float32 (32-Bit-Gleitkomma (Float)) • 6 — Float64 (64-Bit-Gleitkommazahl (Double)) • 7 — CInt16 (komplexe Int16) • 8 — CInt32 (komplexe Int32) • 9 — CFloat32 (komplexe Float32) • 10 — CFloat64 (komplexe Float64) • 11 — Int8 (8-Bit-Ganzzahl mit Vorzeichen (qint8)) Die verfügbaren Optionen hängen von der mit QGIS erstellten GDAL-Version ab (siehe Menü *Hilfe* ► *Über*).
Pre-initialize the output image with value Optional	`INIT`	[numerisch: Double]	Pre-initializes the output image bands with this value. Not marked as the NoData value in the output file. The same value is used in all the bands.
Invert rasterization	`INVERT`	[Boolean] Standard: falsch	Burns the fixed burn value, or the burn value associated with the first feature into all parts of the image not inside the provided polygon.
Additional command-line parameters Optional	`EXTRA`	[Text] Default: ‚‘	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Rasterized	`OUTPUT`	[Raster]	Output raster layer

Pythoncode

Algorithm ID: `gdal:rasterize`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.2.7 Vector geoprocessing

Buffer vectors

Create buffers around the features of a vector layer.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	The input vector layer
Geometry column name	`GEOMETRY`	[Text] Default: ‚geometry'	The name of the input layer geometry column to use
Buffer distance	`DISTANCE`	[numerisch: Double] Default: 10.0	Minimum: 0.0
Dissolve by attribute Optional	`FIELD`	[tablefield: any] Standard: Nicht gesetzt	Field to use for dissolving
Ergebnisse auflösen	`DISSOLVE`	[Boolean] Standard: falsch	If set, the result is dissolved. If no field is set for dissolving, all the buffers are dissolved into one feature.
Produce one feature for each geometry in any kind of geometry collection in the source file	`EXPLODE_COLLEC`	[Boolean] Standard: falsch	

Fortsetzung auf der nächsten Seite

Tab. 24.973 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Buffer	`OUTPUT`	[Vektor: Polygon] Standard: `[In temporäre Datei speichern]`	Specify the output buffer layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘ (no additional options)	Additional GDAL creation options.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Buffer	`OUTPUT`	[Vektor: Polygon]	The output buffer layer

Pythoncode

Algorithm ID: `gdal:buffervectors`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Clip vector by extent

Clips any OGR-supported vector file to a given extent.

This algorithm is derived from the GDAL ogr2ogr utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	The input vector layer

Fortsetzung auf der nächsten Seite

Tab. 24.976 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Clip extent	`EXTENT`	[Ausdehnung]	Defines the bounding box that should be used for the output vector file. It has to be defined in target CRS coordinates. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Ausdehnung der Kartenansicht verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Clipped (extent)	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specify the output (clipped) layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘ (no additional options)	Additional GDAL creation options.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Clipped (extent)	`OUTPUT`	[wie Eingabe]	The output (clipped) layer. The default format is „ESRI Shapefile“.

Pythoncode

Algorithm ID: `gdal:clipvectorbyextent`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Clip vector by mask layer

Clips any OGR-supported vector layer by a mask polygon layer.

This algorithm is derived from the GDAL ogr2ogr utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	The input vector layer
Mask layer	`MASK`	[Vektor: Polygon]	Layer to be used as clipping extent for the input vector layer.
Clipped (mask)	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	The output (masked) layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘ (no additional options)	Additional GDAL creation options.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Clipped (mask)	`OUTPUT`	[wie Eingabe]	The output (masked) layer. The default format is „ESRI Shapefile“.

Pythoncode

Algorithm ID: `gdal:clipvectorbypolygon`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Dissolve

Dissolve (combine) geometries that have the same value for a given attribute / field. The output geometries are multipart.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	The input layer to dissolve
Dissolve field Optional	`FIELD`	[tablefield: any]	The field of the input layer to use for dissolving
Geometry column name	`GEOMETRY`	[Text] Default: ‚geometry‘	The name of the input layer geometry column to use for dissolving.
Dissolved	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specify the output layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Produce one feature for each geometry in any kind of geometry collection in the source file	`EXPLODE_COLLEC`	[Boolean] Standard: falsch	Produce one feature for each geometry in any kind of geometry collection in the source file
Keep input attributes	`KEEP_ATTRIBUTE`	[Boolean] Standard: falsch	Keep all attributes from the input layer
Count dissolved features	`COUNT_FEATURES`	[Boolean] Standard: falsch	Count the dissolved features and include it in the output layer.
Compute area and perimeter of dissolved features	`COMPUTE_AREA`	[Boolean] Standard: falsch	Compute the area and perimeter of dissolved features and include them in the output layer

Fortsetzung auf der nächsten Seite

Tab. 24.983 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Compute min/max/sum/mean for attribute	`COMPUTE_STATIS`	[Boolean] Standard: falsch	Calculate statistics (min, max, sum and mean) for the numeric attribute specified and include them in the output layer
Numeric attribute to calculate statistics on Optional	`STATISTICS_ATT`	[tablefield: numeric]	The numeric attribute to calculate statistics on
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘ (no additional options)	Additional GDAL creation options.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Dissolved	`OUTPUT`	[wie Eingabe]	The output multipart geometry layer (with dissolved geometries)

Pythoncode

Algorithm ID: `gdal:dissolve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Offset curve

Offsets lines by a specified distance. Positive distances will offset lines to the left, and negative distances will offset them to the right.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	The input line layer
Geometry column name	`GEOMETRY`	[Text] Default: ‚geometry‘	The name of the input layer geometry column to use
Offset distance (left-sided: positive, right-sided: negative)	`DISTANCE`	[numerisch: Double] Default: 10.0	

Fortsetzung auf der nächsten Seite

Tab. 24.985 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Offset curve	`OUTPUT`	[vector: line] Standard: `[In temporäre Datei speichern]`	Specify the output line layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘ (no additional options)	Additional GDAL creation options.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Offset curve	`OUTPUT`	[vector: line]	The output offset curve layer

Pythoncode

Algorithm ID: `gdal:offsetcurve`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

One side buffer

Creates a buffer on one side (right or left) of the lines in a line vector layer.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	The input line layer
Geometry column name	`GEOMETRY`	[Text] Default: ‚geometry‘	The name of the input layer geometry column to use

Fortsetzung auf der nächsten Seite

Tab. 24.988 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Buffer distance	`DISTANCE`	[numerisch: Double] Default: 10.0	
Buffer side	`BUFFER_SIDE`	[Aufzählung] Standard: 0	One of: • 0 — Right • 1 — Left
Dissolve by attribute Optional	`FIELD`	[tablefield: any] Standard: Nicht gesetzt	Field to use for dissolving
Dissolve all results	`DISSOLVE`	[Boolean] Standard: falsch	**If set, the result is dissolved.** If no field is set for dissolving, all the buffers are dissolved into one feature.
Produce one feature for each geometry in any kind of geometry collection in the source file	`EXPLODE_COLLEC`	[Boolean] Standard: falsch	
One-sided buffer	`OUTPUT`	[Vektor: Polygon] Standard: `[In temporäre Datei speichern]`	Specify the output buffer layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘ (no additional options)	Additional GDAL creation options.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
One-sided buffer	`OUTPUT`	[Vektor: Polygon]	The output buffer layer

Pythoncode

Algorithm ID: `gdal:onesidebuffer`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Points along lines

Generates a point on each line of a line vector layer at a distance from start. The distance is provided as a fraction of the line length.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[vector: line]	The input line layer
Geometry column name	`GEOMETRY`	[Text] Default: ‚geometry‘	The name of the input layer geometry column to use
Distance from line start represented as a fraction of line length	`DISTANCE`	[numerisch: Double] Default: 0.5	
Points along lines	`OUTPUT`	[Vektor: Punkt] Standard: `[In temporäre Datei speichern]`	Specify the output point layer. *One of*: • In temporärer Datei speichern • In Datei speichern…

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘ (no additional options)	Additional GDAL creation options.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Points along line	`OUTPUT`	[Vektor: Punkt]	The output point layer

Pythoncode

Algorithm ID: `gdal:pointsalonglines`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

24.2.8 Vector miscellaneous

Build virtual vector

Creates a virtual vector layer that contains a set of vector layers.

This algorithm is especially useful in case another algorithm needs multiple layers but accept only one `vrt` in which the layers are specified.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Input Datenquellen	`INPUT`	[vector: any] [list]	Select the vector layers you want to use to build the virtual vector
Create „unioned“ VRT	`UNIONED`	[Boolean] Standard: falsch	Check if you want to unite all the vectors in a single `vrt` file
Virtual vector	`OUTPUT`	[wie Eingabe] Standard: `[In temporäre Datei speichern]`	Specify the output layer containing only the duplicates. *One of*: • In temporärer Datei speichern • In Datei speichern…

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Virtual vector	`OUTPUT`	[Vektor: alle]	The output virtual vector made from the chosen sources

Python-Code

Algorithm ID: `gdal:buildvirtualvector`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Execute SQL

Runs a simple or complex query with SQL syntax on the source layer. The result of the query will be added as a new layer.

This algorithm is derived from the GDAL ogr2ogr utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	OGR-supported input vector layer
SQL expression	`SQL`	[Text]	Defines the SQL query, for example `SELECT * FROM my_table WHERE name is not null`.
SQL dialect	`DIALECT`	[Aufzählung] Standard: 0	SQL dialect to use. One of: • 0 — None • 1 — OGR SQL • 2 — SQLite
SQL result	`OUTPUT`	[Vektor: alle]	Specification of the output layer. *One of*: • In temporärer Datei speichern • In Datei speichern… For `Save to File`, the output format has to be specified. All GDAL vector formats are supported. For `Save to a Temporary File` the default output vector layer format will be used.

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚‘ (no additional options)	Additional GDAL creation options.

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
SQL result	`OUTPUT`	[Vektor: alle]	Vector layer created by the query

Python-Code

Algorithm ID: `gdal:executesql`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export to PostgreSQL (available connections)

Imports vector layers inside a PostgreSQL database on the basis of an available connection. The connection has to *be defined properly* beforehand. Be aware that the checkboxes ‚Save Username‘ and ‚Save Password‘ are activated. Then you can use the algorithm.

This algorithm is derived from the GDAL ogr2ogr utility.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Database (connection name)	`DATABASE`	[Text]	The PostgreSQL database to connect to
Eingabelayer	`INPUT`	[Vektor: alle]	OGR-supported vector layer to export to the database
Shape encoding Optional	`SHAPE_ENCODING`	[Text] Default: ‚‘	Sets the encoding to apply to the data

Fortsetzung auf der nächsten Seite

Tab. 24.999 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Output geometry type	GTYPE	[Aufzählung] Standard: 0	Defines the output geometry type. One of: • 0 — • 1 — NONE • 2 — GEOMETRY • 3 — POINT • 4 — LINESTRING • 5 — POLYGON • 6 — GEOMETRYCOLLECTION • 7 — MULTIPOINT • 8 — MULTIPOLYGON • 9 — MULTILINESTRING • 10 — CIRCULARSTRING • 11 — COUMPOUNDCURVE • 12 — CURVEPOLYGON • 13 — MULTICURVE • 14 — MULTISURFACE • 15 — CONVERT_TO_LINEAR • 16 — CONVERT_TO_CURVE
Assign an output CRS Optional	A_SRS	[KBS] Standard: Nicht gesetzt	Defines the output CRS of the database table
Reproject to this CRS on output Optional	T_SRS	[KBS] Standard: Nicht gesetzt	Reprojects/transforms to this CRS on output
Override source CRS Optional	S_SRS	[KBS] Standard: Nicht gesetzt	Overrides the input layer CRS
Schema (schema name) Optional	SCHEMA	[Text] Default: ‚public‘	Defines the schema for the database table
Table to export to (leave blank to use layer name) Optional	TABLE	[Text] Default: ‚‘	Defines a name for the table that will be imported into the database. By default the table name is the name of the input vector file.
Primary Key (new field) Optional	PK	[Text] Default: ‚id‘	Defines which attribute field will be the primary key of the database table
Primary Key (existing field, used if the above option is left empty) Optional	PRIMARY_KEY	[tablefield: any] Standard: Nicht gesetzt	Defines which attribute field in the exported layer will be the primary key of the database table
Geometry column name Optional	GEOCOLUMN	[Text] Default: ‚geom‘	Defines in which attribute field of the database there will be the geometry information
Vector dimensions Optional	DIM	[Aufzählung] Default: 0 (2D)	Defines the coordinate dimensions of the imported vector data. One of: • 0 — 2 (XY) • 1 — 3 (XYZ) • 2 — 4 (XYZM)

Fortsetzung auf der nächsten Seite

Tab. 24.999 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Distance tolerance for simplification Optional	`SIMPLIFY`	[Text] Default: ‚‘	Defines a distance tolerance for the simplification of the vector geometries to be imported. By default there is no simplification.
Maximum distance between 2 nodes (densification) Optional	`SEGMENTIZE`	[Text] Default: ‚‘	The maximum distance between two nodes. Used to create intermediate points. By default there is no densification.
Select features by extent (defined in input layer CRS) Optional	`SPAT`	[Ausdehnung] Standard: Nicht gesetzt	You can select features from a given extent that will be in the output table. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Aktuelles Kartenansichtsausmaß verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.
Clip the input layer using the above (rectangle) extent	`CLIP`	[Boolean] Standard: falsch	The input layer will be clipped by the extent you defined before
Select features using a SQL „WHERE“ statement (Ex: column=“value“) Optional	`WHERE`	[Text] Default: ‚‘	Defines with a SQL „WHERE“ statement which features should be selected from the input layer
Group N features per transaction (Default: 2000) Optional	`GT`	[Text] Default: ‚‘	You can group the input features in transactions where N defines the size. By default N limits the transaction size to 20000 features.
Overwrite existing table	`OVERWRITE`	[Boolean] Standard: True	If there is a table with the same name in the database, and if this option is set to True, the table will be overwritten.
Append to existing table	`APPEND`	[Boolean] Standard: falsch	If checked / True the vector data will be appended to an existing table. New fields found in the input layer are ignored. By default a new table will be created.

Fortsetzung auf der nächsten Seite

Tab. 24.999 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Append and add new fields to existing table	ADDFIELDS	[Boolean] Standard: falsch	If activated the vector data will be appended to an existing table, there won't be a new table created. New fields found in input layer are added to the table. By default a new table will be created.
Do not launder columns/table names	LAUNDER	[Boolean] Standard: falsch	With this option checked you can prevent the default behaviour (converting column names to lowercase, removing spaces and other invalid characters).
Do not create Spatial Index	INDEX	[Boolean] Standard: falsch	Prevents a spatial index for the output table from being created. By default, a spatial index is added.
Continue after a failure, skipping the failed feature	SKIPFAILURES	[Boolean] Standard: falsch	Continue after a failure, skipping the failed feature.
Validate geometries based on Simple Features specification	MAKEVALID	[Boolean] Standard: falsch	Applies cleanup operations on geometries to ensure they are valid regarding the rules of the Simple Features specification.
Promote to Multipart	PROMOTETOMULTI	[Boolean] Standard: True	Casts features geometry type to multipart in the output table
Keep width and precision of input attributes	PRECISION	[Boolean] Standard: True	Avoids modifying column attributes to comply with input data
Additional creation options Optional	CREATION_OPTIO (for QGIS <= 3.42, this was OPTIONS)	[Text] Default: ‚‘ (no additional options)	Additional GDAL creation options.

Ausgaben

This algorithm has no output.

Python-Code

Algorithm ID: `gdal:importvectorintopostgisdatabaseavailableconnections`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Export to PostgreSQL (new connection)

Imports vector layers inside a PostgreSQL database. A new connection to the PostgreSQL database must be created.

This algorithm is derived from the GDAL ogr2ogr utility.

Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: alle]	OGR-supported vector layer to export to the database
Shape encoding Optional	`SHAPE_ENCODING`	[Text] Default: ‚‘	Sets the encoding to apply to the data
Output geometry type	`GTYPE`	[Aufzählung] Standard: 0	Defines the output geometry type. One of: • 0 — • 1 — NONE • 2 — GEOMETRY • 3 — POINT • 4 — LINESTRING • 5 — POLYGON • 6 — GEOMETRYCOLLECTION • 7 — MULTIPOINT • 8 — MULTIPOLYGON • 9 — MULTILINESTRING • 10 — CIRCULARSTRING • 11 — COUMPOUNDCURVE • 12 — CURVEPOLYGON • 13 — MULTICURVE • 14 — MULTISURFACE • 15 — CONVERT_TO_LINEAR • 16 — CONVERT_TO_CURVE
Assign an output CRS Optional	`A_SRS`	[KBS] Standard: Nicht gesetzt	Defines the output CRS of the database table
Reproject to this CRS on output Optional	`T_SRS`	[KBS] Standard: Nicht gesetzt	Reprojects/transforms to this CRS on output
Override source CRS Optional	`S_SRS`	[KBS] Standard: Nicht gesetzt	Overrides the input layer CRS
Host Optional	`HOST`	[Text] Default: ‚localhost‘	Name of the database host
Port Optional	`PORT`	[Text] Default: ‚5432‘	Port number the PostgreSQL database server listens on
Username Optional	`USER`	[Text] Default: ‚‘	User name used to log in to the database
Database name Optional	`DBNAME`	[Text] Default: ‚‘	Name of the database
Password Optional	`PASSWORD`	[Text] Default: ‚‘	Password used with Username to connect to the database
Schema (schema name) Optional	`SCHEMA`	[Text] Default: ‚public‘	Defines the schema for the database table

Fortsetzung auf der nächsten Seite

Tab. 24.1000 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Table name, leave blank to use input name Optional	`TABLE`	[Text] Default: ‚'	Defines a name for the table that will be imported into the database. By default the table name is the name of the input vector file.
Primary Key (new field) Optional	`PK`	[Text] Default: ‚id'	Defines which attribute field will be the primary key of the database table
Primary Key (existing field, used if the above option is left empty) Optional	`PRIMARY_KEY`	[tablefield: any] Standard: Nicht gesetzt	Defines which attribute field in the exported layer will be the primary key of the database table
Geometry column name Optional	`GEOCOLUMN`	[Text] Default: ‚geom'	Defines in which attribute field to store the geometry information
Vector dimensions Optional	`DIM`	[Aufzählung] Default: 0 (2D)	Defines the coordinate dimensions of the imported vector data. One of: • 0 — 2 (`XY`) • 1 — 3 (`XYZ`) • 2 — 4 (`XYZM`)
Distance tolerance for simplification Optional	`SIMPLIFY`	[Text] Default: ‚'	Defines a distance tolerance for the simplification of the vector geometries to be imported. By default there is no simplification.
Maximum distance between 2 nodes (densification) Optional	`SEGMENTIZE`	[Text] Default: ‚'	The maximum distance between two nodes. Used to create intermediate points. By default there is no densification.
Select features by extent (defined in input layer CRS) Optional	`SPAT`	[Ausdehnung] Standard: Nicht gesetzt	You can select features from a given extent that will be in the output table. Verfügbare Methoden sind: • Aus Layer berechnen…: verwendet die Ausdehnung eines im aktuellen Projekt geladenen Layers • Berechne aus Layout-Karte…: verwendet die Ausdehnung eines *Layout-Kartenelements* im aktiven Projekt • Berechne aus Lesezeichen…: verwendet den Bereich eines gespeicherten *Lesezeichens* • Aktuelles Kartenansichtsausmaß verwenden • Auf Karte zeichnen: Klicken und ziehen Sie ein Rechteck, das den zu berücksichtigenden Bereich abgrenzt • Geben Sie die Koordinaten als `xmin, xmax, ymin, ymax` ein.

Fortsetzung auf der nächsten Seite

Tab. 24.1000 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Clip the input layer using the above (rectangle) extent	`CLIP`	[Boolean] Standard: falsch	The input layer will be clipped by the extent you defined before
Fields to include (leave empty to use all fields) Optional	`FIELDS`	[string] [list] Default: []	Defines fields to keep from the imported vector file. If none is selected, all the fields are imported.
Select features using a SQL „WHERE“ statement (Ex: column=“value“) Optional	`WHERE`	[Text] Default: ‚	Defines with a SQL „WHERE“ statement which features should be selected for the output table
Group N features per transaction (Default: 2000) Optional	`GT`	[Text] Default: ‚	You can group the input features in transactions where N defines the size. By default N limits the transaction size to 20000 features.
Overwrite existing table	`OVERWRITE`	[Boolean] Standard: True	If there is a table with the same name in the database, and if this option is set to True, the table will be overwritten.
Append to existing table	`APPEND`	[Boolean] Standard: falsch	If checked / True the vector data will be appended to an existing table. New fields found in the input layer are ignored. By default a new table will be created.
Append and add new fields to existing table	`ADDFIELDS`	[Boolean] Standard: falsch	If activated the vector data will be appended to an existing table, there won’t be created a new table. New fields found in input layer are added to the table. By default a new table will be created.
Do not launder columns/table names	`LAUNDER`	[Boolean] Standard: falsch	With this option checked you can prevent the default behaviour (converting column names to lowercase, removing spaces and other invalid characters).
Do not create Spatial Index	`INDEX`	[Boolean] Standard: falsch	Prevents a spatial index for the output table from being created. By default, a spatial index is added.
Continue after a failure, skipping the failed feature	`SKIPFAILURES`	[Boolean] Standard: falsch	Continue after a failure, skipping the failed feature.
Validate geometries based on Simple Features specification	`MAKEVALID`	[Boolean] Standard: falsch	Applies cleanup operations on geometries to ensure they are valid regarding the rules of the Simple Features specification.
Promote to Multipart	`PROMOTETOMULTI`	[Boolean] Standard: True	Casts features geometry type to multipart in the output table
Keep width and precision of input attributes	`PRECISION`	[Boolean] Standard: True	Avoids modifying column attributes to comply with input data
Additional creation options Optional	`CREATION_OPTIO` (for QGIS <= 3.42, this was `OPTIONS`)	[Text] Default: ‚ (no additional options)	Additional GDAL creation options.

Ausgaben

This algorithm has no output.

Python-Code

Algorithm ID: `gdal:importvectorintopostgisdatabasenewconnection`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Vector Information

Creates an information file that lists information about an OGR-supported data source. The output will be shown in a ‚Result' window and can be written into a HTML-file. The information includes the geometry type, feature count, the spatial extent, the projection information, the list of attributes and related type, list of relations and field domains, and many more.

This algorithm is derived from the GDAL ogrinfo utility.

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer
Enable listing of all layers in the dataset	`ALL_LAYERS`	[Boolean] Standard: falsch	If checked, QGIS will output the information of all the layers in the input dataset instead of information of the first layer only. Convenient when a multi layer format (GeoPackage, GML, KML, SpatiaLite, SQLite, …) is used as input.
Summary output only	`SUMMARY_ONLY`	[Boolean] Standard: True	Suppress listing of individual features and show only summary information like projection, schema, feature count and extents.
Suppress metadata info	`NO_METADATA`	[Boolean] Standard: falsch	Suppress metadata printing. Some datasets may contain a lot of metadata strings.
Layer Information	`OUTPUT`	[html] Standard: `[In temporäre Datei speichern]`	Specify the output HTML file that includes the file information. *One of*: • In temporärer Datei speichern • In Datei speichern… If no HTML-file is defined the output will be written to a temporary file

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional command line parameters Optional	`EXTRA`	[Text] Default: ‚' (no additional options)	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Layer Information	`OUTPUT`	[html]	The output HTML-file that includes the file information.

Python-Code

Algorithm ID: `gdal:ogrinfo`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

Vector Information (JSON)

Creates an information file that lists information about an OGR-supported data source. The output will be shown in a ‚Result' window and can be written into a `.JSON` file. The information includes the geometry type, feature count, the spatial extent, the projection information, the list of attributes and related type, list of relations and field domains, and many more.

This algorithm is derived from the GDAL ogrinfo utility.

Achtung

Running this algorithm requires QGIS installed with GDAL >= 3.7 (see *Help* ► *About* menu).

Parameter

Grundlegende Parameter

Bezeichnung	Name	Typ	Beschreibung
Eingabelayer	`INPUT`	[Vektor: Geometrie]	Eingabe-Vektorlayer

Fortsetzung auf der nächsten Seite

Tab. 24.1004 – Fortsetzung der vorherigen Seite

Bezeichnung	Name	Typ	Beschreibung
Enable listing of all layers in the dataset	`ALL_LAYERS`	[Boolean] Standard: falsch	If checked, QGIS will output the information of all the layers in the input dataset instead of information of the first layer only. Convenient when a multi layer format (GeoPackage, GML, KML, SpatiaLite, SQLite, …) is used as input.
Enable listing of features	`FEATURES`	[Boolean] Standard: falsch	
Suppress metadata info	`NO_METADATA`	[Boolean] Standard: falsch	Suppress metadata printing. Some datasets may contain a lot of metadata strings.
Layer Information	`OUTPUT`	[Datei] Standard: `[In temporäre Datei speichern]`	Specify the output JSON file that includes the file information. *One of*: • In temporärer Datei speichern • In Datei speichern… If no JSON file is defined the output will be written to a temporary file

Fortgeschrittene Parameter

Bezeichnung	Name	Typ	Beschreibung
Additional command line parameters Optional	`EXTRA`	[Text] Default: ‚‘ (no additional options)	Add extra GDAL command line options

Ausgaben

Bezeichnung	Name	Typ	Beschreibung
Layer Information	`OUTPUT`	[Datei]	The output *.JSON* file that includes the file information.

Python-Code

Algorithm ID: `gdal:ogrinfojson`

```
import processing
processing.run("algorithm_id", {parameter_dictionary})
```

Die *Algorithmus-Kennung* wird angezeigt, wenn Sie den Mauszeiger über den Algorithmus in der Verarbeitungs-Werkzeugkiste bewegen. Die Parameter-Liste (*parameter_dictionary*) enthält die Namen und Werte der Parameter. Siehe *Verarbeitungs-Algorithmen von der Konsole aus verwenden* für Details zur Ausführung von Verarbeitungsalgorithmen über die Python-Konsole.

KAPITEL 25

Erweiterungen

25.1 QGIS Plugins

QGIS wurde mit einer Plugin-Architektur entwickelt, dadurch können neue Funktionen einfach zur Anwendung hinzugefügt werden. Einige der Funktionen in QGIS sind tatsächlich als Plugins implementiert.

25.1.1 Kern-Erweiterungen und externe Erweiterungen

QGIS Plugins sind entweder als **Kern-Erweiterungen** (Core-Plugins) oder als **Externe Plugins** implementiert.

Kern-Erweiterungen werden vom QGIS Entwicklungsteam gepflegt und sind automatisch Teil jeder QGIS Installation. Sie sind in einer von zwei Sprachen geschrieben: **C++** oder **Python**.

Die meisten der externen Plugins sind derzeit in Python geschrieben. Sie sind entweder im „offiziellen“ QGIS Repository unter https://plugins.qgis.org/plugins/ oder in externen Repositories gespeichert und werden von den einzelnen Autoren gepflegt. Eine ausführliche Dokumentation über die Verwendung, die minimale QGIS-Version, die Homepage, die Autoren und andere wichtige Informationen werden für die Plugins im offiziellen Repository bereitgestellt. Für andere externe Repositories ist die Dokumentation möglicherweise mit den externen Plugins selbst verfügbar. Die Dokumentation der externen Plugins ist nicht in diesem Handbuch enthalten.

Um ein Plugin zu installieren oder zu aktivieren, gehen Sie in das Menü *Erweiterungen* und wählen Sie *Erweiterungen verwalten und installieren…*. Installierte externe Python-Plugins werden im Ordner `python/plugins` des aktiven *Benutzerprofil*-Pfades abgelegt.

Pfade zu benutzerdefinierten C ++ Plugins Bibliotheken können auch unter *Einstellungen* ► *Optionen* ► *System* hinzugefügt werden.

25.1.2 Der Erweiterungen Dialog

Der Einstellungen Reiter

Im unteren Teil des linken Bedienfelds befindet sich die Registerkarte *Settings*, auf der Sie einstellen können, welche Plugins in Ihrer Anwendung angezeigt werden können. Sie können die folgenden Optionen verwenden:

- *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.
- *Auch experimentelle Erweiterungen anzeigen*. QGIS zeigt Ihnen Plugins in frühen Entwicklungsstadien an, die im Allgemeinen nicht für den produktiven Einsatz geeignet sind. Für diese Plugins können Sie entweder die stabile oder die experimentelle Version installieren und jederzeit von einer zur anderen wechseln.
- *Auch veraltete Erweiterungen anzeigen*. Diese Plugins werden in der Regel nicht mehr gewartet, weil sie Ersatzfunktionen in QGIS haben, es an Maintainern mangelt, sie auf Funktionen angewiesen sind, die in QGIS nicht mehr verfügbar sind… Sie sind in der Regel für den produktiven Einsatz ungeeignet und erscheinen in der Plugin-Liste ausgegraut.

Standardmäßig stellt Ihnen QGIS im Abschnitt *Plugin Repositories* sein offizielles Plugin-Repository mit der URL `https://plugins.qgis.org/plugins/plugins.xml?qgis=<version>` zur Verfügung (wobei `<version>` für die genaue QGIS-Version steht, die Sie verwenden). Um externe Autoren-Repositories hinzuzufügen, klicken Sie auf *Hinzufügen…* und füllen Sie das Formular *Repositorien Details* mit einem Namen und der URL aus. Die URL kann vom Protokolltyp `http://` oder `file://` sein.

Das standardmäßige QGIS-Repository ist ein offenes Repository und Sie benötigen keine Authentifizierung, um darauf zuzugreifen. Sie können jedoch Ihr eigenes Plugin-Repository bereitstellen und eine Authentifizierung verlangen (Basisauthentifizierung, PKI). Weitere Informationen zur QGIS-Authentifizierungsunterstützung finden Sie im Kapitel *Authentifizierung*.

Wenn Sie eines oder mehrere der hinzugefügten Repositories nicht wünschen, können Sie diese auf der Registerkarte Einstellungen über die Schaltfläche *Bearbeiten…* deaktivieren oder über die Schaltfläche *Löschen* vollständig entfernen.

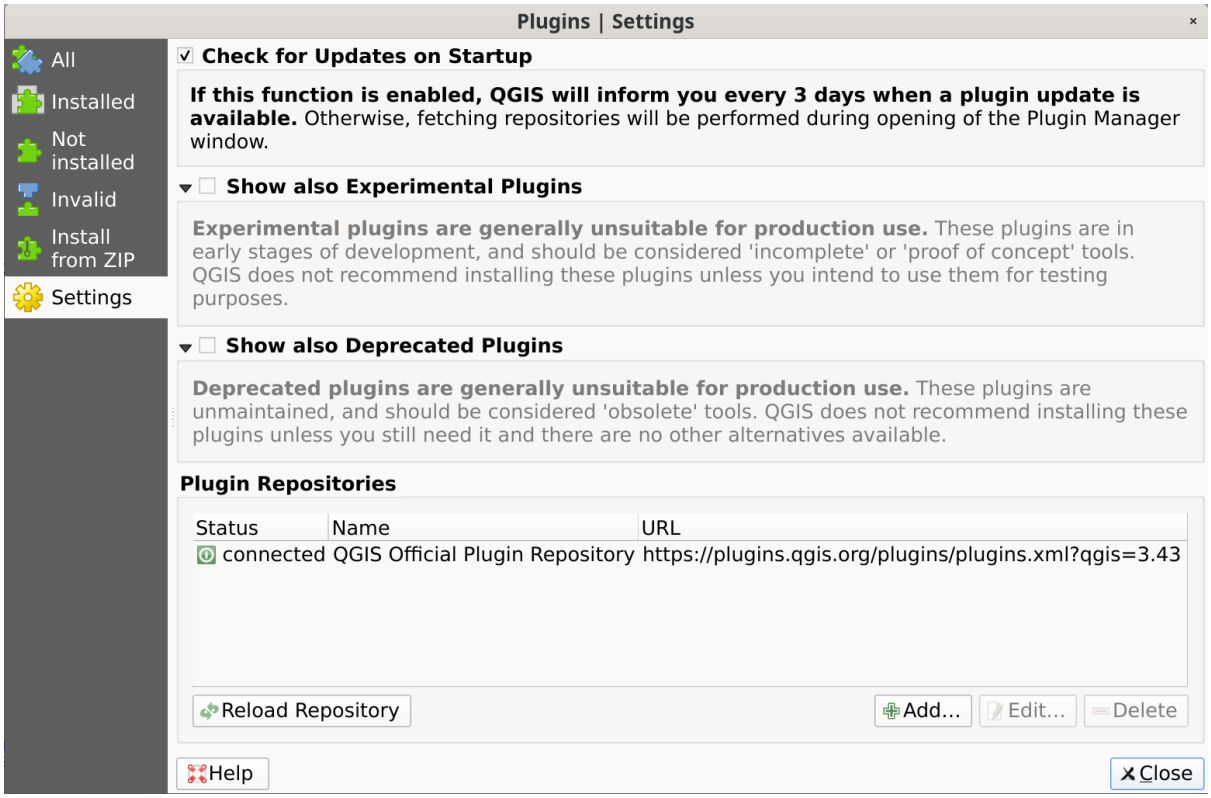


Abb. 25.1: Der Reiter *Einstellungen*

Durchsuchen der Erweiterungen

Die Reiter

Die oberen Reiter im *Erweiterungen*-Dialog bieten Ihnen Listen von Plugins basierend auf ihrem Installations-, Erstellungs- oder Aktualisierungsstatus. Abhängig davon, welche Auswahl Sie auf Einstellungen-Reiter vorgenommen haben, können die verfügbaren Reiter sein:

- *Alle*: zeigt alle verfügbaren Plugins in den aktivierten Repositorien an
- *Installiert*: zeigt sowohl die Plugins, die Sie installiert haben, als auch die Kern-Plugins, die standardmäßig installiert sind und die Sie nicht deinstallieren können
- *Nicht installiert*: zeigt nicht installierte oder noch nicht installierte Plugins in den aktivierten Repositories
- *Neu*: zeigt neu eingestellte Plugins an
- *Aktualisierbar*: zeigt installierte Plugins, die eine neuere Version im Repository veröffentlicht haben
- *Ungültig*: zeigt alle installierten Plugins, die aus irgendeinem Grund nicht funktionieren (fehlende Abhängigkeiten, Fehler beim Laden, inkompatible Funktionen mit der QGIS-Version…)

Am oberen Rand der Registerkarten finden Sie eine *Suchfunktion*, mit der Sie jedes Plugin anhand von Metadaten (Autor, Name, Beschreibung, Tag,…) finden können.

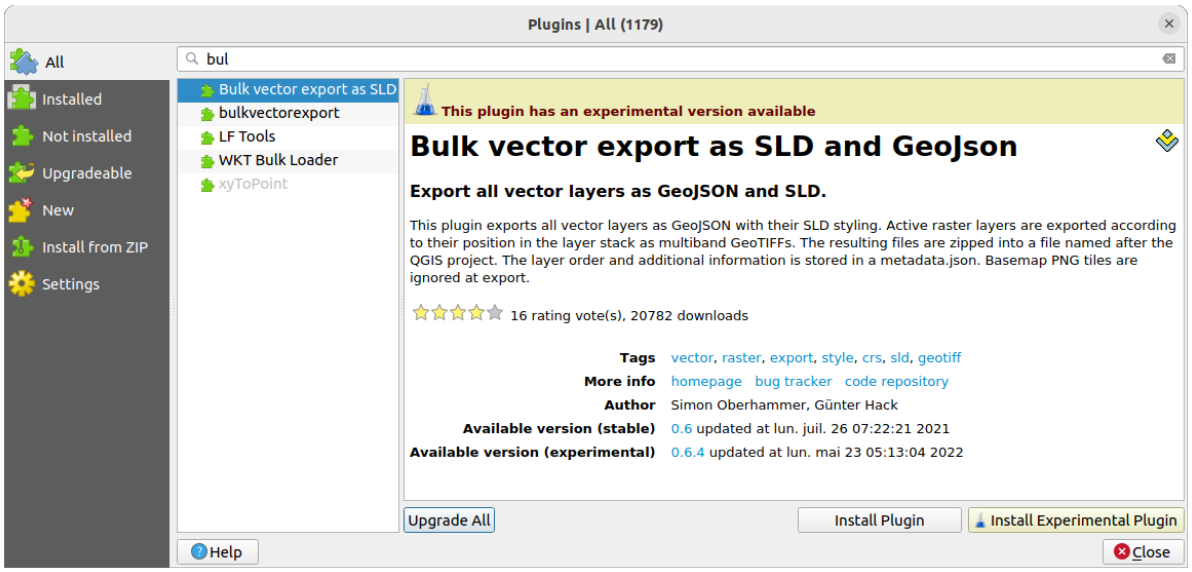


Abb. 25.2: Suche nach einem Plugin auf der Registerkarte *Alle*

Die Erweiterungen

Wählen Sie eine Erweiterung aus, und Sie erhalten im rechten Bedienfeld einige Metadaten angezeigt:

- Informationen darüber, ob das Plugin experimentell ist oder eine experimentelle Version zur Verfügung steht (wenn *Auch experimentelle Erweiterungen anzeigen* aktiviert ist)
- Zusammenfassung und Beschreibung
- Bewertungen (Sie können eine Bewertung für Ihre bevorzugte Erweiterung abgeben)
- Tags
- einige nützliche Links wie die Homepage, die Fehlerverfolgung und das Quellcode-Repositum
- Autoren
- verfügbare Version(en) mit Link zur Download-Seite im Repository oder Pfad zum lokalen Ordner für installierte Plugins

Der *Erweiterungen*-Dialog ermöglicht es Ihnen, mit den neuesten Versionen der Erweiterungen zu interagieren. Wenn experimentelle Version aktiviert sind, werden diese nur angezeigt, wenn sie aktueller sind als die letzte stabile Version. Abhängig von der aktiven Registerkarte und ob das ausgewählte Plugin installiert ist oder nicht, erhalten Sie verschiedene der folgenden Optionen:

- *Erweiterung installieren*: Installiert die letzte stabile Version des ausgewählten Plugins
- *Experimentelle Erweiterung installieren*: Installiert die experimentelle Version des ausgewählten Plugins
- *Erweiterung neu installieren*: installiert die gleiche stabile Version des Plugins, z.B. nachdem es nicht geladen werden konnte
- *Reinstall Experimental Plugin*: installs the same experimental version of the plugin e.g. after it has failed to load
- *Erweiterung aktualisieren*: aktualisiert das ausgewählte Plugin auf seine neueste stabile Version
- *Experimentelle Erweiterung aktualisieren*: aktualisiert das ausgewählte Plugin auf seine neueste experimentelle Version
- *Alle aktualisieren*: aktualisiert alle installierten Plugins auf ihre aktuellere stabile oder experimentelle Version (je nachdem, ob die zuvor installierte Version stabil oder experimentell war).
- *Frühere Version der Erweiterung installieren*: wechselt von der experimentellen Version des Plugins zu seiner vorherigen stabilen Version

- *Experimentelle Erweiterung downgraden*: wechselt von einer experimentellen Version des Plugins zu seiner letzten veröffentlichten experimentellen Version. Dies kann vorkommen, wenn Sie mit einer noch nicht veröffentlichten Version arbeiten.
- *Erweiterung deinstallieren*: entfernt das installierte Plugin aus dem Benutzerprofil

Ein installiertes Plugin zeigt links ein Kontrollkästchen an. Deaktivieren Sie es, um das Plugin vorübergehend zu deaktivieren.

Wenn Sie mit der rechten Maustaste auf ein Plugin in der Liste klicken, können Sie die Liste der Plugins nach verschiedenen Metadaten sortieren. Die neue Reihenfolge gilt für alle Registerkarten. Die Sortieroptionen sind:

- *Nach Name sortieren*
- *Nach Downloads sortieren*
- *Nach Stimmen sortieren*
- *Nach Status sortieren*
- *Nach Erstellungsdatum sortieren*
- *Nach Aktualisierungsdatum sortieren*

Der Reiter „Aus ZIP installieren“

Die Registerkarte *Aus ZIP installieren* bietet ein Dateiselektor-Widget, um Plugins in einem gezippten Format zu importieren, z.B. Plugins, die direkt aus ihrem Repository heruntergeladen wurden. Verschlüsselte Dateien werden unterstützt.

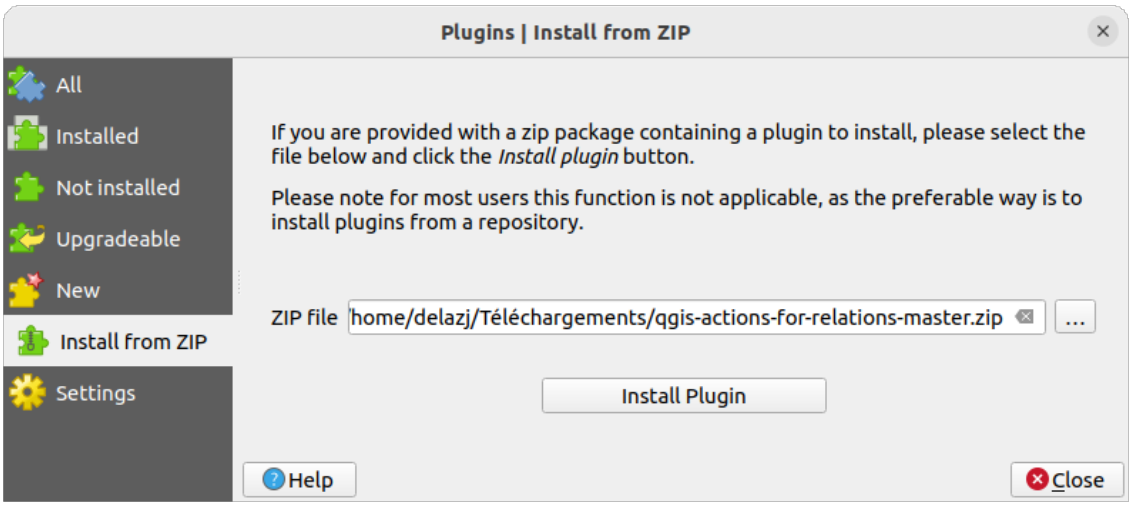


Abb. 25.3: Der *Aus ZIP installieren* Reiter

25.2 QGIS Kern-Erweiterungen verwenden

25.2.1 DB Verwaltung

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.

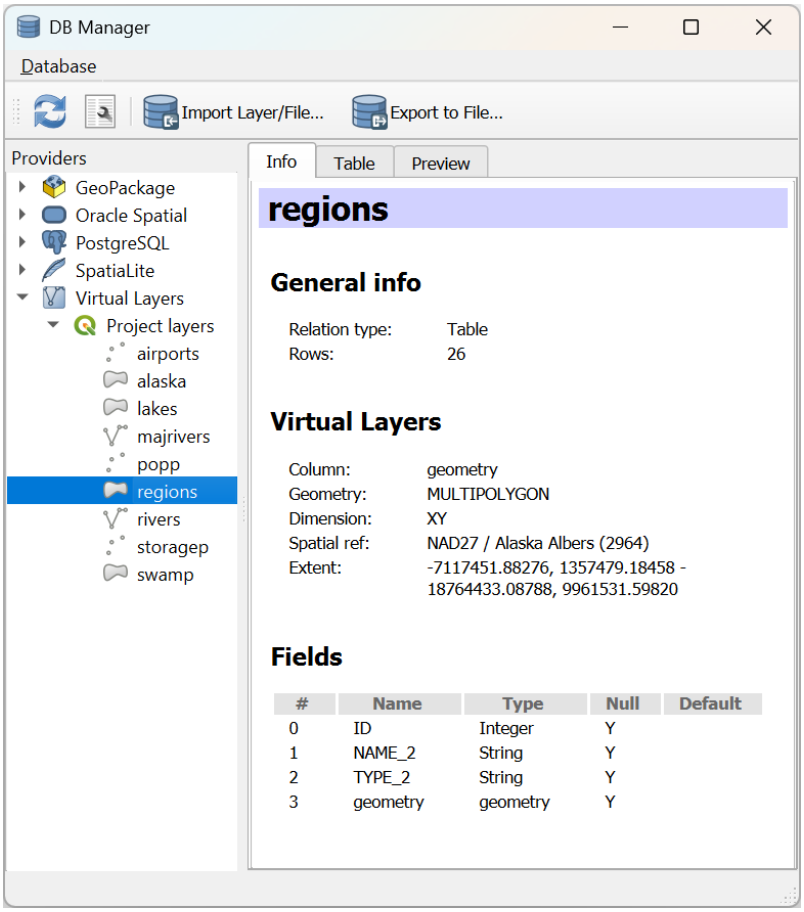


Abb. 25.4: Die DB-Verwaltung

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.

Das Menü *Schema* enthält Werkzeuge zum Erstellen und Löschen (nur wenn leer) von Schemata und, falls Topologie vorhanden ist (z.B. mit PostGIS-Topologie), zum Starten eines *TopoViewer*.

Mit dem Menü *Tabelle* können Sie Tabellen erstellen und bearbeiten sowie Tabellen und Sichten löschen. Es ist auch möglich, Tabellen zu leeren und Tabellen zwischen Schemata zu verschieben. Sie können *Vacuum ausführen* für die ausgewählte Tabelle. Mit *Vacuum* wird Speicherplatz zurückgewonnen und für die Wiederverwendung verfügbar gemacht. *analyze* aktualisiert Statistiken, die zur Ermittlung der effizientesten Art der Ausführung einer Abfrage verwendet werden. *Protokollierung von Änderungen…* ermöglicht das Hinzufügen der Änderungsprotokollierung zu einer Tabelle. Schließlich können Sie *Layer/Datei importieren…* und *In Datei exportieren…*

Bemerkung

Mit dem DB Manager ist es möglich, Kommentare für Tabellen und Spalten einer PostgreSQL Datenbank hinzuzufügen.

Das Fenster *Datenanbieter* listet alle existierenden Datenbanken auf, die von QGIS unterstützt werden. Mit einem Doppelklick können Sie sich mit der Datenbank verbinden. Mit der rechten Maustaste können Sie bestehende Schemata und Tabellen umbenennen und löschen. Tabellen können auch über das Kontextmenü in die Kartenansicht von QGIS eingefügt werden.

Wenn eine Verbindung zu einer Datenbank besteht, bietet das **Hauptfenster** des DB-Managers vier Registerkarten. Die Registerkarte *Info* bietet Informationen über die Tabelle und ihre Geometrie sowie über vorhandene Felder, Beschränkungen und Indizes. Hier können Sie einen räumlichen Index für die ausgewählte Tabelle erstellen. Die Registerkarte *Tabelle* zeigt die Tabelle an, und die Registerkarte *Vorschau* stellt die Geometrien als Vorschau dar. Wenn Sie ein *SQL-Fenster* öffnen, wird es in einer neuen Registerkarte platziert.

Arbeiten mit dem SQL-Fenster

Mit der DB-Verwaltung können Sie SQL-Abfragen für Ihre räumliche Datenbank ausführen. Abfragen können gespeichert und geladen werden, und der *SQL Query Builder* hilft Ihnen bei der Formulierung Ihrer Abfragen. Sie können sogar räumliche Ausgaben anzeigen, indem Sie *Als neuen Layer laden* aktivieren und *Spalte(n) mit eindeutigen Werten* (IDs), *Geometriespalte* und *Layername (Präfix)* angeben. Es ist möglich, einen Teil der SQL zu markieren, um nur diesen Teil auszuführen, wenn Sie `Strg+R` drücken oder auf die Schaltfläche *Ausführen* klicken.

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.

After executing your query, 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.

Die Schaltfläche *Abfrageprotokoll* speichert die letzten 20 Abfragen jeder Datenbank und jedes Anbieters.

Ein Doppelklick auf einen Eintrag fügt die Zeichenfolge in das SQL-Fenster ein.

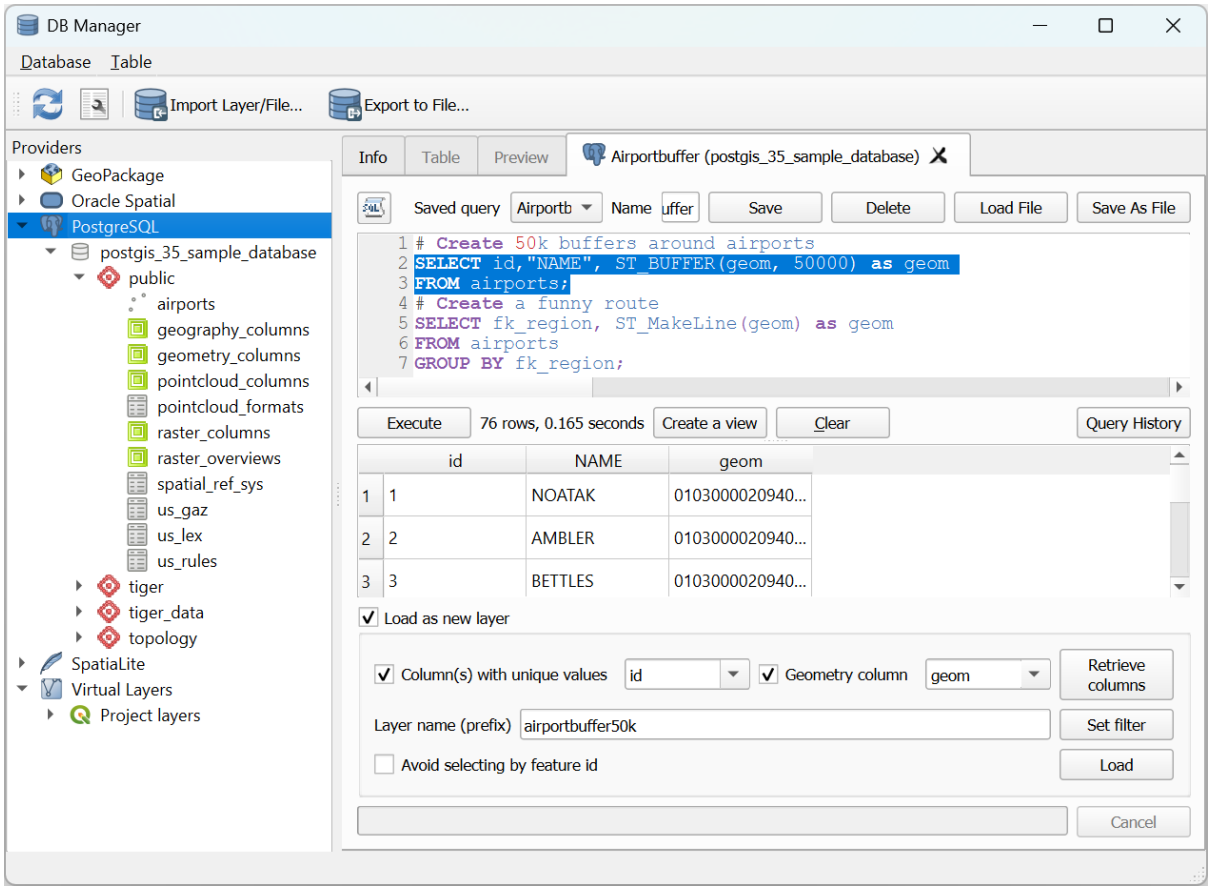


Abb. 25.5: Ausführen von Abfragen im SQL-Fenster der DB-Verwaltung

Bemerkung

Das SQL-Fenster kann auch zur Erstellung virtueller Layer verwendet werden. In diesem Fall wählen Sie im Datenanbieter-Fenster statt einer Datenbank **Projektlayer** unter **Virtuelle Layer** aus, bevor Sie das SQL-Fenster öffnen. Siehe *Erstellen von virtuellen Layern* für Anweisungen zur zu verwendenden SQL-Syntax.

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.

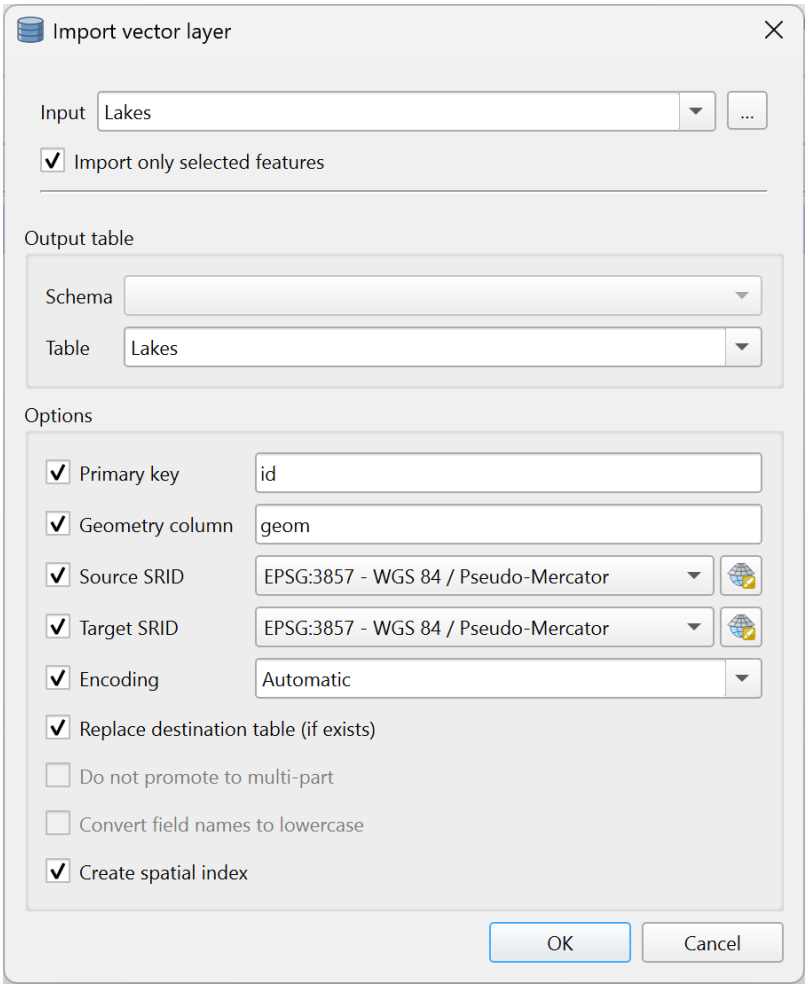


Abb. 25.6: Importing a vector layer into a spatial database using DB Manager

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.

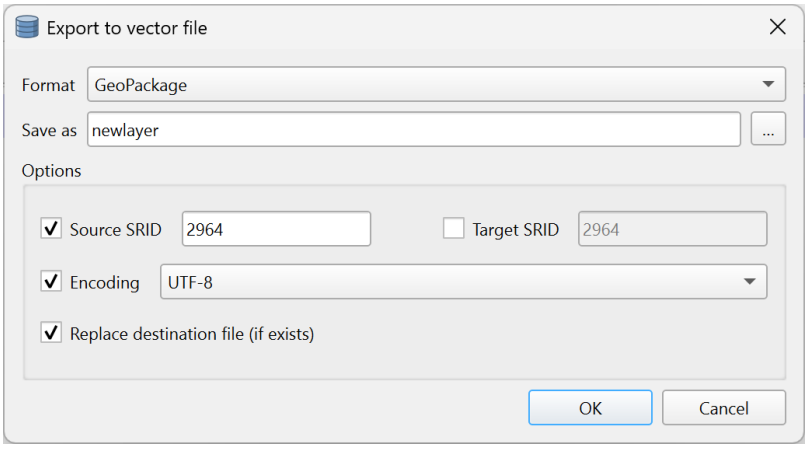


Abb. 25.7: Exporting a table to a vector file using DB Manager

25.2.2 Die Erweiterung Geometrien prüfen

Die Geometrien-prüfen-Erweiterung ist ein leistungsfähiges Kern-Plugin, um die Gültigkeit der Geometrie eines Layers zu prüfen und zu korrigieren. Es ist über das Menü *Vektor* verfügbar (*Geometrien prüfen…*). Ggf. muss es erst in der Verwaltung der Erweiterungen aktiviert werden.

Konfigurieren der Prüfungen

Der Dialog *Geometrien prüfen* zeigt auf der ersten Registerkarte (*Einstellung*) verschiedene, gruppierte Einstellungsmöglichkeiten:

- *Eingabevektorlayer*: zur Auswahl der zu prüfenden Layer. Ein Kontrollkästchen *Nur gewählte Objekte* kann verwendet werden, um die Prüfung auf die Geometrien der gewählten Objekte zu beschränken.
- guilabel:*Erlaubte Geometrietypen* gibt die Möglichkeit, den Geometrietyp des/der Input Layer(s) zu beschränken:
  - Punkt
  - Multipunkt
  - Linie
  - Multilinie

  - ☑ Polygon
  - ☑ Multipolygon
- *Geometriegültigkeit*. Je nach Geometrietyp können Sie wählen zwischen:
  - ☑ *Selbstüberschneidung*
  - ☑ *Doppelte Knoten*
  - ☑ *Eigenverbindung*
  - ☑ *Polygone mit weniger als 3 Punkten*.
- *Geometrieeigenschaften*. Je nach Geometrietyp sind unterschiedliche Optionen verfügbar:
  - ☑ *Polygone und Multipolygone dürfen keine Löcher enthalten*
  - ☑ *Mehrteilige Objekte müssen mehr als einen Teil haben*
  - ☑ *Linien dürfen keine Überlängen haben*
- *Geometriebedingungen*. Ermöglicht das Hinzufügen einer Bedingung zur Validierung der Geometrien:
  - ☑ *Minimale Segmentlänge (Karteneinheiten)* 1,00
  - ☑ *Minimaler Winkel zwischen Segmenten (Grad)* 1,00
  - ☑ *Minimale Polygonfläche (Karteneinheiten²)* 1,00
  - ☑ *keine Splitterpolygone* mit einer *Maximale Dünnheit* 1,00 und einer ☑ *Maximalen Fläche (Karteneinheiten²)* 1,00
- *Topologieprüfungen*. Je nach Geometrietyp sind verschiedene Optionen verfügbar:
  - ☑ *Doppelte Objekte prüfen*
  - ☑ *Objekte in anderen Objekten prüfen*
  - ☑ *Überlappungen kleiner als prüfen (Karteneinheiten²)* 1,00
  - ☑ *Lücken kleiner als prüfen (Karteneinheiten²)* 1,00
  - ☑ *Punkte müssen durch Linien bedeckt sein*
  - ☑ *Punkt muss richtig in einem Polygon liegen*
  - ☑ *Linien dürfen sich nicht mit anderen überschneiden*
  - ☑ *Linien dürfen sich nicht mit Objekten des Layers überschneiden* ...
  - ☑ *Polygone müssen den Grenzen des Layers folgen* ...
- *Toleranz*. Sie können die Toleranz der Prüfung in Einheiten des Map Layers festlegen.
- *Ausgabevektorlayer* bietet die Möglichkeit:
  - ◉ *Eingabelayer modifizieren*
  - ◉ *Neuen Layer erzeugen*

Wenn Sie mit der Konfiguration zufrieden sind, können Sie auf die Schaltfläche *Starten* klicken.

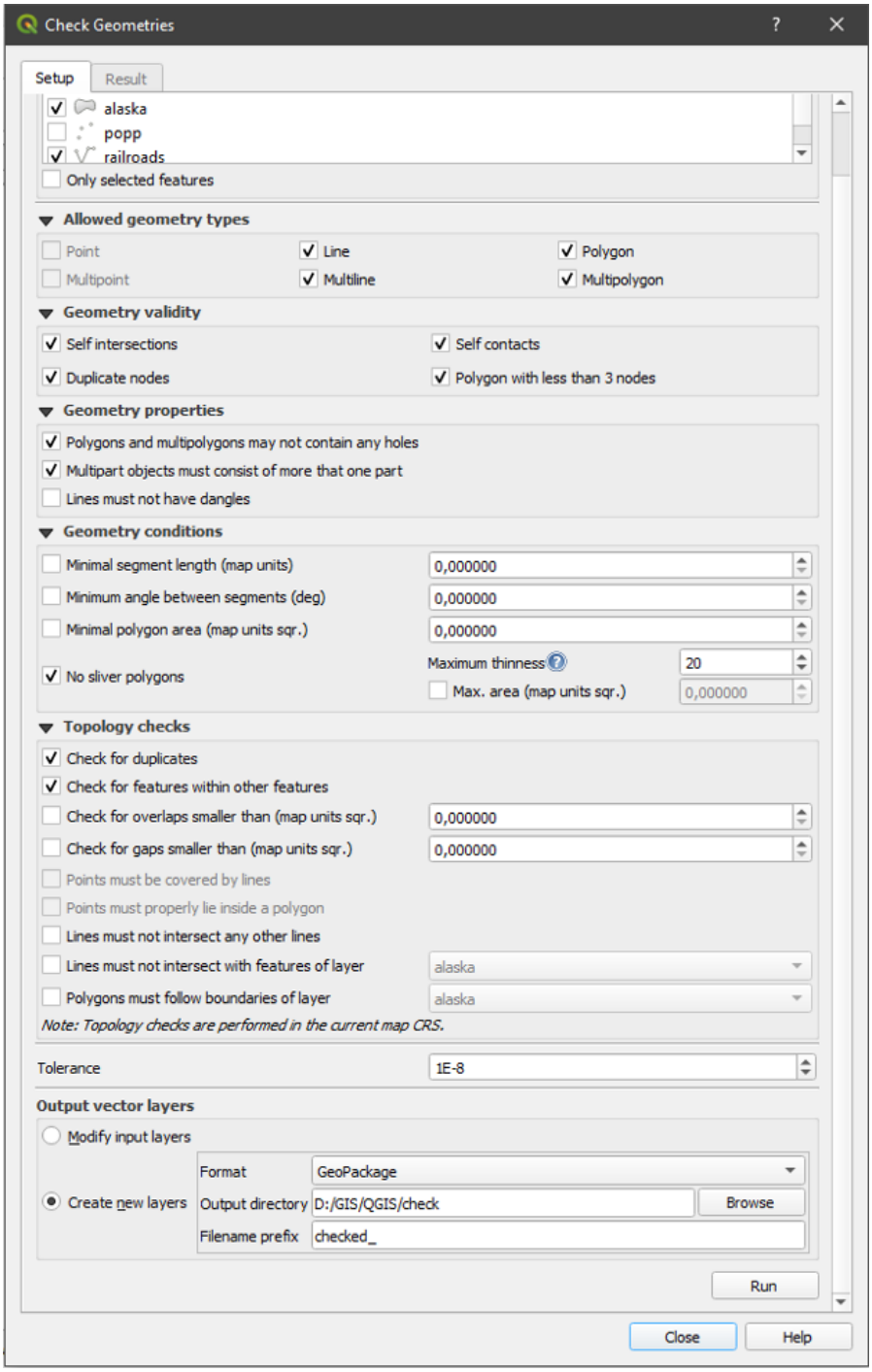


Abb. 25.8: Die Geometrien-prüfen-Erweiterung

Die *Geometrie-Prüfen-Erweiterung* kann die folgenden Fehler finden:

- Selbstschnittpunkte: ein Polygon mit einem Selbstschnittpunkt
- Doppelte Knoten: zwei doppelte Knoten in einem Segment
- Löcher: Loch in einem Polygon
- Segmentlänge: eine Segmentlänge kleiner als ein Schwellenwert
- Mindestwinkel: zwei Segmente mit einem Winkel kleiner als ein Schwellenwert
- Mindestfläche: Polygonfläche kleiner als ein Schwellenwert

- Splitter-Polygon: Dieser Fehler kommt von einem sehr kleinen Polygon (mit kleiner Fläche), mit einem großen Umfang
- Dupliziert Objekte
- Objekt im Objekt
- Überschneidungen: Polygonüberschneidungen
- Lücken: Lücken zwischen Polygonen

Die folgende Abbildung zeigt die verschiedenen Prüfungen, die das Plugin durchführt.

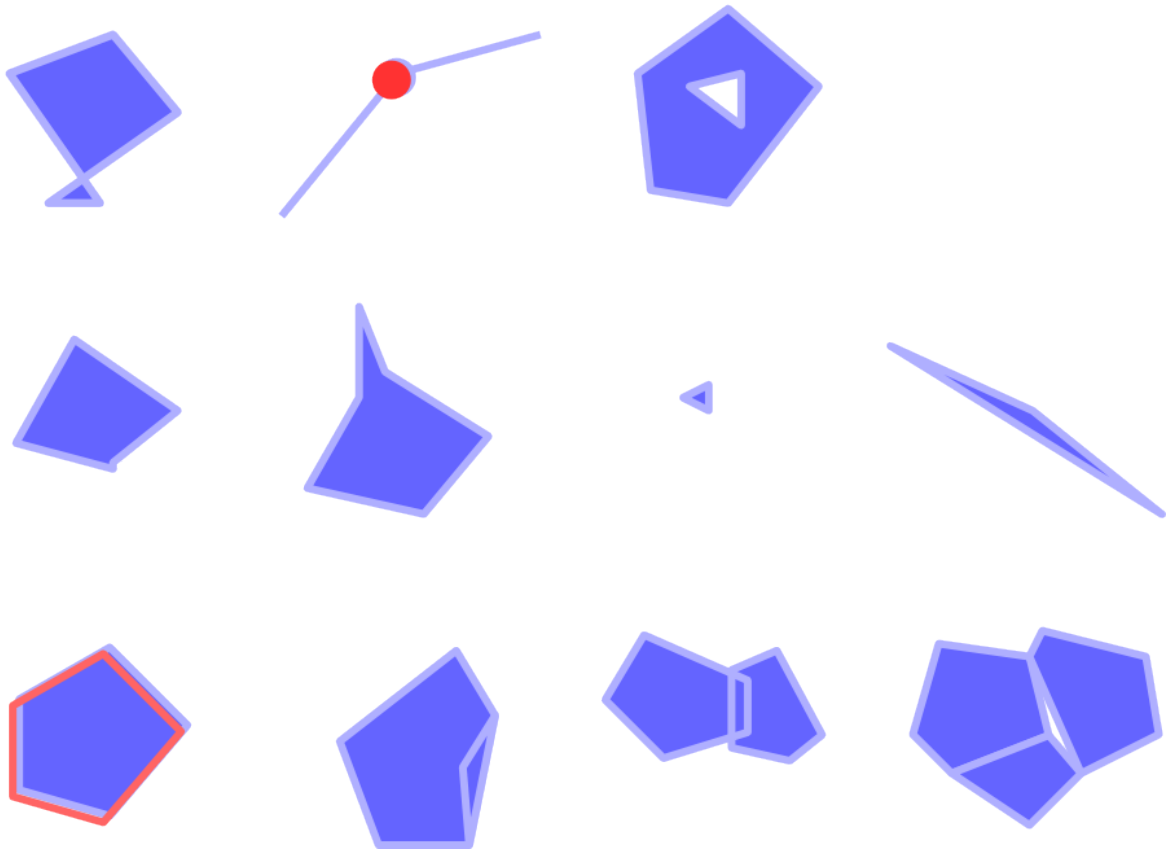


Abb. 25.9: Einige vom Plugin unterstützte Prüfungen

Die Ergebnisse analysieren

Die Ergebnisse erscheinen in der zweiten Registerkarte (*Result*) und als Übersichts-Layer der Fehler in der Kartenansicht (sein Name hat das Standardpräfix `checked_`). Eine Tabelle listet das *Geometry check result* auf, mit einem Fehler pro Zeile und Spalten, die Folgendes enthalten: den Namen des Layers, eine ID, den Fehlertyp, dann die Koordinaten des Fehlers, einen Wert (abhängig vom Fehlertyp) und schließlich die Auflösungsspalte, die die Auflösung des Fehlers angibt. Am unteren Ende dieser Tabelle können Sie den Fehler in verschiedene Dateiformate exportieren. Außerdem gibt es einen Zähler, der die Anzahl der gesamten Fehler und der behobenen Fehler anzeigt.

Sie können eine Zeile auswählen, um den Kartenausschnitt dort zu zentrieren, wo der Fehler aufgetreten ist. Sie können dieses Verhalten ändern, indem Sie unter der Tabelle eine andere Option wählen, außer *Fehler* (Standard) stehen *Objekt*, *Nicht verschieben* zur Verfügung, darüber hinaus kann die Option *Gewähltes Objekte hervorheben* aktiviert/deaktiviert werden.

Unterhalb der Zoom-Aktion, wenn Sie auf die Tabellenzeile klicken, können Sie:

- *Gewählte Objekte in Attributtabelle anzeigen*
- *Gewählte Fehler mit vorgegebener Fehlerlösung beheben*
- *Fehlerlösung zu gewählten Fehlern abfragen* Es wird ein Fenster angezeigt, in dem Sie die Auflösungsmethode auswählen können:
  - Benachbarte Polygone mit der längsten gemeinsamen Kante zusammenfassen
  - Benachbarten Polygone mit der größten Flächen zusammenfassen
  - Benachbarte Polygone mit identischem Attributwert zusammenfassen, wenn vorhanden oder beibehalten
  - Objekt löschen
  - kein Aktion
- *Fehlerlösungsoptionen* ermöglicht es Ihnen, die Standardauflösungsmethode je nach Fehlertyp zu ändern

Tipp

Mehrere Fehler beheben

Sie können mehrere Fehler beheben, indem Sie mehr als eine Zeile in der Tabelle mit der Aktion *Strg+Mausklick* auswählen.

Schließlich können Sie noch dass *Zu verwendende Attribut für Objektzusammenfassung nach Attribut:* festlegen.

25.2.3 MetaSearch Catalog Client

Einleitung

MetaSearch ist ein QGIS-Plugin zur Interaktion mit Metadaten-Katalogdiensten, das sowohl die OGC API - Records als auch die OGC Catalog Service for the Web (CSW) Standards unterstützt.

MetaSearch bietet einen einfachen und intuitiven Ansatz und eine benutzerfreundliche Oberfläche für die Suche in Metadatenkatalogen innerhalb von QGIS.

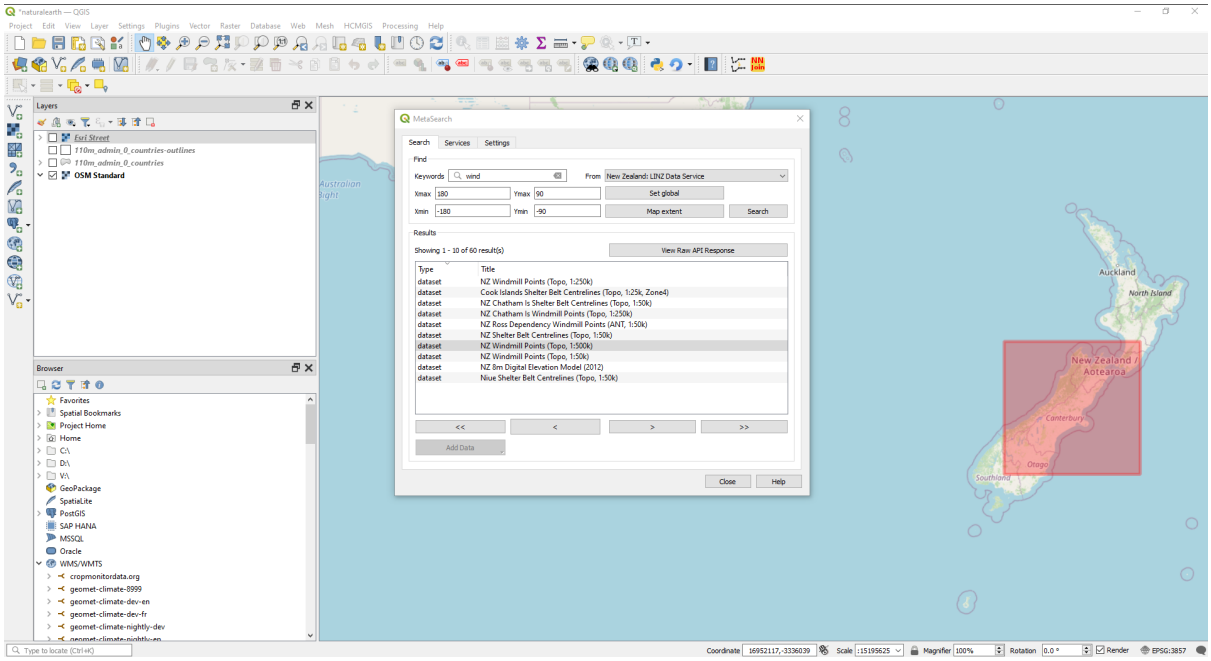


Abb. 25.10: Suche und Ergebnisse von Angeboten in MetaSearch

Arbeiten mit Metadaten-Katalogen in QGIS

MetaSearch ist standardmäßig in QGIS enthalten, mit allen Abhängigkeiten, und kann über den QGIS Plugin Manager aktiviert werden.

OGC API - Records

OGC API - Records ist ein OGC (Open Geospatial Consortium) Standard für die Erkennung von Georessourcen im Web.

OGC CSW (Catalog Service for the Web)

OGC CSW (Catalog Service for the Web) ist eine Spezifikation des OGC (Open Geospatial Consortium), die gemeinsame Schnittstellen zum Auffinden, Durchsuchen und Abfragen von Metadaten über Daten, Dienste und andere potenzielle Ressourcen definiert.

Einrichtung

Um MetaSearch zu starten, klicken Sie auf das entsprechende Symbol in der *Webwerkzeugleiste`oder wählen Sie :menuselection:`Web –> MetaSearch –> MetaSearch* über das QGIS Hauptmenü. Der MetaSearch-Dialog wird angezeigt. Das Hauptfenster besteht aus drei Registerkarten: *Suchen*, *Dienste* und *Einstellungen*.

Katalog-Dienste verwalten

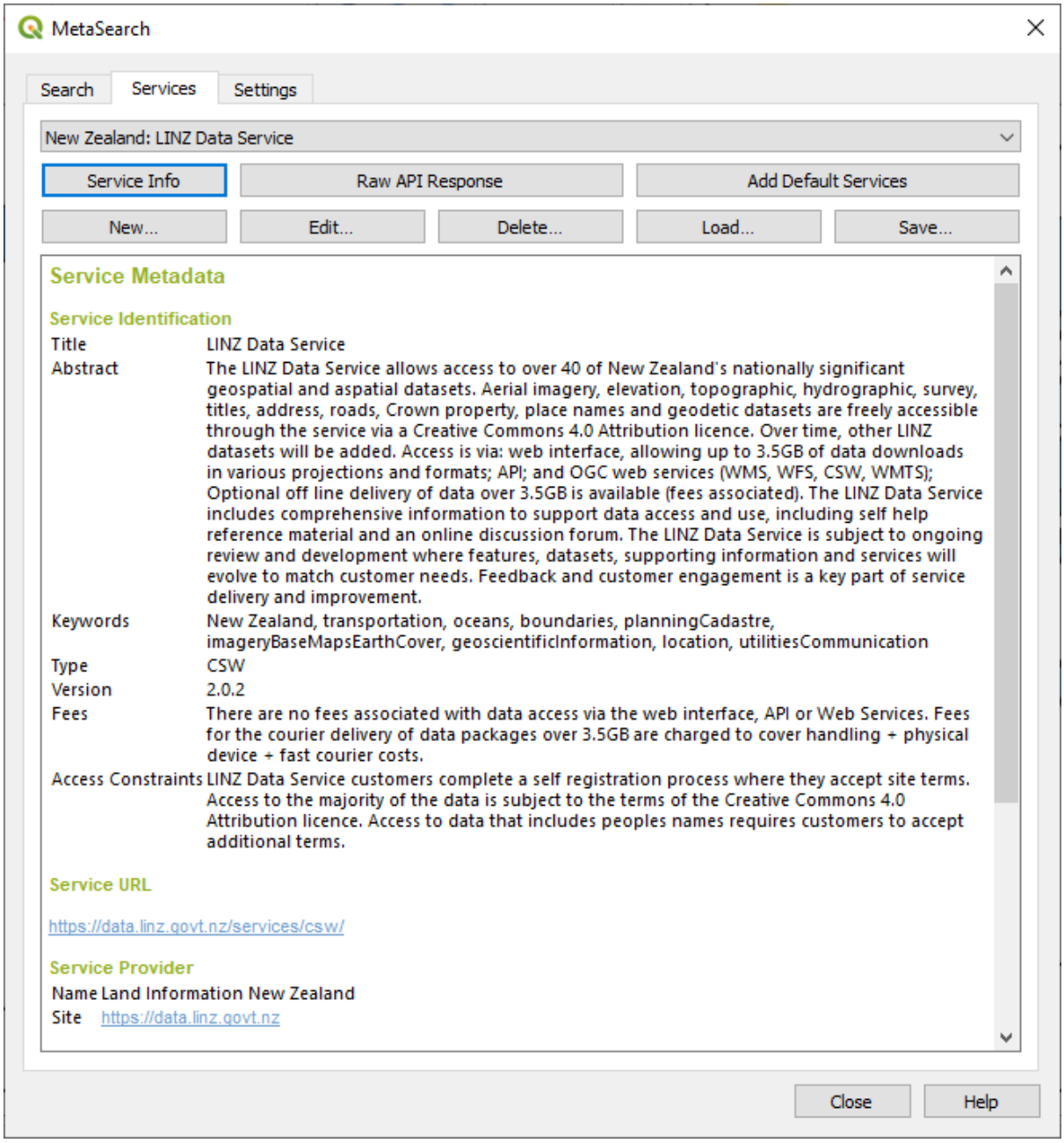


Abb. 25.11: Katalog-Dienste verwalten

Die Registerkarte *Dienste* ermöglicht dem Benutzer die Verwaltung aller verfügbaren Katalogdienste. MetaSearch bietet eine Standardliste von Katalogdiensten, die durch Drücken der Schaltfläche *Vorgabedienste hinzufügen* hinzugefügt werden können.

Um alle aufgelisteten Katalogdiensteinträge zu finden, klicken Sie auf das Dropdown-Auswahlfeld.

Um einen weiteren Katalog-Diensteintrag hinzuzufügen:

1. Klicken Sie auf die Schaltfläche *Neu*
2. Geben Sie einen *Namen* für den Dienst ein, sowie die *URL*. Beachten Sie, dass für OGC CSW 2.0.2 Kataloge nur die Basis-URL erforderlich ist (keine vollständige GetCapabilities-URL). Für OGC API - Records Catalogs sollte die URL der Pfad zum Endpunkt der Sammlung sein.

3. Wenn der Katalog eine Authentifizierung erfordert, geben Sie die entsprechenden Anmeldedaten *Benutzername* und *Passwort* ein.
4. Klicken Sie auf *OK*, um den Dienst in die Liste der Einträge aufzunehmen.

Um einen bestehenden Katalogdiensteintrag zu bearbeiten:

1. Wählen Sie den Eintrag, den Sie bearbeiten möchten
2. Klicken Sie auf die Schaltfläche *Bearbeiten*.
3. Und ändern Sie die Eingaben für *Name* oder *URL*
4. Klicken Sie *OK*.

Um einen Katalogdiensteintrag zu löschen, wählen Sie den Eintrag aus, den Sie löschen möchten, und klicken Sie auf die Schaltfläche *Löschen*. Sie werden aufgefordert, das Löschen des Eintrags zu bestätigen.

MetaSearch ermöglicht das Laden und Speichern von Verbindungen in einer XML-Datei. Dies ist nützlich, wenn Sie Einstellungen zwischen Anwendungen austauschen wollen. Nachfolgend finden Sie ein Beispiel für das XML-Dateiformat.

```
<?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>
```

Um eine Liste von Einträgen zu laden:

1. Klicken Sie auf die Schaltfläche *Laden*. Ein neues Fenster wird angezeigt.
2. Klicken Sie auf die Schaltfläche *Durchsuchen* und navigieren Sie zu der XML-Datei mit den Einträgen, die Sie laden möchten.
3. Klicken Sie auf *Öffnen*. Die Liste der Einträge wird angezeigt.
4. Wählen Sie die Einträge, die Sie hinzufügen möchten, aus der Liste aus und klicken Sie auf *Laden*.

Klicken Sie auf die Schaltfläche *Dienstinfo*, um Informationen über den ausgewählten Katalogdienst anzuzeigen, z. B. Dienstkennung, Dienstanbieter und Kontaktinformationen. Wenn Sie die unbearbeitete API-Antwort anzeigen möchten, klicken Sie auf die Schaltfläche *API-Antwort in Rohform*. Es öffnet sich ein separates Fenster, in dem die Serverinformationen im JSON- oder XML-Rohformat angezeigt werden.

Suche in den Katalog-Diensten

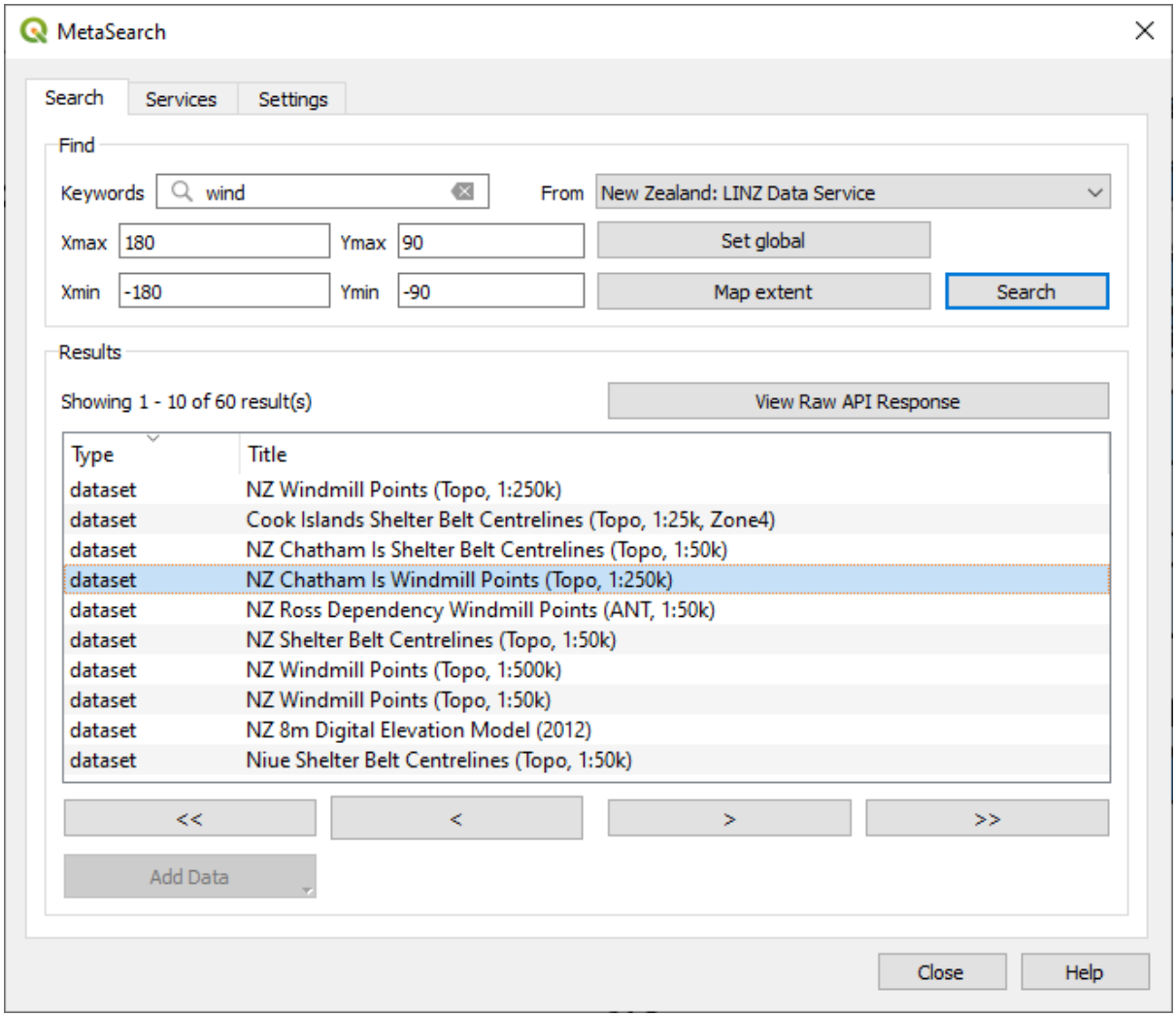


Abb. 25.12: Suche in den Katalog-Diensten

Die Registerkarte *Suchen* ermöglicht es dem Benutzer, die Katalogdienste nach Daten und Diensten abzufragen, verschiedene Suchparameter einzustellen und die Ergebnisse anzuzeigen.

Die folgenden Suchparameter sind verfügbar:

- *Schlüsselwörter*: Freitext-Suchbegriffe;
- *Von*: der Katalogdienst, in dem die Abfrage durchgeführt werden soll;
- **Suchraum**: der räumliche Bereich, der gefiltert werden soll, ist definiert durch *X-max*, *X-min*, *Y-max* und *Y-min* in Grad. Klicken Sie auf *Welt setzen*, um eine globale Suche durchzuführen, klicken Sie auf *Kartenausdehnung*, um eine Suche im sichtbaren Bereich durchzuführen, oder geben Sie die Werte manuell ein.

Wenn Sie auf die Schaltfläche *Suchen* klicken, wird der ausgewählte Metadatenkatalog durchsucht. Die Suchergebnisse werden in einer Liste angezeigt und können durch einen Klick auf die Spaltenüberschrift sortiert werden. Mit den Richtungsschaltflächen unter den Suchergebnissen können Sie durch die Suchergebnisse navigieren.

Wählen Sie ein Ergebnis und:

- Klicken Sie auf die Schaltfläche *API-Antwort in Rohform anzeigen*, um ein Fenster mit der Antwort des Dienstes im Rohformat JSON oder XML zu öffnen.

- Wenn dem Metadatensatz ein Begrenzungsrahmen zugeordnet ist, wird ein Fußabdruck des Begrenzungsrahmens auf der Karte angezeigt.
- Doppelklicken Sie auf den Datensatz, um die Metadaten des Datensatzes mit allen zugehörigen Zugriffslinks anzuzeigen. Wenn Sie auf einen Link klicken, wird der Link im Webbrowser des Benutzers geöffnet.
- Wenn es sich bei dem Datensatz um einen unterstützten Webservice (WMS/WMTS, WFS, WCS, ArcGIS REST Service usw.) oder eine GIS-Datei handelt, wird die Schaltfläche *Daten hinzufügen* aktiviert. Wenn Sie auf diese Schaltfläche klicken, prüft MetaSearch, ob es sich um einen gültigen OWS handelt. Der Dienst wird dann der entsprechenden QGIS-Verbindungsliste hinzugefügt, und der entsprechende Verbindungsdialog wird angezeigt. Wenn Sie *GIS-Datei hinzufügen* wählen, wird die verknüpfte GIS-Datei zum aktuellen Projekt hinzufügt.

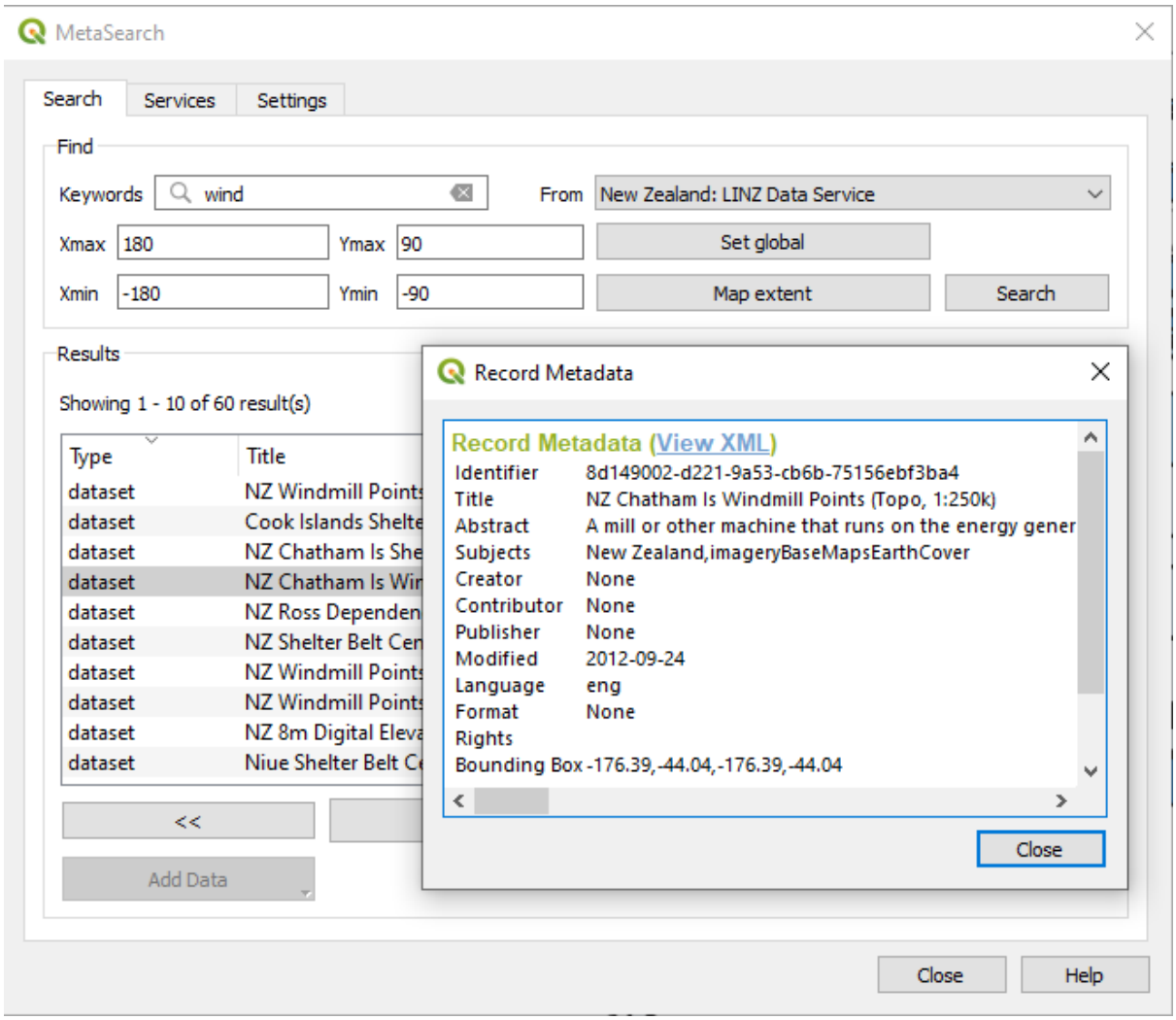


Abb. 25.13: Anzeige des Metadatensatzes

Einstellungen

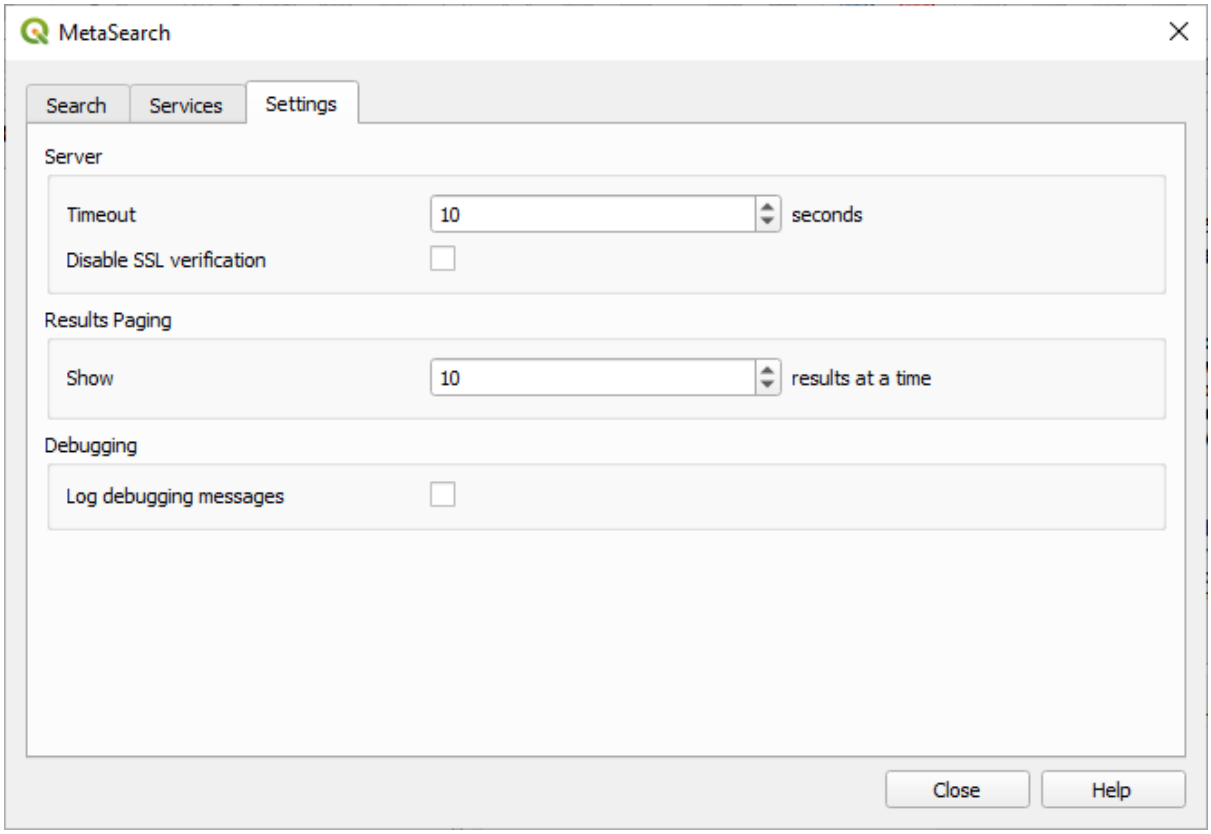


Abb. 25.14: MetaSearch-Einstellungen

Sie können MetaSearch mit den folgenden *Einstellungen* feinabstimmen:

- *Zeitüberschreitung*: bei der Suche in Metadatenkatalogen die Anzahl der Sekunden zwischen Verbindungsversuchen. Der Standardwert ist 10.
- *SSL-Verifikation abschalten*: Option zum Abschalten der SSL-Verifikation.
- *Ergebnisseiten*: bei der Suche in Metadatenkatalogen die Anzahl der Ergebnisse, die pro Seite angezeigt werden. Der Standardwert ist 10.
- *Log debugging messages*: option to enable the logging of debugging messages in the *Das Protokoll-Bedienfeld*.

Katalog-Server-Fehler

In manchen Fällen funktioniert der Katalog in einem Webbrowser, aber nicht in MetaSearch. Dies kann an der Konfiguration/Einrichtung des Katalogservers liegen. Anbieter von Katalogservern sollten sicherstellen, dass die URLs in ihrer Konfiguration konsistent und auf dem neuesten Stand sind (dies ist z.B. bei HTTP -> HTTPS-Umleitungen nicht immer der Fall). Bitte lesen Sie den pycsw FAQ item für eine genauere Erklärung von Problemen und Lösungen. Obwohl der FAQ-Artikel pycsw-spezifisch ist, kann er auch allgemein für andere Katalogserver angewendet werden.

25.2.4 Offline-Bearbeitung Erweiterung

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.

Das Offline-Bearbeitung Plugin automatisiert die Synchronisation, indem es den Inhalt der Datenquelle in eine SpatiaLite oder GeoPackage Datenbank kopiert und die Offline-Bearbeitungen in speziellen Tabellen speichert. Nach einer erneuten Verbindung mit dem Netzwerk können die Offline-Bearbeitungen auf den Master-Datensatz übertragen werden.

So verwenden Sie das Plugin:

1. Open a project with some vector layers (e.g., from an Esri Shapefile, PostgreSQL or WFS-T datasource).
2. Vorausgesetzt, Sie haben das Plugin bereits aktiviert (siehe *Kern-Erweiterungen und externe Erweiterungen*), gehen Sie zu *Datenbank* ► *Offline-Bearbeitung* ► *In Offline-Projekt konvertieren*. Es öffnet sich der Namensgeber-Dialog.
3. Wählen Sie den *Speichertyp*. Es kann sich um den Datenbanktyp *GeoPackage* oder *SpatiaLite* handeln.
4. Verwenden Sie die Schaltfläche *Durchsuchen*, um den Ort der Datenbank anzugeben, in der die *Offline-Daten* gespeichert werden sollen. Dabei kann es sich um eine vorhandene oder eine zu erstellende Datei handeln.
5. Markieren Sie im Abschnitt *Ferne Layer auswählen* die Layer, die Sie speichern möchten. Der Inhalt der Layer wird in Datenbanktabellen gespeichert.

   Bemerkung

   Da die Zieldatenbankformate keine native Listenunterstützung haben, wandelt das Plugin für die Offline-Bearbeitung Listenfelder {string, number} in Stringfelder um, deren Werte durch Kommas getrennt sind. Dies ermöglicht das Lesen und Bearbeiten des Inhalts dieser Felder im Offline-Modus.

   Wenn Sie sowohl das Feld des ursprünglichen Layers als auch das des Offline-Layers bearbeiten möchten, können Sie die Ausdrucksfunktionen *try()* und *array()* verwenden, z. B.:

   ```
   try(array_contains("field",1),array_contains(string_to_array("field"),1))
   ```

6. Sie können das Kontrollkästchen *Nur gewählte Objekte synchronisieren, wenn eine Auswahl vorliegt* aktivieren, um nur eine Teilmenge zu speichern und zu bearbeiten. Dies kann bei großen Layern von hohem Wert sein.

   Das ist alles!
7. Speichern Sie Ihr Projekt und bringen Sie es vor Ort.
8. Bearbeiten Sie die Layer offline.
9. Nachdem Sie wieder verbunden sind, laden Sie die Änderungen mit *Datenbank* ► *Offline-Bearbeitung* ► *Synchronisieren* hoch.

Bemerkung

Layer, die offline verwendet werden, sind im Bedienfeld *Layer* mit dem Symbol gekennzeichnet.

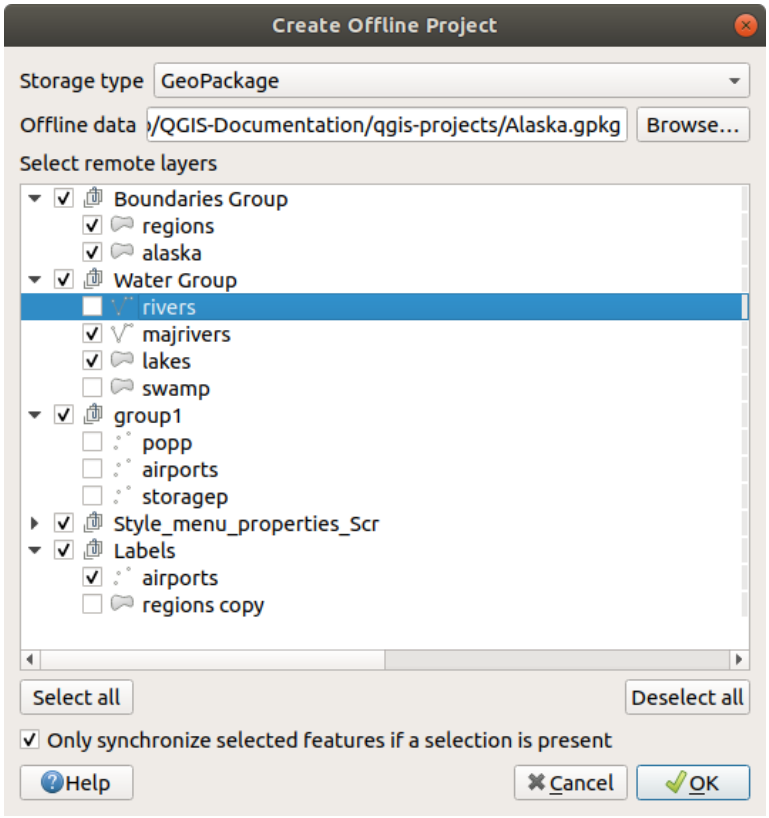


Abb. 25.15: Ein Offline-Projekt erstellen

25.2.5 Die Topologie-prüfen-Erweiterung

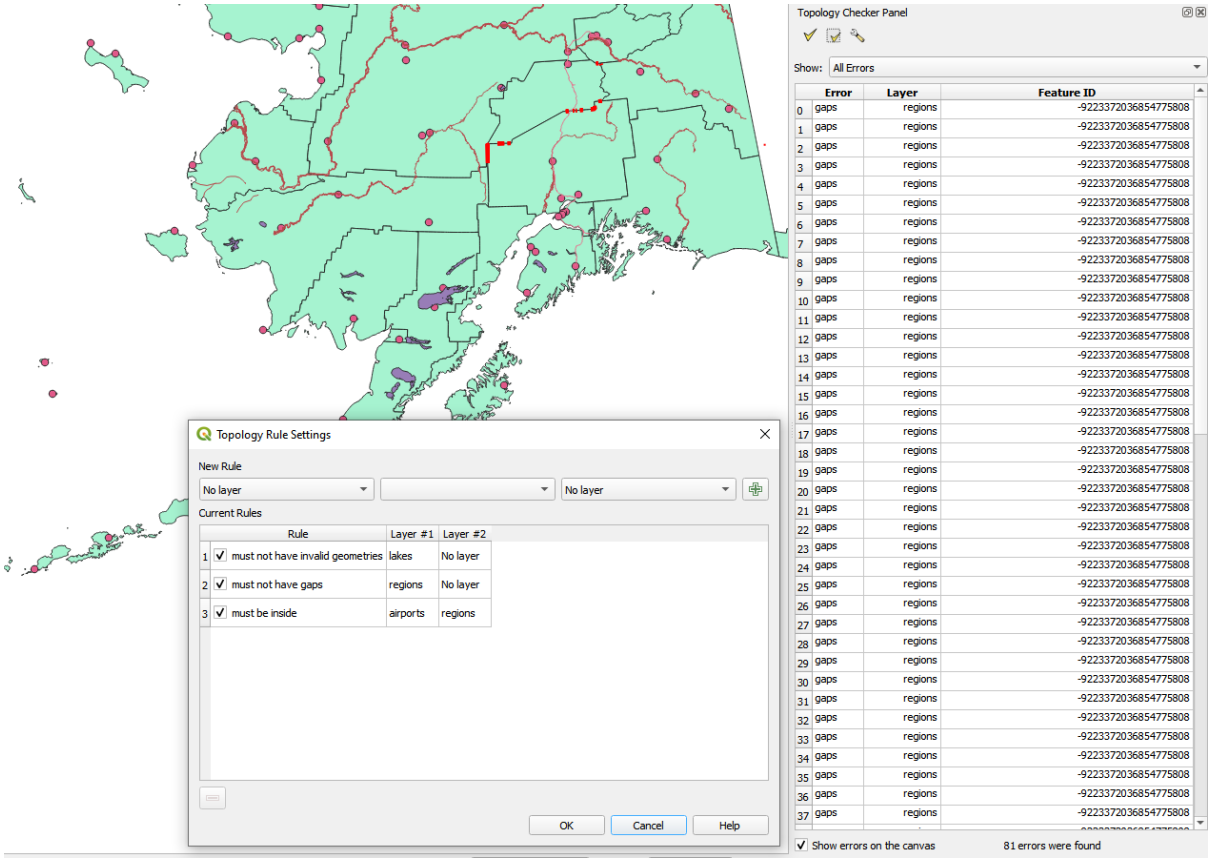


Abb. 25.16: Die Topologie-prüfen-Erweiterung

Die Topologie beschreibt die Beziehungen zwischen Punkten, Linien und Polygonen, die die Objekte einer geografischen Region darstellen. Mit dem Topology Checker Plugin können Sie Ihre Vektordateien durchsehen und die Topologie mit verschiedenen Topologieregeln überprüfen. Diese Regeln prüfen anhand von räumlichen Beziehungen, ob Ihre Objekte die Bedingungen ‚gleich', ‚enthalten', ‚bedecken', ‚überdecken', ‚kreuzen', ‚disjunkt', ‚schneiden', ‚überlappen', ‚berühren' oder ‚innerhalb' erfüllen. Es hängt von Ihren individuellen Fragen ab, welche Topologieregeln Sie auf Ihre Vektordaten anwenden (z. B. werden Sie normalerweise keine Überschneidungen in Linien-Layern akzeptieren, aber wenn sie Sackgassen darstellen, werden Sie sie nicht aus Ihrem Layer entfernen).

QGIS verfügt über eine eingebaute topologische Bearbeitungsfunktion, welche sich hervorragend eignet, um neue Objekte ohne Fehler zu erstellen. Aber bestehende Datenfehler und vom Benutzer verursachte Fehler sind schwer zu finden. Dieses Plugin hilft Ihnen, solche Fehler anhand einer Liste von Regeln zu finden.

So aktivieren Sie das *Topologie-Prüfung* Plugin:

1. Gehen Sie zum Menü *Erweiterungen*
2. Öffnen Sie *Erweiterungen verwalten und installieren* und wählen Sie den Reiter *Installiert*
3. Aktivieren Sie *Topologie-Prüfung*
4. Schließen Sie den Dialog *Erweiterungen*. Dem Menü *Vektor* wurde ein neuer Eintrag *Topologie-Prüfung* hinzugefügt.

Nachdem Sie die *Topologie-Prüfung* aktiviert haben, öffnen Sie das Fenster und wählen *Konfigurieren*, um Ihre Topologie-Regeln zu erstellen.

Für **Punkt-Layer** stehen die folgenden Regeln zur Verfügung:

- *muss abgedeckt sein durch*: Hier können Sie einen Vektorlayer aus Ihrem Projekt auswählen. Punkte, die nicht durch den angegebenen Layer abgedeckt sind, erscheinen im Feld „Fehler“.
- *muss abgedeckt sein durch Endpunkte von*: Hier können Sie einen Layer aus Ihrem Projekt auswählen.
- *muss innerhalb sein von*: Hier können Sie einen Polygonlayer aus Ihrem Projekt auswählen. Die Punkte müssen innerhalb eines Polygons liegen. Andernfalls schreibt QGIS einen ‚Fehler‘ für den Punkt.
- *darf keine Duplikate enthalten*: Wenn ein Punkt zweimal oder öfter dargestellt wird, erscheint er im Feld „Fehler“.
- *darf keine ungültige Geometrie haben*: Prüft, ob die Geometrien gültig sind.
- *Darf keine mehrteiligen Geometrien enthalten*: Alle mehrteiligen Punkte werden in das Feld „Fehler“ geschrieben.

Für **Linien-Layer** stehen die folgenden Regeln zur Verfügung:

- *Endpunkte müssen abgedeckt sein durch*: Hier können Sie einen Punkt-Layer aus Ihrem Projekt auswählen.
- *darf keine Überstände (Dangles) haben*: Dadurch werden die Überstände im Layer angezeigt.
- *darf keine Duplikate haben*: Immer wenn ein Objekt zweimal oder öfter dargestellt wird, erscheint es im Feld „Fehler“.
- *darf keine ungültige Geometrie haben*: Prüft, ob die Geometrien gültig sind.
- *darf keine mehrteiligen Geometrien enthalten*: Manchmal ist eine Geometrie eigentlich eine Sammlung einfacher (einteiliger) Geometrien. Eine solche Geometrie wird als mehrteilige Geometrie bezeichnet. Enthält sie nur eine Art von einfacher Geometrie, wird sie Multipunkt-, Multilinien- oder Multipolygongeometrie genannt. Alle mehrteiligen Linien werden in das Feld „Fehler“ geschrieben.
- *soll keine Doppelpunkte/pseudos enthalten*: Der Endpunkt einer Liniengeometrie sollte mit den Endpunkten von zwei anderen Geometrien verbunden sein. Wenn der Endpunkt nur mit einem Endpunkt einer anderen Geometrie verbunden ist, wird der Endpunkt als Pseudoknoten bezeichnet.

Für **Polygon-Layer** stehen die folgenden Regeln zur Verfügung:

- *muss enthalten*: Polygon Layer muss mindestens eine Punktgeometrie aus dem zweiten Layer enthalten.
- *darf keine Duplikate haben*: Polygone desselben Layers dürfen keine identischen Geometrien aufweisen. Wenn ein Polygonobjekt zweimal oder öfter dargestellt wird, erscheint es im Feld „Fehler“.
- *darf keine Lücken haben*: Benachbarte Polygone sollten keine Lücken zwischen ihnen bilden. Verwaltungsgrenzen könnten als Beispiel genannt werden (Polygone der US-Bundesstaaten haben keine Lücken zwischen sich…).
- *darf keine ungültigen Geometrien haben*: Prüft, ob die Geometrien gültig sind. Einige der Regeln, die eine gültige Geometrie definieren, sind:
  - Polygonringe müssen geschlossen sein.
  - Ringe, die Löcher definieren, müssen innerhalb von Ringen liegen, die äußere Grenzen definieren.
  - Ringe dürfen sich nicht selbst schneiden (sie dürfen sich weder berühren noch kreuzen).
  - Ringe dürfen andere Ringe nicht berühren, außer an einem Punkt.
- *Darf keine mehrteiligen Geometrien haben*: Manchmal besteht eine Geometrie aus einer Sammlung von einfachen (einteiligen) Geometrien. Eine solche Geometrie wird als mehrteilige Geometrie bezeichnet. Enthält sie nur eine Art von einfacher Geometrie, spricht man von Multipunkt-, Multilinien- oder Multipolygongeometrie. Zum Beispiel kann ein Land, das aus mehreren Inseln besteht, als Multipolygon dargestellt werden.
- *Darf sich nicht überlappen*: Benachbarte Polygone sollten keine gemeinsame Fläche haben.
- *Darf sich nicht überlappen mit*: Benachbarte Polygone eines Layers dürfen keine gemeinsame Fläche mit Polygonen eines anderen Layers haben.

Wenn Sie eine *Neue Regel* erstellen wollen, klicken Sie auf die Regel hinzufügen, um sie in *Aktuelle Regeln* aufzunehmen. Sie können einzelne Regeln aktivieren oder deaktivieren, indem Sie auf das entsprechende Kontrollkästchen klicken. Wenn Sie mit der rechten Maustaste auf eine Regel klicken, stehen Ihnen die folgenden Optionen zur Verfügung:

- *Alle wählen*
- *Aktivieren* oder *Deaktivieren* der ausgewählten Regel
- *Aktivierung umschalten* der ausgewählten Regeln
- *Löschen* der ausgewählten Regel. Dies kann auch mit der Schaltfläche Lösche gewählte Regeln erreicht werden

Drücken Sie *OK* und wählen Sie dann aus dem Bedienfeld *Topologie-Prüfungsfenster*:

- Alle prüfen: wendet die aktiven Regeln auf alle Objekte des/der beteiligten Layer an
- oder Prüfe Ausmasse: wendet die aktiven Regeln auf die Objekte des/der beteiligten Layer(s) innerhalb der aktuellen Kartenansicht an. Die Schaltfläche wird gedrückt gehalten und die Ergebnisse werden aktualisiert, wenn sich die Kartenansicht ändert.

Die Fehler werden in der Ergebnistabelle mit Fehlertyp, Layer und Objekt-ID angezeigt. Verwenden Sie das Menü *Filtert Fehler nach Regel*, um die Fehler nach einem bestimmten Fehlertyp zu filtern.

Aktivieren Sie das Kontrollkästchen *Fehler auf Karte anzeigen*, um die Fehlerposition auf der Kartenansicht anzuzeigen. Wenn Sie auf eine Zeile in der Tabelle klicken, wird die Kartenansicht auf das betreffende Objekt gezoomt, wo Sie die *QGIS-Digitalisierungswerkzeuge* verwenden können, um den Fehler zu beheben.

Nachfolgend finden Sie eine Liste der Kern-Erweiterungen, die mit QGIS geliefert werden. Sie sind nicht unbedingt standardmäßig aktiviert.

Icon	Erweiterung	Beschreibung	Handbuch Referenz
	DB Manager	Verwalten Sie Ihre Datenbanken in QGIS	*DB Verwaltung*
	Geometrieprüfung	Überprüfung und Reparatur von Fehlern in Vektorgeometrien	*Die Erweiterung Geometrien prüfen*
	GRASS 8	GRASS Funktionalität	*GRASS GIS Integration*
	GRASS GIS provider	GRASS GIS Verarbeitungsfunktionalität	*GRASS GIS Integration*
	MetaSearch Catalog Client	Interaktion mit Metadatenkatalogdiensten (CSW)	*MetaSearch Catalog Client*
	Offline-Bearbeitung	Offline-Bearbeitung und Synchronisierung mit der Datenbank	*Offline-Bearbeitung Erweiterung*
	Processing	Framework für die Verarbeitung räumlicher Daten	*Die QGIS Verarbeitungsumgebung*
	Topologie-Prüfung	Topologische Fehler in Vektorlayern finden	*Die Topologie-prüfen-Erweiterung*

Bemerkung

To use the Core Plugins GRASS 7, GRASS GIS provider, they have to be configured. Informations can be found *here*.

25.3 QGIS Python Konsole

Wie Sie später in diesem Kapitel sehen werden, wurde QGIS mit einer Plugin-Architektur entwickelt. Plugins können in Python geschrieben werden, einer sehr bekannten Sprache in der Welt der Geodaten.

QGIS verfügt über eine Python-API (siehe PyQGIS Developer Cookbook für einige Code-Beispiele), um dem Benutzer die Interaktion mit seinen Elementen (Layern, Objekten oder Schnittstellen) zu ermöglichen. QGIS hat auch eine Python-Konsole.

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 Die interaktive Konsole

Die Konsole ist ein Python-Interpreter, mit dem Sie Python-Befehle ausführen können. Module von QGIS (analysis, core, gui, server, processing, 3d) und Qt (QtCore, QtGui, QtNetwork, QtWidgets, QtXml) sowie die Python-Module math, os, re und sys sind bereits importiert und können direkt verwendet werden.

Die interaktive Konsole besteht aus einer Symbolleiste, einem Eingabebereich und einem Ausgabebereich.

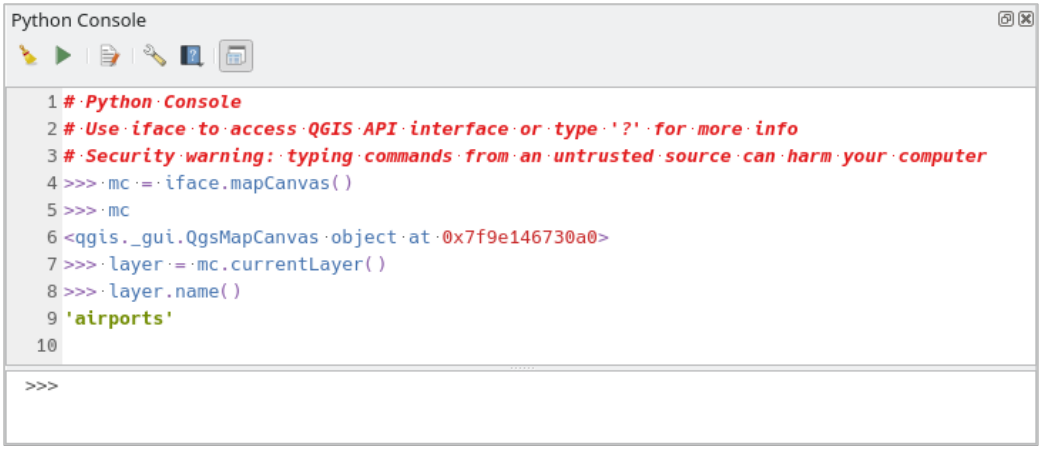


Abb. 25.17: Die Python-Konsole

Werkzeugleiste

Die Symbolleiste bietet folgende Werkzeuge:

- Konsole leeren, um den Ausgabebereich zu löschen;
- Befehl ausführen im Eingabebereich verfügbar: dasselbe wie das Drücken von `Enter`;
- Editor anzeigen: schaltet die Sichtbarkeit von *Der Code-Editor* um;
- |Optionen| Optionen…: öffnet den Dialog zur Konfiguration der *Python-Konsole* in den Optionen;
- Hilfe… bietet ein Menü für den Zugriff auf verschiedene Dokumentationen:
  - *Python-Konsole-Hilfe* (diese Dokumentation)
  - PyQGIS API Dokumentation
  - PyQGIS Cookbook

- Dock Code Editor zum Andocken bzw. Abdocken des Bedienfeldes in der QGIS-Oberfläche

Eingabebereich

Die wichtigsten Objekte des Eingabebereichs der Konsole sind:

- Code-Vervollständigung, Hervorhebung der Syntax und Calltips für die folgenden APIs:
  - Python
  - PyQGIS
  - PyQt5
  - QScintilla2
  - osgeo-gdal-ogr
  - osgeo-geos
- `Strg+Alt+Leertaste`, um die Liste der automatischen Vervollständigung anzuzeigen, wenn sie in *Python* aktiviert ist;
- Führen Sie Code-Schnipsel aus dem Eingabebereich aus, indem Sie `Enter` oder *Befehl ausführen* drücken;
- Führen Sie Code-Schnipsel aus dem Ausgabebereich aus, indem Sie die Option *Gewähltes eingeben* aus dem Kontextmenü verwenden oder `Strg+E` drücken;
- Navigieren Sie vom Eingabebereich aus mit den Pfeiltasten `Auf` und `Ab` durch den Befehlsverlauf und führen Sie den gewünschten Befehl aus;
- `Strg+Umschalt+Leertaste`, um die Befehlshistorie anzuzeigen: ein Doppelklick auf eine Zeile führt den Befehl aus. Der Dialog *Befehlsverlauf* kann auch über das Kontextmenü des Eingabebereichs aufgerufen werden;
- Speichern und löschen Sie den Befehlsverlauf. Der Verlauf wird in der Datei `console_history.txt` unter dem aktiven *Benutzerprofil-Ordner* gespeichert;
- Geben Sie die folgenden speziellen Befehle ein:
  - `?` um eine Hilfe der Python-Konsole anzuzeigen
  - `_api` um die QGIS C++ API Dokumentation zu öffnen oder `_api(object)` für eine spezifische Objektdokumentation (in QGIS C++ API oder Qt API Dokumentation)
  - `_pyqgis` um die QGIS Python API Dokumentation zu öffnen oder `_pyqgis(object)` für eine spezifische Objektdokumentation (in der QGIS Python API oder Qt API Dokumentation)
  - `_cookbook` um das PyQGIS Cookbook zu öffnen.
  - `!` gefolgt von einem Befehl, um Shell-Befehle von der Python-Konsole auszuführen. Die Konsole startet einen Unterprozess und leitet dessen Ausgabe an den Python-Konsolen Ausgang weiter. Während der Unterprozess läuft, schaltet die Python- Konsole Input in den STDIN-Modus und leitet eingegebene Zeichen an den Kindprozess weiter. Dadurch ist es möglich, eine Bestätigung zu senden, wenn das untergeordnete Programm darum bittet. Wenn sich die Konsole im STDIN-Modus befindet, wird der Unterprozess durch Drücken von `Strg+C` beendet. Es ist auch möglich, das Ergebnis eines Befehls in einer Variablen mit der Syntax `var = !cmd` zu beeinflussen.

```
>>> !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)
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
>>> sql_formats = !ogrinfo --formats | grep SQL
>>> sql_formats
['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']
```

Tipp

Ausgeführte Befehle aus dem Bedienfeld wiederverwenden

Sie können Codeschnipsel aus dem Bedienfeld ausführen, indem Sie einen Text auswählen und `Strg+E` drücken. Es spielt keine Rolle, ob der ausgewählte Text den Interpreter-Prompt enthält (`>>>`, `...`).

25.3.2 Der Code-Editor

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.

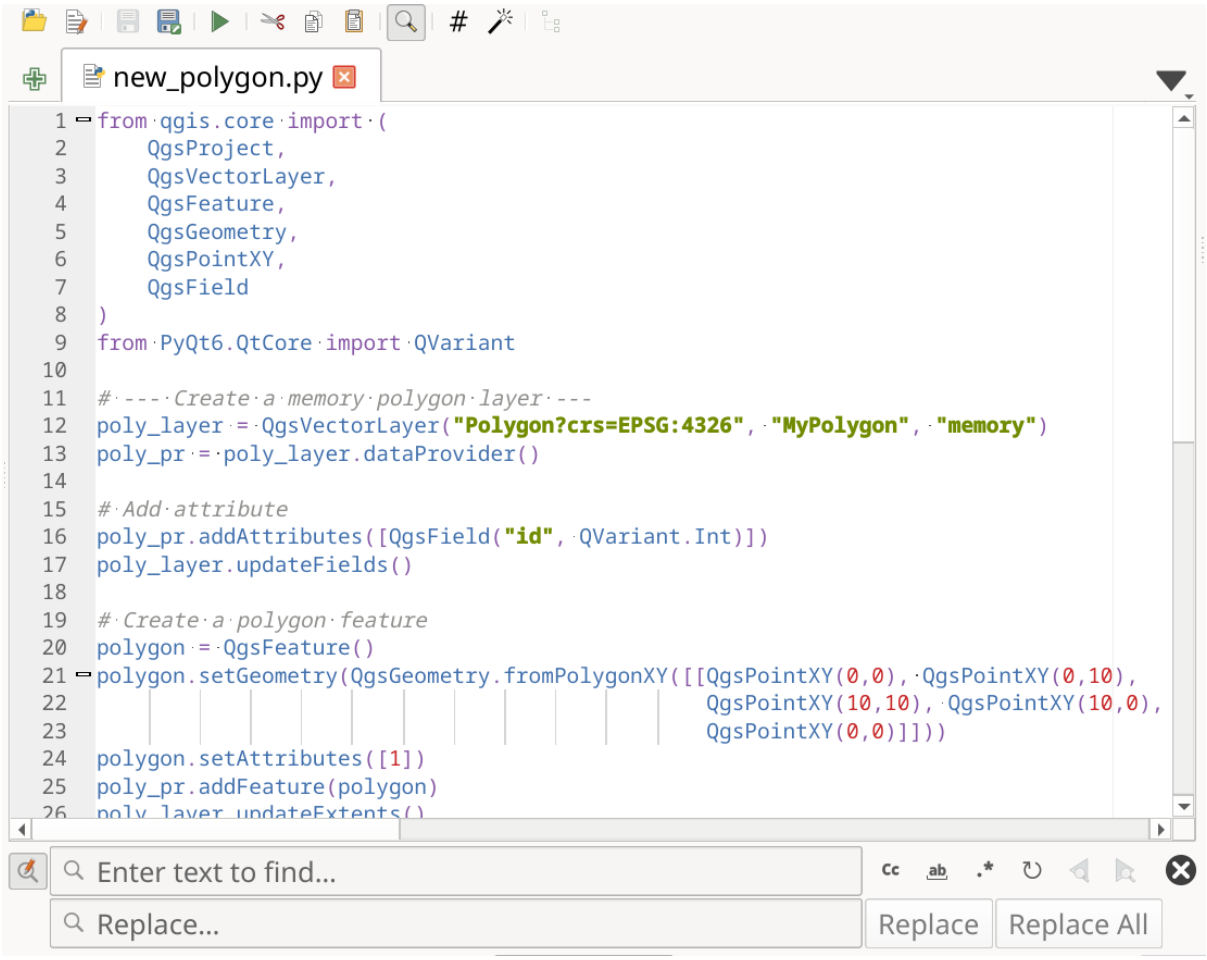


Abb. 25.18: Der Python-Konsolen-Editor

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.

Tipp

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.

Tab. 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	Werkzeugleiste
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. • Verwenden Sie die Standard-Verknüpfung der Desktop-Umgebung für die Suche nach dem Nächsten/Vorherigen (`Strg+G` und `Umschalt+Strg+G`); • Automatische Suche nach der ersten Übereinstimmung bei der Eingabe in das Suchfeld; • Setzt die anfängliche Suchzeichenfolge beim Öffnen der Suche auf die Auswahl; • Enable the *Use Regular Expressions* option in the search bar to use regular expressions instead of plain text matching; • 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. • Durch Drücken von `Esc` wird die Suchleiste geschlossen.	

Fortsetzung auf der nächsten Seite

Tab. 25.1 – Fortsetzung der vorherigen Seite

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 *Python* dialog.	

KAPITEL 26

Anhänge

26.1 Anhang C: QGIS-Dateiformate

26.1.1 QGS/QGZ - Das QGIS-Projekt-Dateiformat

Das **QGS**-Format ist ein XML-Format zur Speicherung von QGIS-Projekten. Das **QGZ**-Format ist ein komprimiertes (Zip-)Archiv, das eine QGS-Datei und eine QGD-Datei enthält. Die **QGD**-Datei ist die zugehörige Sqlite-Datenbank des qgis-Projekts, die Hilfsdaten für das Projekt enthält. Wenn es keine Hilfsdaten gibt, ist die QGD-Datei leer.

Eine QGIS-Datei enthält alles, was zum Speichern eines QGIS-Projekts erforderlich ist, einschließlich:

- Projekttitel
- Projekt-KBS
- Layer-Baum
- Einrastoptionen
- Beziehungen
- Kartenausschnitt
- Projekt-Modelle
- legend
- Mapview-Docks (2D und 3D)
- Layer mit Links zu den zugrundeliegenden Datensätzen (Datenquellen) und andere Layer-Eigenschaften wie Ausdehnung, SRS, Verbindungen, Stile, Renderer, Mischmodus, Deckkraft und mehr.
- Projekteigenschaften

Die folgenden Abbildungen zeigen die Top-Level-Tags in einer QGS-Datei und das erweiterte `ProjektLayer`-Tag.

```
−<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>
```

Abb. 26.1: Die Top-Level-Tags in einer QGS-Datei

```
−<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>
```


Abb. 26.2: Das erweiterte Top-Level ProjectLayer-Tag einer QGS-Datei

26.1.2 QLR – die QGIS-Layer-Definitionsdatei

Eine Layerdefinitionsdatei (QLR) ist eine XML-Datei, die zusätzlich zu den QGIS-Stil-Informationen einen Verweis auf die Layer-Datenquelle enthält.

Der Anwendungsfall für diese Datei ist einfach: eine einzelne Datei zu haben, mit der die Datenquelle geöffnet und alle zugehörigen Stil-Informationen geladen werden können. QLR-Dateien erlauben es auch, die zugrundeliegende Datenquelle mit einer leicht zu öffnenden Datei zu maskieren.

Ein Beispiel für die Verwendung von QLR ist das Öffnen von MS SQL-Layern. Anstatt den MS SQL-Verbindungsdialog aufrufen, verbinden, auswählen, laden und schließlich stylen zu müssen, können Sie einfach eine .qlr-Datei hinzufügen, die auf den richtigen MS SQL-Layer mit allen erforderlichen Stilen verweist.

In Zukunft kann eine .qlr-Datei einen Verweis auf mehr als eine Ebene enthalten.

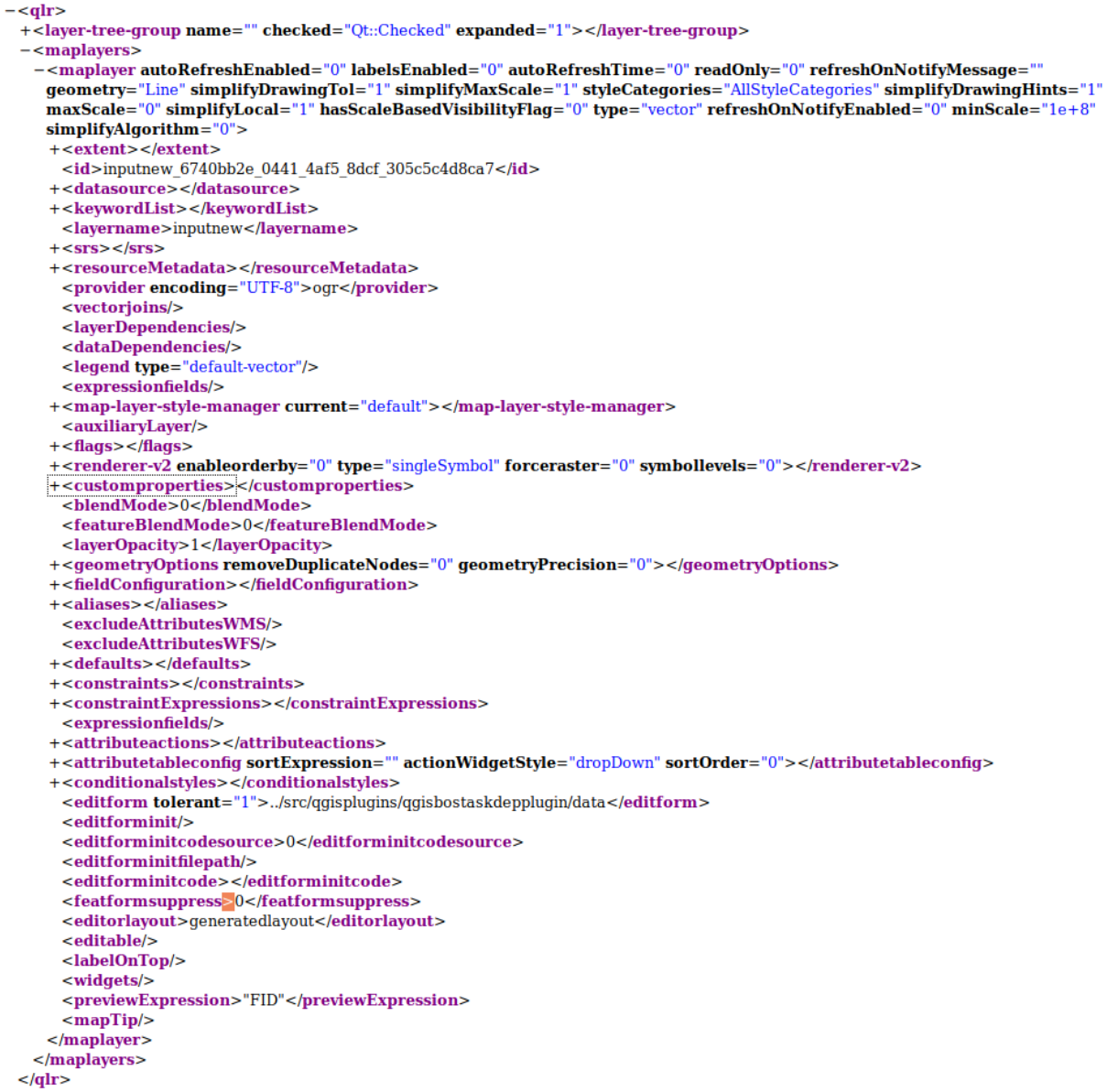


Abb. 26.3: Die Tags der obersten Ebene einer QLR-Datei

26.1.3 QML - Das QGIS-Stil-Dateiformat

QML ist ein XML-Format zum Speichern von Layer-Stilen.

Eine QML-Datei enthält alle Informationen, die QGIS für das Rendern von Feature-Geometrien verarbeiten kann, einschließlich Symboldefinitionen, Größen und Drehungen, Beschriftung, Deckkraft und Mischmodus und mehr.

Die folgende Abbildung zeigt die Top-Level-Tags einer QML-Datei (nur mit `Renderer_v2` und seinem `Symbol` Tag expandiert).

```
−<qgis version="3.4.13-Madeira" styleCategories="AllStyleCategories" readOnly="0" maxScale="0"
 labelsEnabled="0" simplifyDrawingHints="1" hasScaleBasedVisibilityFlag="0" simplifyDrawingTol="1"
 simplifyMaxScale="1" minScale="1e+8" simplifyAlgorithm="0" simplifyLocal="1">
 +<flags></flags>
 −<renderer-v2 symbollevels="0" enableorderby="0" type="singleSymbol" forceraster="0">
   −<symbols>
     +<symbol clip_to_extent="1" name="0" alpha="1" type="fill" force_rhr="0"></symbol>
    </symbols>
    <rotation/>
    <sizescale/>
  </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/>
  <layerGeometryType>2</layerGeometryType>
</qgis>
```

Abb. 26.4: Die Top-Level-Tags einer QML-Datei (nur das Renderer_v2-Tag mit seinem Symbol-Tag ist expandiert)

26.2 Anhang D: QGIS R Skriptsyntax

Contributed by Matteo Ghetta - funded by Scuola Superiore Sant'Anna

Das Schreiben von R-Skripten in Processing ist wegen seiner speziellen Syntax etwas knifflig.

Ein Verarbeitungsskript für die R beginnt mit der Definition seiner Eingaben und Ausgaben, denen jeweils doppelte Hashzeichen (##) vorangestellt sind.

Vor den Eingaben kann die Gruppe, in der der Algorithmus platziert werden soll, angegeben werden. Wenn die Gruppe bereits existiert, wird der Algorithmus zu ihr hinzugefügt; ist dies nicht der Fall, wird die Gruppe erstellt. Im folgenden Beispiel lautet der Name der Gruppe Meine Gruppe:

```
##Meine Gruppe=group
```

26.2.1 Eingaben

Alle Eingabedaten und Parameter müssen angegeben werden. Es gibt verschiedene Arten von Eingaben:

- Vektor: `##Layer = vector`
- Vektorfeld: `##F = Field Layer` (wobei *Layer* der Name eines Eingabe-Vektorayers ist, zu der das Feld gehört))
- Raster: `##r = raster`
- Tabelle: `##t = table`
- Zahl: `##Num = number`
- String: `##Str = string`
- Boolean: `##Bol = boolean`
- Elemente in einem Dropdown-Menü. Die Elemente müssen durch Semikolons `;` getrennt werden: `##type=selection point;lines;point+lines`

26.2.2 Ausgaben

Genau wie die Eingaben müssen alle Ausgaben am Beginn des Skriptes definiert werden:

- Vektor: `##output= output vector`
- Raster: `##output= output raster`
- Tabelle: `##output= output table`
- Diagramme: `##output_plots_to_html` (##showplots in früheren Versionen)
- Um die R-Ausgabe im *Ergebnis-Viewer* anzuzeigen, schreiben Sie > vor den Befehl, dessen Ausgabe Sie anzeigen möchten.

26.2.3 Syntaxzusammenfassung für QGIS R-Skripte

Es wird eine Reihe von Eingabe- und Ausgangsparametertypen angeboten.

Typen von Eingabeparametern

Parameter	Syntax-Beispiel	Rückgabeobjekte
Vektor	Layer = Vektor	sf-Objekt (oder SpatialDataFrame-Objekt, wenn ##load_vector_using_rgdal angegeben ist)
vector point	Layer = Vektor Punkt	sf-Objekt (oder SpatialDataFrame-Objekt, wenn ##load_vector_using_rgdal angegeben ist)
Vektor Linie	Layer = Vektor Linie	sf-Objekt (oder SpatialDataFrame-Objekt, wenn ##load_vector_using_rgdal angegeben ist)
Vektor Polygon	Layer = Vektor Polygon	sf-Objekt (oder SpatialDataFrame-Objekt, wenn ##load_vector_using_rgdal genutzt wird)
Vielfach Vektor	Layer = Vielfach Vektor	sf-Objekt (oder SpatialDataFrame-Objekt, wenn ##load_vector_using_rgdal angegeben ist)
Tabelle	Layer = Tabelle	Daten-Tabellen-Umwandlung von csv, voreingestelltes Objekt der `read.csv` Funktion
Feld	Feld = Feld Layer	Name des ausgewählten Feldes, z.B. `"Fläche"`
raster	Layer = Raster	RasterBrick-Objekt, voreingestelltes Objekt des `Raster` Pakets
Vielfach Raster	Layer = Vielfach Raster	RasterBrick-Objekt, voreingestelltes Objekt des `Raster` Pakets
Zahl	N = Zahl	gewählte Ganzzahl oder Fließkommazahl
String	S = String	in der Box eingefügter String
Longstring	LS = Longstring	in der Box eingefügter String, kann länger als ein normaler String sein
Auswahl	S = Auswahl erste;zweite;third;dritte	String des im Dropdown-Menü gewählten Objektes
crs	C = CRS/KBS	String des sich ergebenen KBS im Format: `"EPSG:4326"`
extent	E = Ausdehnung	Ausdehnungsobjekt des `raster` Pakets, man kann Werte als *E@xmin`* extrahieren
Punkt	P = Punkt	beim Klicken auf die Karte erhält man die Koordinaten des Punktes
Datei	F = Datei	Pfad der gewählten Datei, z.B. „/home/matteo/file.txt“
folder	F = Ordner	Pfad des gewählten Ordners, z.B „/home/matteo/Downloads“

Ein Parameter kann **OPTIONAL** ein, was bedeutet, dass er ignoriert werden kann.

Um eine Eingabe als optional festzulegen, fügen Sie den String `optional` **vor** die Eingabe ein, z.B.:

```
##Layer = vector
##Field1 = Field Layer
##Field2 = optional Field Layer
```

Arten von Ausgabeparametern

Parameter	Syntax-Beispiel
Vektor	Output = Ausgabe-Vektor
raster	Output = Ausgabe-Raster
Tabelle	Output = Ausgabe-Tabelle
Datei	Output = Ausgabe-Datei

Bemerkung

Sie können Diagramme als `png` aus dem *Prozess-Ergebnis-Viewer* oder direkt über die Algorithmus-Schnittstelle speichern.

Hauptteil des Skripts

Der Skripttext folgt der R-Syntax, und der Bereich „Protokoll“ kann Helfen Sie Ihnen, wenn mit Ihrem Skript etwas nicht stimmt.

Denken Sie daran, dass Sie im Skript alle zusätzlichen Bibliotheken laden müssen:

```
library(sp)
```

26.2.4 Beispiele

Beispiel mit Vektorausgabe

Sehen wir uns einen Algorithmus aus der Onlinesammlung an, der zufällige Punkte innerhalb der Ausdehnung eines Eingabelayers erstellt:

```
##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)
```

Erklärung (pro Zeile im Skript):

1. `Point pattern analysis` ist die Gruppe des Algorithmus
2. `Layer` ist der **Eingabevektorlayer**
3. `Size` ist ein **numerischer** Parameter mit dem Standardwert 10
4. `Output` ist der **Vektorlayer** der vom Algorithmus erstellt wird
5. `library(sp)` lädt die **sp-Bibliothek**
6. `spatpoly = as(Layer, "Spatial")` übersetzen in ein sp-Objekt
7. Aufruf der `spsample`-Funktion der `sp`-Bibliothek und Ausführung unter Verwendung der oben definierten Eingabe (`Layer` und `Size`)
8. Erstellen eines *SpatialPointsDataFrame*-Objekts unter Nutzung der `SpatialPointsDataFrame`-Funktion
9. Erstellen des Ausgabevektor-Layers mit der `st_as_sf`-Funktion

Das wars! Führen Sie einfach den Algorithmus mit einem Vektorlayer aus, den Sie in der QGIS-Legende haben. Wählen Sie die Anzahl der zufälligen Punkte. Der Ergebnis-Layer wird der Karte hinzugefügt.

Beispiel mit Rasterausgabe

Das folgende Skript führt mit der ``autoKrige`-Funktion des R-Pakets ein normales Kriging aus, um eine Raster-Karte mit interpolierten Werten aus einem angegebenen Feld des Eingabe-Punkt-Vektorlayers zu erstellen. Es wird zuerst das Kriging-Modell berechnet und dann ein Raster. erstellt Das Raster wird mit der ``raster`-Funktion des R-Raster-Pakets erstellt:

```
##Basic statistics=group
##Layer=vector point
##Field=Field Layer
##Output=output raster
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
##load_vector_using_rgdal
require("automap")
require("sp")
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
```

Durch die Verwendung von `##load_vector_using_rgdal`, wird der Eingabevektorlayer als `SpatialDataFrame`-Objekt zur Verfügung gestellt. So vermeiden wir es, ihn aus einem *sf*`-Objekt übersetzen zu müssen.

Beispiel mit Tabellenausgabe

Lassen Sie uns den Algorithmus `Summary Statistics` bearbeiten, so dass als Ausgabe eine Tabelle (csv) erzeugt wird.

Der Hauptteil des Skriptes sieht wie folgt aus:

```
##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
```

Die dritte Zeile gibt als Eingabe **Vektor Feld** vor. In der vierten Zeile wird die Ausgabe des Algorithmus als Tabelle festgelegt.

In der letzten Zeile wird das im Skript erstellte Objekt `Stat` in eine `csv` Tabelle umgewandelt.

Beispiel mit Konsolenausgabe

Wir können das vorherige Beispiel verwenden; aber anstatt eine Tabelle zu erstellen, wird das Ergebnis im **Ergebnis-Viewer** ausgedruckt:

```
##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]]),
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
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)
>Summary_statistics
```

Das Skript ist bis auf zwei Bearbeitungen genau das gleiche wie das obige:

1. keine Ausgabe-Datei angegeben (die vierte Zeile wurde entfernt)
2. die letzte Zeile beginnt mit ``>` und weist *Processing* an, das Objekt über den *Ergebnis-Viewer* verfügbar ist

Beispiel mit Diagramm

Um Diagramme zu erstellen, müssen Sie den `##output_plots_to_html`-Parameter wie im folgenden Skript verwenden:

```
##Basic statistics=group
##Layer=vector
##Field=Field Layer
##output_plots_to_html
####output_plots_to_html
qqnorm(Layer[[Field]])
qqline(Layer[[Field]])
```

Das Skript verwendet ein Feld (`Field`) eines Vektorlayers (`Layer`) als Input und erstellt ein *QQ-Diagramm* (um die Normalität der Verteilung zu testen).

Der Plot wird automatisch zum *Prozess-Ergebnis-Viewer* hinzugefügt.

26.3 Anhang E: Netzwerkverbindungen der QGIS-Anwendung

Dies ist eine Liste der vorkonfigurierten/automatisierten Netzwerkverbindungen, die QGIS herstellt. Einige der Verbindungen sind benutzerinitiiert, da sie eine Aktion des Benutzers erfordern, andere erfolgen automatisch.

Name	Zweck	UX-Modus	Server	An den Server gesendete Informationen	Auf dem Server gespeicherte Informationen
qgis.org					
Hilfe zur Python-API	Browser-PyQGIS-Dokumentation	Vom Benutzer initiiert	https://qgis.org/pyqgis/3.44/search	IP, QGIS-Version, Betriebssystem	IP im Serverprotokoll
version					
Prüfung der neuen Version	Benutzer über die Verfügbarkeit neuer QGIS-Versionen benachrichtigen	Automatisch	https://version.qgis.org	IP, QGIS-Version, Betriebssystem	IP im Serverprotokoll
feed.qg					
QGIS-Feed	QGIS-Nachrichten aus dem Feed abrufen	Automatisch	https://feed.qgis.org	IP, QGIS-Version, Sprachcode, Zeitstempel des letzten Downloads, Betriebssystem	IP im Serverprotokoll; QGIS-Version, Betriebssystem und IP werden aggregiert und zur Erfassung einiger Statistiken verwendet
plugins					
Nach Plugin-Updates suchen	Benutzer über Plugin-Updates benachrichtigen	Vom Benutzer initiiert/autor (konfigurierb	https://plugins.qgis.org	IP, QGIS-Version, Betriebssystem	IP im Serverprotokoll
Plugin-Liste	Abufen der Plugin-Liste	Vom Benutzer initiiert	https://plugins.qgis.org	IP, QGIS-Version, Betriebssystem	IP im Serverprotokoll
Plugin-Installat	Plugin-Paket herunterladen und installieren	Vom Benutzer initiiert	https://plugins.qgis.org	IP, QGIS-Version, Betriebssystem	Plugin-Download-Zähler um eins erhöhen
Stile	Auflisten von Benutzerstilen	Vom Benutzer initiiert	https://hub.qgis.org/styles	IP, QGIS-Version, Betriebssystem	Increase download counter by one
3. Partei					
Geländ	Erstellen eines DEM für 3D-Ansichten	Vom Benutzer initiiert	https://s3.amazonaws.com/elevation-tiles-pr terrarium/{z}/{x}	IP, QGIS-Version, Betriebssystem	Siehe Amazon TOS
Google Map-Geocod	Geocoding-Dienste	Vom Benutzer initiiert	https://maps.googleapis.com/maps/api/geocode/json	IP, QGIS-Version, Betriebssystem	Siehe Google Maps API TOS
Nomina Geocod	Geocoding-Dienste	Vom Benutzer initiiert	https://nominatim.qgis.org/ui/search.html	IP, QGIS-Version, Betriebssystem	
Geodäti Gitter	Hinzufügen eines neuen PROJ-Rasters	Vom Benutzer initiiert	https://cdn.proj.org	IP, PROJ-Version	Zugriffsprotokolle werden nach einem Tag endgültig gelöscht

KAPITEL 27

Literatur und Internetreferenzen

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.